



SAS[®] Studio: 사용 설명서

2023.05*

* 이 문서는 추가적인 소프트웨어 버전에 해당될 수 있습니다. 이 문서를 [SAS Help Center](#) 에서 열고 배너에서 버전을 클릭하여 사용 가능한 모든 버전을 확인하십시오.

SAS[®] 문서
2023년 5월 28일 (일)

이 설명서의 올바른 참고문헌 인용은 다음과 같습니다. SAS Institute Inc. 2023. *SAS® Studio: 사용 설명서*. Cary, NC: SAS Institute Inc.

SAS® Studio: 사용 설명서

Copyright © 2023, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

May 2023

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

v_030-P1:webeditorug

목차

1장 / SAS Studio 소개	1
SAS Studio 정보	2
PWA(Progressive Web App)로서의 SAS Studio	4
SAS Studio 인터페이스 이해	5
작업 공간 구성 파일을 사용하여 시작을 사용자 정의	6
탐색 영역 사용	7
작업 영역 사용	17
SAS Studio에서 검색하기	22
콘솔 사용	24
백그라운드 실행 기능 사용	25
실행 상태 창 사용	26
문서 복구 창 사용	27
프로그램, 작업, 질의 또는 플로우를 작업으로 스케줄링	29
스케줄 작업 창 사용	29
영역 이해	31
자동실행 파일 편집	33
사용자 정의 SAS 코드 삽입	33
SAS 세션 재설정	34
바로 가기 키 관리	34
컴퓨팅 컨텍스트 변경	34
2장 / 프로그램 작업	37
SAS 코드 편집기 정보	38
Python 코드 편집기 정보	38
프로그램 열기 및 생성	40
DATA 스텝 디버거 사용	55
Snippet 작업	62
코드 편집기 사용자 정의	73
3장 / 질의 작업	75
질의란?	76
독립형 질의 및 플로우 질의의 차이점 이해	76
이전 버전의 SAS Studio에서 저장된 질의 마이그레이션	77
독립형 질의 생성	78
조인 이해	79
데이터 선택	83
데이터 필터링	89
출력 관리	92
FedSQL 질의 생성	102
4장 / 데이터 작업	105
테이블 뷰어 정보	105
데이터 열기 및 보기	107
테이블을 생성하는 데 사용된 코드 보기	111
테이블 뷰어의 작업 또는 질의에서 데이터 열기	111
데이터 새로 고침	113
데이터 정렬 및 고정	113
데이터 필터링	114

데이터 가져오기	118
데이터 내보내기	124
5장 / 결과 및 출력 데이터 작업	127
결과 보기	127
기본 SAS Studio 출력	128
다른 사용자에게 결과 보내기	129
출력 데이터 뷰	131
SAS Output Delivery System 및 SAS ODS Statistical Graphics 정보	132
6장 / SAS Studio의 Git 통합 이해	133
SAS Studio의 Git 통합 정보	134
Git 프로파일 작업	135
Git 레파지토리 복제 및 열기	137
로컬 레파지토리 생성	139
SAS Studio의 Git 레파지토리 탭 이해	140
커밋 히스토리 보기	141
변경 사항을 로컬 레파지토리에 커밋	142
파일 풀(pull) 및 페치(fetch)	146
파일 푸시	146
로컬 레파지토리 재설정	147
Git에서 브랜치 작업	147
브랜치 생성	148
브랜치 체크아웃	149
병합 및 리베이스 이해	150
브랜치 병합	150
현재 브랜치 리베이스	151
변경 사항 스테시	151
병합 충돌 해결	152
레파지토리 삭제	154
샘플 Git 워크플로우 시나리오	155
7장 / SAS Studio의 작업 이해	167
작업이란?	167
작업을 실행하는 방법	167
사전 정의된 작업 편집	168
사용자 정의 작업 생성	169
작업 코드에 대한 옵션 지정	170
부록 1 / SAS Studio 사용자 정의	171
SAS Studio 사용자 정의 정보	171
글로벌 설정 생성	172
우선 설정 지정	174
부록 2 / 표현식 빌더 사용	187
표현식 구성	187
부록 3 / 텍스트 인코딩 옵션 및 언어 매핑	189
텍스트 인코딩 및 언어 매핑 정보	189
텍스트 인코딩 옵션 및 언어 매핑	189
부록 4 / 사용자 정의된 출력 환경	191
개요	191
다른 출력 대상에 대한 출력 생성	192
다른 위치에 결과 전송	192
출력에 대한 사용자 정의 스타일 사용	193

기본값 대신 이미지 출력형식 사용	193
드릴다운 그래프 생성	193
애니메이션 GIF 또는 SVG 이미지 생성	194
부록 5 / SAS Studio 명령행	195
명령행 인터페이스 정보	195
표준 영역의 명령	196
대화식 영역의 명령	201
콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅	204
부록 6 / SAS Studio로 전환	207
SAS Studio 3.x 사용자를 위한 팁	208
SAS Enterprise Guide 사용자를 위한 팁	211
SAS Data Integration Studio 사용자를 위한 팁	213
SAS Universal Viewer 사용자를 위한 팁	215
부록 7 / SAS Studio 기능 요약	217
SAS Studio 라이선스 정보	217
플로우 기능 요약	217
사용자 정의 스텝 기능 요약	222

SAS Studio 소개

SAS Studio 정보	2
PWA(Progressive Web App)로서의 SAS Studio	4
PWA(Progressive Web App) 사용의 이점	4
SAS Studio를 PWA(프로그레시브 웹 앱)로 설치	4
PWA에 대한 타임아웃 설정	5
SAS Studio 인터페이스 이해	5
작업 공간 구성 파일을 사용하여 작업을 사용자 정의	6
탐색 영역 사용	7
탐색 영역 사용 정보	7
열린 파일 액세스	8
탐색기 사용	8
스텝 작업	10
작업 사용	10
Snippet 작업	11
라이브러리 작업	11
파일 참조 사용	15
Git 레파지토리 작업	17
탐색 영역 사용자 정의	17
작업 영역 사용	17
작업 영역 사용 정보	17
작업 영역에서 탭 생성	18
최근에 연 항목 및 즐겨찾는 항목 액세스	19
작업 영역 사용자 정의	20
작업 영역에서 탭 재정렬	21
SAS Studio에서 검색하기	22
콘솔 사용	24
백그라운드 실행 기능 사용	25
백그라운드 실행 기능 정보	25
백그라운드 실행으로 파일 실행	26
백그라운드 실행 사용자 정의	26
실행 상태 창 사용	26
문서 복구 창 사용	27

<i>프로그램, 작업, 질의 또는 플로우를 작업으로 스케줄링</i>	29
프로그램, 작업, 질의 또는 플로우 스케줄링	29
<i>스케줄 작업 창 사용</i>	29
<i>영역 이해</i>	31
<i>자동실행 파일 편집</i>	33
<i>사용자 정의 SAS 코드 삽입</i>	33
<i>SAS 세션 재설정</i>	34
<i>바로 가기 키 관리</i>	34
<i>컴퓨팅 컨텍스트 변경</i>	34

SAS Studio 정보

SAS Studio는 웹 브라우저를 통해 액세스할 수 있는 SAS용 개발 응용 프로그램입니다. SAS Studio를 사용하여 데이터 파일, 라이브러리 및 기존 프로그램에 액세스하고 새로운 프로그램을 작성할 수 있습니다. 플로우에서 작업을 정리하고 SAS Studio의 사전 정의된 작업을 사용하여 SAS 코드를 생성할 수도 있습니다. 프로그램이나 작업을 실행하면 SAS Studio가 SAS 서버에 연결하여 SAS 코드를 처리합니다. SAS 서버는 클라우드 환경의 호스트 서버이거나 로컬 환경의 원격 서버일 수 있습니다. 코드가 처리되면 결과가 브라우저 내 SAS Studio로 반환됩니다.



SAS Studio는 Apple Safari, Google Chrome, Microsoft Edge, Microsoft Edge on Chromium 및 Mozilla Firefox 등 여러 웹 브라우저를 지원합니다. 자세한 내용은 [SAS Viya for Containers: Deployment Guide](#)의 “Web Browsers”에서 참조하십시오.

사용자 고유의 SAS 프로그램을 작성하고 실행할 뿐만 아니라 SAS Studio와 함께 제공되는 사전 정의된 작업을 사용하여 데이터를 분석할 수도 있습니다. 작업은 SAS 프로시저를 기반으로 하며, 가장 일반적으로 사용되는 그래프 및 분석 프로시저 중 일부에 액세스할 수 있게 해 줍니다. 기본 작업 템플릿을 사용하여 사용자 고유의 작업을 작성할 수도 있습니다.

SAS Studio에는 표준 영역과 대화식 영역의 두 가지 다른 영역이 포함되어 있습니다. **영역**은 특정 사용자 유형의 요구 사항에 맞게 사용자 정의된 사전 결정 기능 집합입니다. 특정 영역을 선택하면 인터페이스에서 사용할 수 있는 선택 항목의 범위를 좁혀 주기적으로 사용해야 하는 기능에 포커스를 맞출 수 있습니다. 기본적으로 SAS Studio를 열면 표준 영역이 선택되어 있습니다. SAS Studio를 연 후에는 도구 모음의 **옵션** 메뉴를 사용하여 영역을 변경할 수 있습니다. 자세한 내용은 “[영역 이해](#)” (31페이지)에서 확인하십시오.

PWA(Progressive Web App)로서의 SAS Studio

PWA(Progressive Web App) 사용의 이점

SAS Studio를 PWA(Progressive Web App)로 사용하면 다음과 같은 이점이 있습니다.

- 앱 지속성 - Windows 시작 메뉴에서 PWA로 SAS Studio에 액세스하면 세션 시간이 초과되지 않습니다. 웹 브라우저를 통해 SAS Studio에 액세스하면 일정 시간 동안 사용하지 않으면 세션이 타임아웃되고 로그아웃됩니다. 자세한 내용은 [“PWA에 대한 타임아웃 설정” \(5페이지\)](#)에서 참조하십시오.
- 성능 - SAS Studio를 PWA로 설치하면 일반적으로 브라우저에서 액세스할 때보다 빠르게 시작됩니다.
- 데스크톱 환경 - SAS Studio를 PWA로 설치하면 기존 설치의 모든 이점이 제공됩니다. **시작** 메뉴 또는 작업 표시줄에서 SAS Studio를 시작할 수 있으며 SAS Studio의 올바른 인스턴스를 찾기 위해 수많은 탭을 정렬할 필요가 없습니다.
- 이름 바꾸기 - 각 PWA 인스턴스의 이름을 변경하여 개발, 테스트 또는 프로덕션 서버와 같은 다양한 환경에 빠르게 액세스할 수 있습니다.

SAS Studio를 PWA(프로그레시브 웹 앱)로 설치

SAS Studio는 TLS 및 HTTPS가 활성화된 상태로 배포되어야 합니다.

SAS Studio를 PWA로 설치하려면 다음을 수행하십시오.

- 1 Chromium 기반 웹 브라우저에서 SAS Drive 또는 SAS Visual Analytics와 같은 다른 SAS 응용 프로그램을 엽니다.
- 2 웹 브라우저의 **추가** 메뉴를 열고 **SAS 설치**를 선택합니다.

SAS 응용 프로그램은 이제 **SAS** 데스크톱 프로그램으로 설치됩니다. 하나의 웹 응용 프로그램이 PWA로 설치되면 SAS Studio를 포함한 다른 모든 SAS 응용 프로그램을 PWA에서 액세스할 수 있습니다.

주: 사이트 URL로 직접 이동하여 SAS Studio를 시작하거나 SAS 드라이브 설정에서 SAS Studio가 초기 화면으로 설정된 경우에는 Chrome에서 **SAS 설치** 버튼을 사용할 수 없습니다.

SAS Studio용 PWA를 제거하려면 다음을 수행합니다.

- 1 SAS Studio용 PWA를 엽니다.
- 2 PWA 메뉴에서 **SAS 제거**를 선택합니다.

PWA에 대한 타임아웃 설정

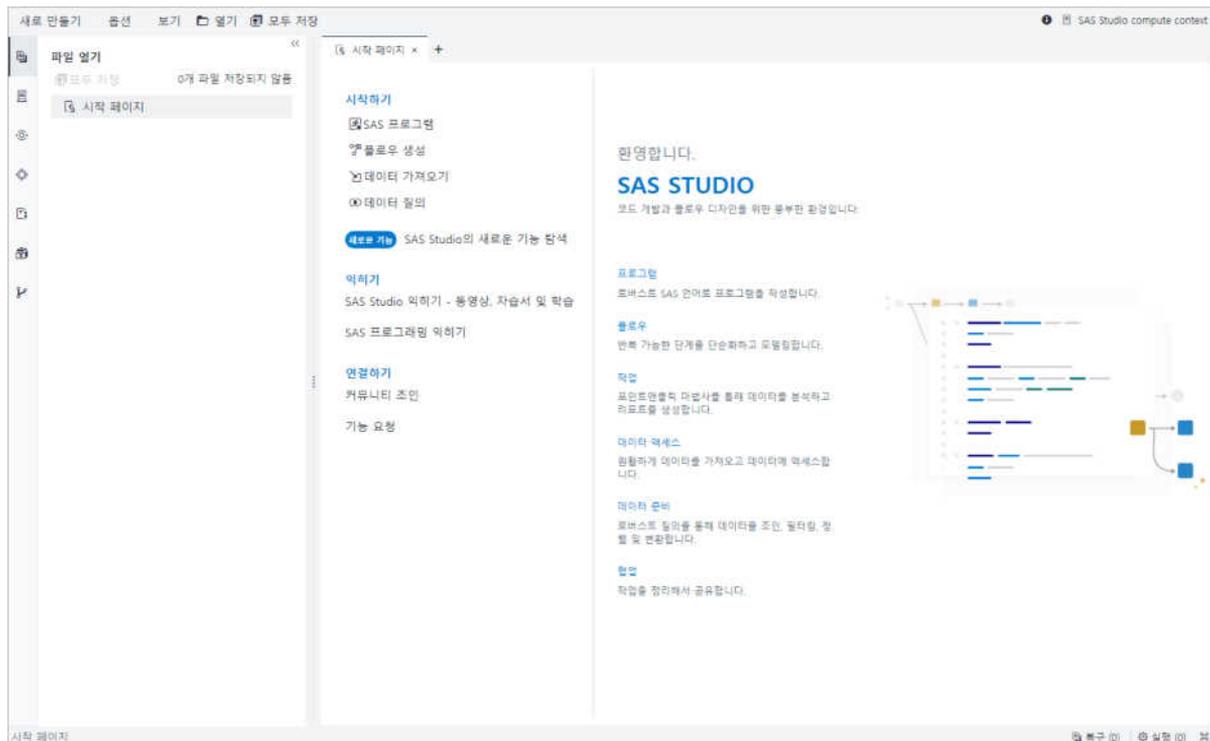
PWA에 대한 타임아웃을 지정하기 위해 SAS 관리자는 SAS Environment Manager에서 다음 구성 속성을 설정합니다:

- EnablesPWATimeout 구성 속성. 자세한 내용은 [SAS Viya Platform: Configuration Reference](#)의 “HTML Clients”에서 참조하십시오.
- sas.studio.pwaSessionTimeout 구성 속성. 자세한 내용은 [SAS Studio: Administrator's Guide](#)의 “General Configuration Properties”에서 참조하십시오.

SAS Studio 인터페이스 이해

SAS Studio에 로그인하면 시작 페이지를 사용할 수 있으므로 새로운 SAS 프로그램 작성, 플로우 구성, 데이터 가져오기 또는 질의 생성을 빠르게 시작할 수 있습니다. 아이콘을 사용하여 탐색 영역의 다른 섹션에 액세스합니다.

주: SAS Studio에서 로그아웃하려면 응용 프로그램 도구 모음 가장 오른쪽에 있는 사용자 이름의 처음 이니셜이 표시된 버튼을 클릭하고 **로그아웃**을 선택합니다. 웹 브라우저에서 뒤로 가기 버튼을 사용하지 마십시오.



맨 위의 응용 프로그램 표시줄에서 다른 SAS 응용 프로그램에 액세스할 수 있습니다. 검색 아이콘을 클릭하여 SAS 응용 프로그램에서 항목을 검색합니다.

SAS Studio의 기본 창은 왼쪽의 탐색 영역과 오른쪽의 작업 영역으로 구성됩니다. 탐색 영역에서는 열려 있는 파일, 폴더 바로 가기, 파일 시스템 및 SAS 콘텐츠, 스텝, 작업 및 Snippet, Git 레파지토리 액세스 권한이 있는 라이브러리, 파일 참조 등에 쉽게 액세스할 수 있습니다. 기본적으로 **파일 열기** 섹션이 표시됩니다. **파일 참조** 섹션은 기본적으로 표시되지 않지만 **보기** ⇒ **탐색 영역** ⇒ **파일 참조** 메뉴를 사용하여 열 수 있습니다. 자세한 내용은 **“파일 참조 사용” (15페이지)**에서 확인하십시오.

작업 영역은 데이터, 코드, 작업, 로그 및 플로우를 표시하는 데 사용됩니다. 이러한 항목을 열면 해당 항목이 작업 영역의 탭으로 구분된 인터페이스에 창으로 추가됩니다. SAS Studio를 처음 열면 기본적으로 작업 영역에 **시작 페이지** 탭이 표시됩니다. 자세한 내용은 **“작업 영역 사용” (17페이지)**에서 확인하십시오.

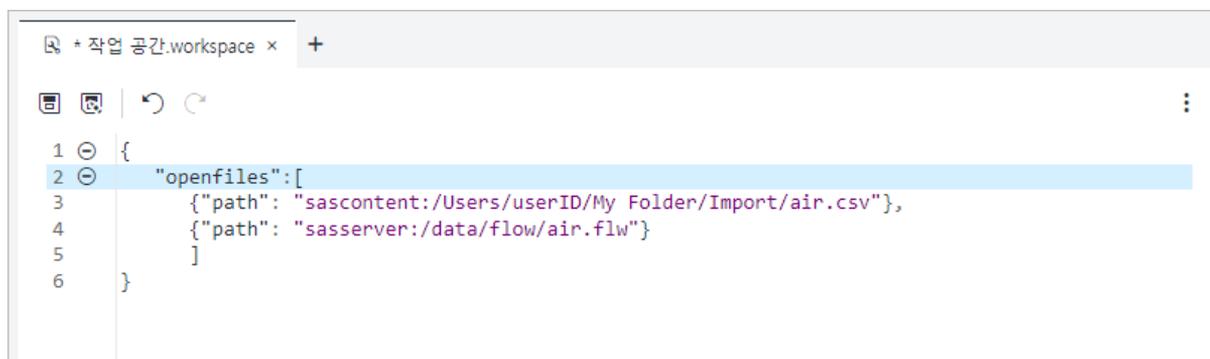
작업 공간 구성 파일을 사용하여 시작을 사용자 정의

시작 시 SAS Studio 세션에서 열리는 하나 이상의 파일을 지정하도록 작업 공간 구성 파일을 생성할 수 있습니다.

- 1 **새로 만들기** ⇒ **추가 파일 형식** ⇒ **작업 공간**을 선택합니다.
- 2 코드 편집기에서 JSON 코드를 입력하여 열려는 파일을 지정합니다. 경로에는 파일이 SAS 콘텐츠 또는 SAS 서버에 있는지가 표시되어야 합니다(예: sascontent:/path/to/filename 또는 sasserver:/path/to/filename).

이 예에서는 SAS Studio 세션이 시작될 때 air.csv 파일과 air.flw가 열려고 합니다.

```
{
  "openfiles":[
    {"path": "sascontent:/Users/userID/My Folder/Import/air.csv"},
    {"path": "sasserver:/data/flow/air.flw"}
  ]
}
```



- 3 작업 공간 구성 파일을 SAS 콘텐츠 또는 SAS 서버에 저장합니다.

작업 공간 구성 파일이 생성된 후 다음 방법 중 하나로 이 작업 공간 구성 파일을 선택해야 합니다.

- 우선 설정에서 **작업 공간 구성 파일 로드** 옵션을 설정합니다. 자세한 내용은 “**시작 기본 설정 지정**” (174페이지)에서 참조하십시오.
- 다음 URL 파라미터를 사용합니다: `sas-studio-workspace=path`. 경로에는 파일이 SAS 콘텐츠 또는 SAS 서버에 있는지가 표시되어야 합니다(예: `sascontent:/path/to/filename` 또는 `sasserver:/path/to/filename`).

주: 우선 설정에서 작업 공간 구성 파일을 지정하고 URL 파라미터를 사용하는 경우 URL 파라미터 값이 사용됩니다.

작업 공간 구성 파일을 사용할 때 작업 공간 구성 파일에서 참조하는 모든 파일 경로는 **탭 열기** 옵션의 설정을 재정의합니다. 다음과 같은 예가 있습니다.

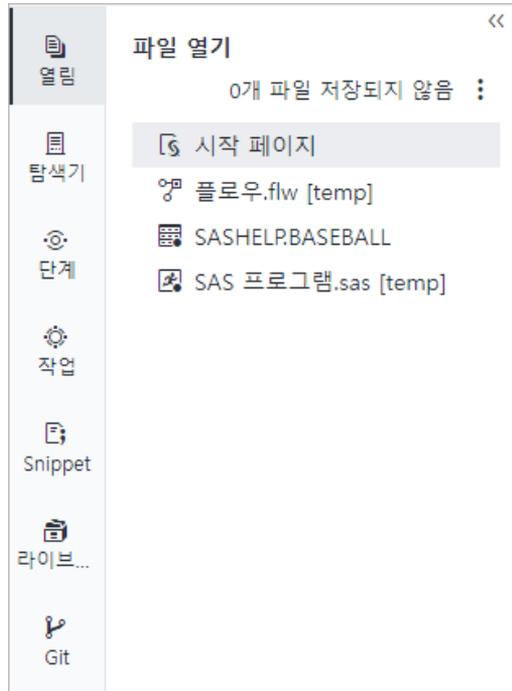
- **탭 열기** 옵션이 **없음**으로 설정되어 있어도 작업 공간 구성 파일에서 참조되는 모든 파일은 시작 시 새로운 탭에서 계속 열립니다.
- **탭 열기** 옵션이 **파일 열기 영역에 나열된 탭 복원**으로 설정된 경우 시작 시 작업 공간 구성 파일에서 참조되는 파일만 열립니다. 파일이 복원되지 않습니다.
- **탭 열기** 옵션이 **새로운 파일 열기**로 설정되어 있을 때만 작업 공간 구성 파일에서 참조되는 모든 파일이 시작 시 새로운 탭에서 열립니다. 새로운 파일이 열리지 않습니다.

탐색 영역 사용

탐색 영역 사용 정보

보려는 섹션을 클릭하여 탐색 영역의 섹션을 펼칠 수 있습니다. 탐색 영역을 닫았다가 다시 열려면 섹션 아이콘을 더블 클릭합니다.

응용 프로그램 탐색 영역에 레이블 표시 옵션을 사용하여 탐색 영역에 아이콘이 있는 레이블을 표시할 수 있습니다. 자세한 내용은 “**일반 우선 설정 지정**” (175페이지)에서 참조하십시오.



열린 파일 액세스

탐색 영역의 **파일 열기** 섹션에서 현재 SAS Studio 세션에서 열어 본 모든 파일을 빠르게 보고 액세스할 수 있습니다. 열린 파일 리스트의 파일에 액세스하려면 보려는 파일을 클릭합니다. **모두 저장**을 클릭하여 리스트에 있는 모든 파일의 변경 내용을 저장할 수 있습니다.

파일 열기 섹션을 사용하여 프로그램 파일을 마우스 오른쪽 버튼으로 클릭하고 **플로우에 추가**를 선택하여 열려 있는 프로그램을 활성 플로우에 추가할 수도 있습니다. 이 옵션은 작업 영역에 활성 플로우가 있을 때만 프로그램에 대해서만 사용할 수 있습니다.

탐색기 사용

탐색 영역의 **탐색기** 섹션에서 폴더 바로 가기, 서버 파일 시스템 및 SAS Content Server 위치의 파일과 폴더에 액세스할 수 있습니다. SAS Studio 2020.1 이상 사용자의 경우 기본 파일 저장 위치는 SAS 콘텐츠입니다. SAS 콘텐츠 위치는 SAS Drive와 같은 다른 SAS Viya 플랫폼의 응용 프로그램과 통합되므로 응용 프로그램 및 다른 SAS 사용자와 파일을 원활하게 공유할 수 있습니다. 자세한 내용은 [SAS Drive: Getting Started](#)에서 참조하십시오.

Git 레파지토리 작업과 같은 다른 작업을 수행하기 위해 파일 시스템에 대한 액세스도 필요한 경우 Kubernetes 및 SAS Studio 관리자는 배포 중에 파일 시스템에 대한 액세스를 생성할 수 있습니다. SAS Studio 5.x에서는 기본적으로 SAS 콘텐츠 및 파일 시스템 저장 위치에 모두 액세스할 수 있습니다. 자세한 내용은 [SAS Studio: Administrator's Guide](#)의 ["Creating Persistent File Storage"](#)에서 확인하십시오.

주: SAS Content Server의 파일에 액세스하는 기능은 SAS Studio 5.1 이상에서만 사용할 수 있습니다. 자세한 내용은 사이트 관리자에게 문의하십시오.

탐색기 섹션에는 추가한 파일에 빠르게 액세스하는 데 사용할 수 있는 내 즐겨찾기 폴더가 포함되어 있습니다. SAS 데이터셋, 기타 데이터 파일, 프로그램, 플로우, Snippet, 작업, 질의 등 여러 유형의 파일을 내 즐겨찾기 폴더에 추가할 수 있습니다.

- 내 즐겨찾기 폴더에 파일을 추가하려면 탐색 영역의 해당 섹션에서 파일을 마우스 오른쪽 버튼으로 클릭하고 **내 즐겨찾기에 추가**를 선택합니다.

주: 시작 페이지 탭에서 최근에 연 항목과 즐겨찾는 항목에 액세스할 수 있습니다. 자세한 내용은 ["최근에 연 항목 및 즐겨찾는 항목 액세스" \(19페이지\)](#)에서 확인하십시오.

- 내 즐겨찾기 폴더에서 파일을 제거하려면 탐색 영역의 해당 섹션에서 파일을 마우스 오른쪽 버튼으로 클릭하고 **내 즐겨찾기에서 제거**를 선택합니다. 내 즐겨찾기 폴더에서 파일을 마우스 오른쪽 버튼으로 클릭하고 을 클릭할 수도 있습니다.

탐색기 섹션을 사용하여 파일을 다운로드 및 업로드할 수 있을 뿐만 아니라 폴더와 폴더 바로 가기를 생성할 수도 있습니다. 폴더 트리에서 폴더를 펼치고 접을 수 있고, 파일과 항목을 복사, 이동, 이름을 바꿀 수 있으며, 폴더를 더블 클릭하거나 작업 공간으로 끌어와서 항목을 열 수 있습니다. 또한 SAS Studio 도구 모음에서 **열기**를 클릭하여 폴더 및 폴더 바로 가기에 새 파일을 열 수 있습니다.

주: 스크롤 막대가 있을 때 항목을 **탐색기** 섹션의 다른 위치로 끌어오려면 항목을 오른쪽 위 또는 오른쪽 아래 모서리로 끌어서 위 또는 아래로 스크롤하십시오.

다음과 같은 방법으로 **탐색기** 섹션에 표시된 파일을 변경할 수 있습니다.

- SAS 프로그램 파일만 표시하려면 을 클릭하고 **.sas 파일 표시**를 선택합니다.
- SAS 로그 파일만 표시하려면 을 클릭하고 **.log 파일 표시**를 선택합니다.

파일을 마우스 오른쪽 버튼으로 클릭하고 **파일을 텍스트로 보기**를 선택하여 파일을 볼 수 있습니다.

파일을 SAS 프로그램으로 열려면 파일을 마우스 오른쪽 버튼으로 클릭하고 **프로그램으로 열기**를 선택합니다.

폴더 바로 가기를 생성하려면 다음을 수행합니다.

- 1 탐색 영역에서 을 클릭하여 **탐색기** 섹션을 엽니다. 다음으로 을 클릭하고 **폴더 바로 가기**를 선택합니다.
- 2 **이름** 상자에 폴더 바로 가기의 이름을 입력합니다.
- 3 을 클릭하여 폴더를 선택합니다. 폴더를 생성하려면 을 클릭합니다.
- 4 **확인**을 클릭하여 폴더 바로 가기를 생성합니다. 새로운 바로 가기가 폴더 바로 가기 리스트에 추가됩니다.

주: **탐색기** 섹션에서 폴더 바로 가기를 선택하고 을 클릭하여 폴더 바로 가기의 이름을 바꿀 수 있습니다. **바로 가기 속성** 창에서 **이름** 상자에 새로운 이름을 입력하십시오.

폴더를 생성하려면 **탐색기** 섹션에서 폴더를 생성하려는 폴더를 선택합니다. 을 클릭하고 **폴더**를 새로운 폴더의 이름을 입력합니다. 새로운 폴더가 폴더 리스트에 추가됩니다.

하나 이상의 파일을 다운로드하려면 다운로드할 파일을 마우스 오른쪽 버튼으로 클릭하고 **파일 다운로드**를 선택합니다.

주: SAS Compute Server 폴더에서만 플로우를 다운로드할 수 있습니다. SAS 콘텐츠에서는 플로우를 다운로드할 수 없습니다.

로컬 컴퓨터에서 하나 이상의 파일을 업로드하려면 파일을 업로드하려는 대상 폴더를 선택하고 **↑**을 클릭합니다 **+**을 클릭하여 업로드하려는 파일을 찾습니다. 업로드하는 각 파일의 최대 크기는 100MB를 초과할 수 없습니다.

주: **탐색기** 섹션에서 정렬할 폴더를 마우스 오른쪽 버튼으로 클릭하고 **정렬 기준**을 선택하여 나열된 파일을 정렬할 수 있습니다. 이름, 수정일 또는 크기를 기준으로 파일을 정렬하거나 정렬 순서를 선택하지 않을 수 있습니다. 기본적으로 파일은 파일 이름을 기준으로 하여 오름차순으로 정렬됩니다.

스텝 작업

탐색 영역의 **스텝** 섹션에서는 SAS Studio의 사전 정의된 스텝과 사용자 정의 스텝 인터페이스에 액세스할 수 있습니다. 플로우에 스텝을 추가하여 데이터에 대한 일련의 작업을 빌드할 수 있습니다. 스텝은 플로우에서 노드로 표시됩니다. 플로우는 테이블, 칼럼 및 작업에 대한 참조를 하드코딩하지 않고도 일반화된 데이터 플로우를 만들어 복잡한 프로세스를 관리할 수 있도록 설계되었습니다.

예를 들어, 프로그램에서 입력으로 사용하려는 출력 테이블을 생성하는 질의가 플로우에 포함되어 있음을 알고 있는 경우 질의 노드, 테이블 노드 및 SAS 프로그램 노드를 플로우에 추가하여 시작할 수 있습니다. 플로우의 스텝을 정의한 후 각 노드의 노드 속성을 사용하여 노드의 속성과 콘텐츠를 지정할 수 있습니다.

자세한 내용은 [SAS Studio: 플로우 작업의 “플로우란?”](#)에서 참조하십시오. 메시지가 나타납니다.

주: 탐색 영역의 다른 섹션에서 SAS 프로그램, SAS 데이터셋, Snippet 및 외부 파일과 같은 기존 파일을 끌어와서 플로우에 추가할 수도 있습니다.

작업 사용

탐색 영역의 **작업** 섹션에서는 SAS Studio의 작업에 액세스할 수 있습니다. 작업은 SAS 프 로시저를 기반으로 하며 SAS 코드와 출력형식이 지정된 결과를 자동으로 생성합니다. SAS Studio는 실행할 수 있는 많은 사전 정의된 작업과 함께 제공됩니다. 사전 정의된 대부분의 작업 사본은 편집할 수 있고 고유한 새로운 작업을 생성할 수도 있습니다.

작업을 생성하려면 **[*]**를 클릭하고 **작업**을 선택합니다. 그러면 SAS Studio의 작업 영역에 템플릿이 생성되며 이 템플릿을 사용하여 사용자의 사이트에 맞는 사용자 정의 작업을 생성할 수 있습니다. 사용자 정의 작업은 내 작업 폴더 또는 탐색 영역의 **탐색기** 섹션에서 액세스할 수 있습니다. 자세한 내용은 [“사용자 정의 작업 생성” \(169페이지\)](#)에서 참조하십시오.

생성한 작업을 편집하려면 내 작업 폴더에서 작업을 마우스 오른쪽 버튼으로 클릭하고 **작업 템플릿 편집**을 선택합니다. 그러면 작업을 생성하는 데 사용되는 XML 코드가 작업 영역에 열립니다. 사전 정의된 작업을 편집하려면 먼저 작업을 마우스 오른쪽 버튼으로 클릭하고 **작업**

업 템플릿으로 복사를 선택한 다음 **내 작업** 또는 **탐색기**를 선택합니다. 자세한 내용은 “[사전 정의된 작업 편집](#)” (168페이지)에서 참조하십시오.

Snippet 작업

탐색 영역의 **Snippet** 섹션에서는 저장된 Snippet에 액세스할 수 있습니다. Snippet은 공통으로 사용되는 코드 또는 텍스트 행이며, 저장해서 다시 사용할 수 있습니다. Snippet은 SAS 코드, XML 코드 및 텍스트를 포함할 수 있습니다. 프로그램 편집기, XML 편집기, 텍스트 편집기, 작업 정의, 작업 폼 및 작업 프롬프트를 포함한 위치에 Snippet을 삽입할 수 있습니다. Snippet을 삽입하려면, Snippet을 삽입할 위치를 마우스 오른쪽 버튼으로 클릭하고 **Snippet 삽입**를 선택합니다.

SAS Studio에서는 사전 정의된 코드 Snippet이 제공되므로 이를 사용할 수 있습니다. 이러한 Snippet의 복사본을 편집하여 사용자 본인의 사용자 정의 Snippet을 생성할 수도 있습니다. 사용자 정의 Snippet은 내 Snippet 폴더에서 액세스할 수 있습니다. 자세한 내용은 [2 장, “프로그램 작업”](#) (37페이지)에서 참조하십시오.

생성한 Snippet을 편집하려면 Snippet을 마우스 오른쪽 버튼으로 클릭하고 **편집**를 선택합니다. 사전 정의된 Snippet을 편집하려면 먼저 Snippet을 마우스 오른쪽 버튼으로 클릭하고 **Snippet을 다음으로 복사**를 선택한 다음 **내 Snippet** 또는 **탐색기**를 선택합니다.

.....

주: **내 Snippet** 폴더 또는 **탐색기** 섹션에서 액세스할 수 있는 폴더에 있는 Snippet만 편집할 수 있습니다.

.....

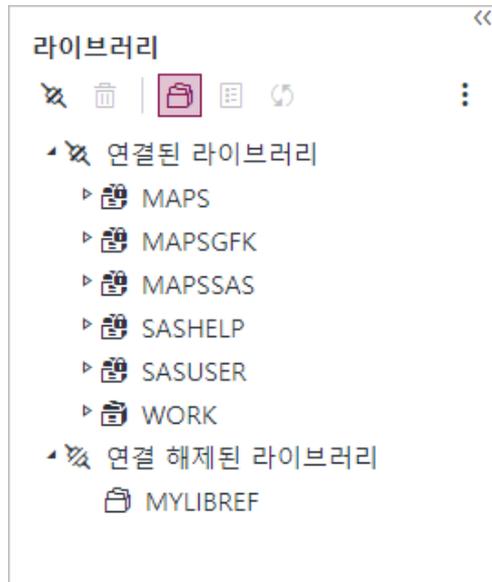
라이브러리 작업

탐색 영역의 **라이브러리** 섹션에서는 SAS 라이브러리에 액세스할 수 있습니다. SAS 테이블은 SAS 라이브러리에 저장됩니다. 라이브러리가 데이터 소스에 연결되었는지 여부에 따라 라이브러리는 연결된 라이브러리 및 연결 해제된 라이브러리 루트 노드로 그룹화됩니다.

.....

주: 연결이 끊긴 라이브러리는 기본적으로 표시되지 않습니다. 연결이 끊긴 라이브러리를 표시하려면 을 클릭합니다.

.....



주: Sasuser 라이브러리는 모든 SAS 서버 환경에서와 같이 읽기 전용입니다. 이 라이브러리에 콘텐츠를 저장할 수 없습니다.

라이브러리 섹션에서 SAS 테이블을 열고 프로그램에 추가할 수 있습니다. 라이브러리 섹션을 사용하여 테이블을 펼치고 해당 테이블 내의 칼럼을 볼 수 있습니다. 칼럼 이름 앞의 아이콘은 칼럼 유형을 나타냅니다.

주: {}(중괄호)가 포함된 테이블 이름은 공백으로 표시됩니다. 테이블 이름에 {}를 사용하지 마십시오.

다음은 일반적인 칼럼 유형 아이콘의 예입니다.

아이콘	칼럼 유형
	문자
	숫자
	날짜
	날짜/시간

라이브러리 섹션에서 다음 액션을 수행할 수 있습니다.

- 프로그램에 테이블이나 칼럼을 추가하려면 라이브러리 섹션에서 프로그램으로 항목을 끕니다. SAS Studio는 프로그램으로 끌어온 항목의 이름을 자동으로 추가합니다. 항목을 끌다 보면 항목을 놓아 프로그램에 삽입할 수 있을 때 커서 모양이 바뀝니다. 항목 이름이 데이터베이스에서 예약어인 경우 SAS Studio는 자동으로 따옴표와 소문자 n을 항목 이름에 추가하여 이름 리터럴을 생성합니다. 자세한 내용은 [“프로그램 열기 및 생성” \(40페이지\)](#)에서 확인하십시오.
- 라이브러리 섹션에서 라이브러리 및 테이블을 모두 새로 고침하려면 > 새로 고침을 선택합니다. 개별 테이블 또는 라이브러리를 새로 고침하려면 테이블 또는 라이브러리를 마우스 오른쪽 버튼으로 클릭하고 새로 고침을 선택합니다.

- 연결된 라이브러리의 속성을 보려면 **라이브러리** 섹션에서 라이브러리를 선택하고 을 클릭합니다.
- **라이브러리** 섹션에서 테이블 칼럼을 정렬하려면 테이블을 마우스 오른쪽 버튼으로 클릭하고 **정렬**을 선택합니다. 칼럼을 오름차순, 내림차순 또는 데이터 순서로 정렬할 수 있습니다. 기본적으로 칼럼은 데이터 순서대로 정렬됩니다. **라이브러리** 섹션에서 칼럼을 정렬해도 테이블 뷰어에 칼럼이 표시되는 방식에는 영향을 미치지 않습니다.
- 라이브러리를 연결하거나 연결 해제하려면 라이브러리를 마우스 오른쪽 버튼으로 클릭하고 라이브러리 **연결** 또는 **라이브러리 연결 해제**를 선택합니다.

.....
 주: 사전 할당된 라이브러리는 연결을 해제할 수 없습니다.

- 연결이 해제된 라이브러리 표시를 켜거나 끄려면 **라이브러리** 섹션 도구 모음에서 을 클릭합니다.
- 생성한 라이브러리를 편집하려면 라이브러리를 마우스 오른쪽 버튼으로 클릭하고 **연결 편집**을 선택합니다.

.....
 주: 사전 할당된 라이브러리는 편집할 수 없습니다.

라이브러리를 생성하려면 다음을 수행합니다.

- 1 탐색 영역의 을 클릭한 다음 을 클릭하여 **라이브러리** 섹션을 엽니다.

새로운 라이브러리 연결

연결 이름: *

My Library

라이브러리 이름(libref): *

MyLibRef

라이브러리 유형:

Cloud Analytic Services engine

시작 시 데이터 소스 할당 및 연결 ⓘ

모든 사용자가 라이브러리 연결을 볼 수 있도록 허용 ⓘ

연결이 해제된 상태로 추가

연결된 상태로 추가

속성 연결 옵션

라이브러리 속성: ⓘ

READONLY

UUID 매크로 변수 이름: ⓘ

저장 및 연결 저장 취소

- 2 **연결 이름** 상자에 라이브러리 연결을 설명하는 이름을 입력합니다.
 - 3 **라이브러리 이름(libref)** 상자에 라이브러리의 라이브러리 참조를 입력합니다.
-
- 주: Caslib 이름은 256자보다 길 수 없습니다. SAS librefs는 8자로 제한됩니다.
-
- 4 **라이브러리 유형** 상자에서 사용할 SAS 라이브러리 엔진을 선택합니다. 사용 가능한 라이브러리 엔진은 배포에 따라 다릅니다.
-
- 주: 지정된 라이브러리 형식을 사용하여 라이브러리를 생성하려면 라이브러리 엔진과 알맞은 클라이언트 소프트웨어가 모두 설치되어 있어야 합니다.
- SAS Base Engine 라이브러리 유형 사용에 대한 내용은 [SAS V9 LIBNAME Engine: Reference](#)에서 참조하십시오.
-
- 5 SAS Studio를 사용할 때마다 이 라이브러리에 액세스하려면 **시작 시 데이터 소스 할당 및 연결**을 선택하여 LIBNAME 문을 autoexec.sas 파일에 추가합니다.

- 6 관리자 권한이 있고 현재 컴퓨팅 컨텍스트의 모든 사용자가 이 라이브러리를 사용할 수 있도록 하려면 **모든 사용자가 라이브러리 연결을 보도록 허용**을 선택합니다. 라이브러리를 연결된 라이브러리인지 또는 연결 해제된 라이브러리인지를 선택할 수 있습니다. 라이브러리는 기본적으로 연결 해제되어 있습니다.

주: 이 옵션은 현재 컴퓨팅 컨텍스트를 업데이트할 권한이 있어야만 사용할 수 있습니다.

- 7 **속성 및 연결 옵션**을 사용하여 필요한 구성 및 연결 옵션을 지정합니다. 특정 옵션에 대한 도움말은 라이브러리 엔진 문서를 참조하십시오.
- 8 새로운 라이브러리에 대한 연결이 올바른지 테스트하려면 **연결 테스트**를 클릭하십시오. 연결에 실패하면 **상세 정보 표시**를 클릭하여 로그 및 오류 메시지를 봅니다.

주: 대화식 영역을 사용 중이면 라이브러리 연결을 테스트할 수 없습니다.

- 9 라이브러리를 저장하고 연결하려면 **저장 및 연결**을 클릭합니다. 라이브러리를 연결하지 않고 저장하려면 **저장**을 클릭합니다. 탐색 영역의 라이브러리 리스트에 새로운 라이브러리가 추가됩니다.

팁 검색 영역의 **라이브러리** 섹션에서 라이브러리를 마우스 오른쪽 버튼으로 클릭하고 **연결 편집**을 선택하여 생성한 라이브러리를 편집할 수 있습니다.

파일 참조 사용

파일 참조를 사용하면 지정한 파일에 빠르게 액세스할 수 있습니다. SAS 파일 시스템의 파일, SAS 콘텐츠 폴더의 파일 또는 URL을 통해 액세스하는 파일에 대한 파일 참조를 생성할 수 있습니다.

주: FILENAME 문을 사용하여 SAS 서버에서 액세스할 수 있는 원격 위치에 있는 파일에 대한 파일 참조를 생성할 수도 있습니다.

SAS 파일 시스템의 파일, SAS 콘텐츠 폴더의 파일 또는 URL을 통해 액세스하는 파일에 대한 파일 참조를 생성하려면 다음을 수행합니다.

- 1 탐색 영역의 을 클릭하고 을 클릭합니다.

새로운 파일 참조 ✕

이 세션에 대한 파일 참조를 생성하려면 다음 값을 지정하십시오.

이름:

액세스 방법: ⓘ

위치:

시작 시 이 파일 참조 재생성 ⓘ

주: 탐색 영역에서 파일 참조 버튼을 사용할 수 없으면 보기 ⇨ 탐색 영역 ⇨ 파일 참조를 선택합니다.

- 2 **이름** 상자에 파일 참조 이름을 입력합니다.
- 3 **액세스 방법** 상자를 사용하여 파일에 액세스하는 방법을 지정합니다.
 - **DISK** - SAS 파일 시스템의 파일에 액세스합니다. **위치** 상자에서 사용할 파일을 찾아 지정합니다.
 - **FILESRV** - SAS 콘텐츠 폴더의 파일에 액세스합니다. **위치** 상자에서 사용할 파일을 찾아 지정합니다.
 - **URL** - 지정된 URL을 사용하여 파일에 액세스합니다. **URL** 상자에 파일의 URL을 입력합니다.
- 4 다음에 SAS Studio를 사용할 때 이 참조를 사용할 수 있도록 하려면 **시작 시 이 파일 참조 재생성**을 선택합니다.
- 5 **확인**을 클릭하여 파일 참조를 생성합니다. 탐색 영역의 파일 참조 리스트에 새로운 파일 참조가 추가됩니다.

또한 SAS 파일 시스템(DISK)의 파일, SAS 콘텐츠 폴더(FILESRV)의 파일 또는 URL(URL)을 통해 프로그래밍 방식으로 액세스하는 파일에 대한 파일 참조를 생성할 수도 있습니다. 파일 참조 영역에 파일 참조가 표시됩니다. 파일 참조를 더블 클릭하거나 작업 영역으로 끌어오는 방법으로 파일 참조에서 파일 또는 URL을 열 수 있습니다.

다음은 몇 가지 샘플 코드입니다.

```
FILENAME MyProg FILESRVC folderpath='/Users/userid/' filename='MyProgram.sas';
```

```
FILENAME MyURL URL 'http://www.sas.com';
FILENAME SASPROG '/sastest/General_Test/MyProgram.sas';
```

Git 레파지토리 작업

탐색 영역의 **Git 레파지토리** 섹션에서 SAS Studio 내의 Git 기능에 액세스할 수 있습니다. 여기에는 SAS Studio 내에서 레파지토리 복제(clone), 파일 변경 커밋(commit) 및 스택시(stash), 파일 풀(pull) 및 푸시(push), 레파지토리 히스토리 보기, 브랜치 생성 및 병합(creat & merge), 브랜치 리베이스(rebase), 로컬 레파지토리에서 파일 간 기본 diff 수행 등이 포함됩니다. 자세한 내용은 ["SAS Studio의 Git 통합 정보" \(134페이지\)](#)에서 참조하십시오.

탐색 영역 사용자 정의

표시되는 탐색 영역의 섹션을 사용자 정의하려면 **보기**를 클릭하고 **탐색 영역**을 선택합니다. 그런 다음 추가 또는 제거할 섹션을 선택 또는 선택 취소합니다. 탐색 영역은 즉시 업데이트됩니다.

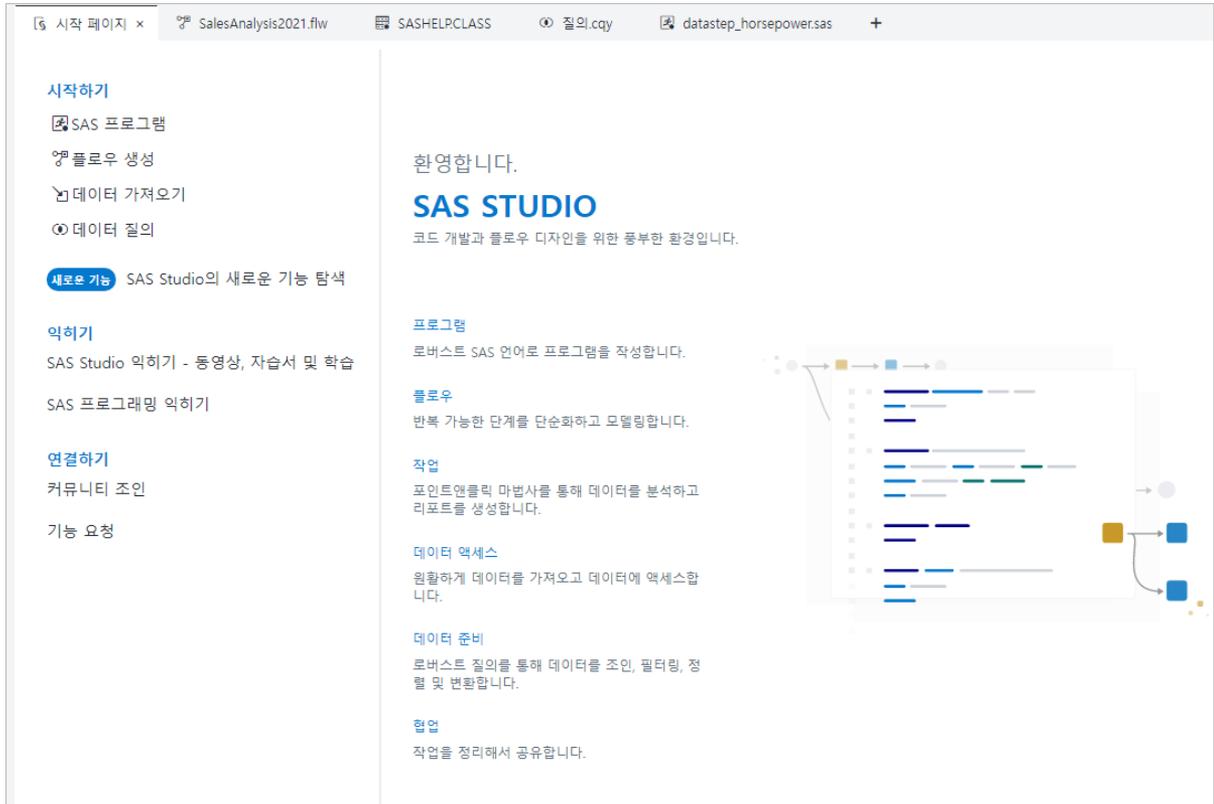
.....
주: 파일 참조 섹션은 기본적으로 표시되지 않습니다.

작업 영역 사용

작업 영역 사용 정보

작업 영역은 프로그램, 작업 및 플로우에 액세스하고 데이터를 볼 수 있는 SAS Studio 응용 프로그램의 기본 부분입니다. 작업 영역은 항상 표시되며 닫을 수 없습니다. 프로그램, 작업, 테이블 또는 플로우를 열면 작업 영역에서 창이 새로운 탭으로 열립니다. 프로그램 및 작업과 관련된 코드, 로그 및 결과는 해당 프로그램 또는 작업의 기본 탭 아래에 함께 표시됩니다.

SAS Studio를 처음 열면 기본적으로 작업 영역에 **시작 페이지** 탭이 표시됩니다. 시작 페이지에서는 몇 가지 기본 작업(예: 프로그램 또는 질의 생성)에 빠르게 액세스할 수 있고 학습 리소스에 대한 링크도 제공합니다. **시작 페이지** 탭을 닫았으면 **보기** ⇨ **시작 페이지**를 선택하여 다시 열 수 있습니다. **시작 페이지 표시** 옵션을 사용하여 SAS Studio를 시작할 때마다 **시작 페이지**를 표시할지를 제어합니다. 자세한 내용은 ["시작 기본 설정 지정" \(174페이지\)](#)에서 참조하십시오.



작업 영역에서 탭 생성

최근에 연 탭 옆에 있는 **+**을 클릭하고 열리는 탭 유형을 선택하여 작업 영역에서 새로운 탭을 생성할 수 있습니다.

The screenshot shows the SAS Studio interface with a data table. The table has columns: Name, Sex, Age, and Weight. The data rows are as follows:

	Name	Sex	Age	Weight
1	Alfred	M	14	112.5
2	Alice	F	13	84
3	Barbara	F	13	98
4	Carol	F	14	102.5
5	Henry	M	14	102.5
6	James	M	12	83
7	Jane	F	12	84.5
8	Janet	F	15	112.5
9	Jeffrey	M	13	84
10	John	M	12	99.5
11	Joyce	F	11	50.5
12	Judy	F	14	90
13	Louise	F	12	77
14	Mary	F	15	112
15	Philip	M	16	150
16	Robert	M	12	128

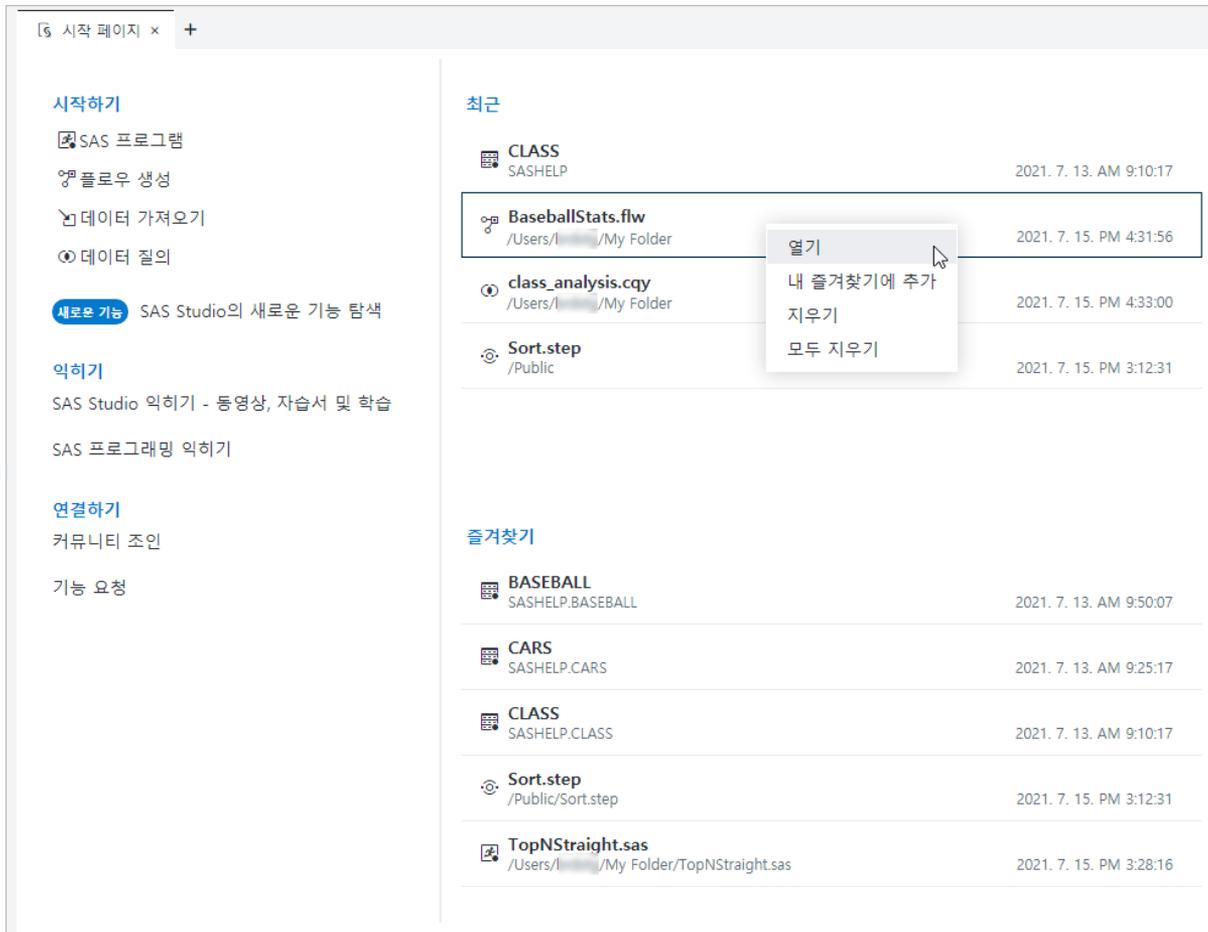
최근에 연 항목 및 즐겨찾는 항목 액세스

SAS Studio는 최근에 열고 저장한 항목과 즐겨찾기로 표시한 항목을 추적합니다. 최근에 작업 영역에서 항목을 열거나 저장했거나 즐겨찾기 폴더에 일부 항목을 추가한 경우 시작 페이지의 도입 그래픽이 최근 항목, 저장 항목 및 즐겨찾기 항목 리스트로 바뀝니다.

주: 내 즐겨찾기 폴더에 항목을 추가하는 방법에 대한 자세한 내용은 ["탐색기 사용"](#) (8페이지)에서 참조하십시오.

최근 및 즐겨찾기 리스트를 사용하여 다음 작업을 수행할 수 있습니다.

- 항목을 더블 클릭하거나 항목을 마우스 오른쪽 버튼으로 클릭하고 **열기**를 선택하여 리스트에서 항목을 열 수 있습니다.
- 항목을 마우스 오른쪽 버튼으로 클릭하고 **내 즐겨찾기에 추가**를 선택하여 최근 리스트의 항목을 내 즐겨찾기 폴더에 추가할 수 있습니다.
- 항목을 마우스 오른쪽 버튼으로 클릭하고 **지우기** 또는 **모두 지우기**를 선택하여 최근 항목 리스트에서 하나 또는 모든 항목을 지울 수 있습니다.
- 즐겨찾기 리스트에서 항목을 마우스 오른쪽 버튼으로 클릭하고 **내 즐겨찾기에서 제거**를 선택하여 내 즐겨찾기 폴더에서 항목을 제거할 수 있습니다.



주: 최근 리스트는 영역 간에 공유되지 않습니다. 각 영역에는 별도의 최근 리스트가 있습니다. 자세한 내용은 “영역 이해” (31페이지)에서 확인하십시오.

작업 영역 사용자 정의

기본적으로 작업 영역은 탐색 영역 옆에 표시되지만 작업 영역을 최대화하고 탐색 영역을 숨길 수 있습니다. 작업 영역의 모든 탭을 한 번에 닫을 수도 있습니다.

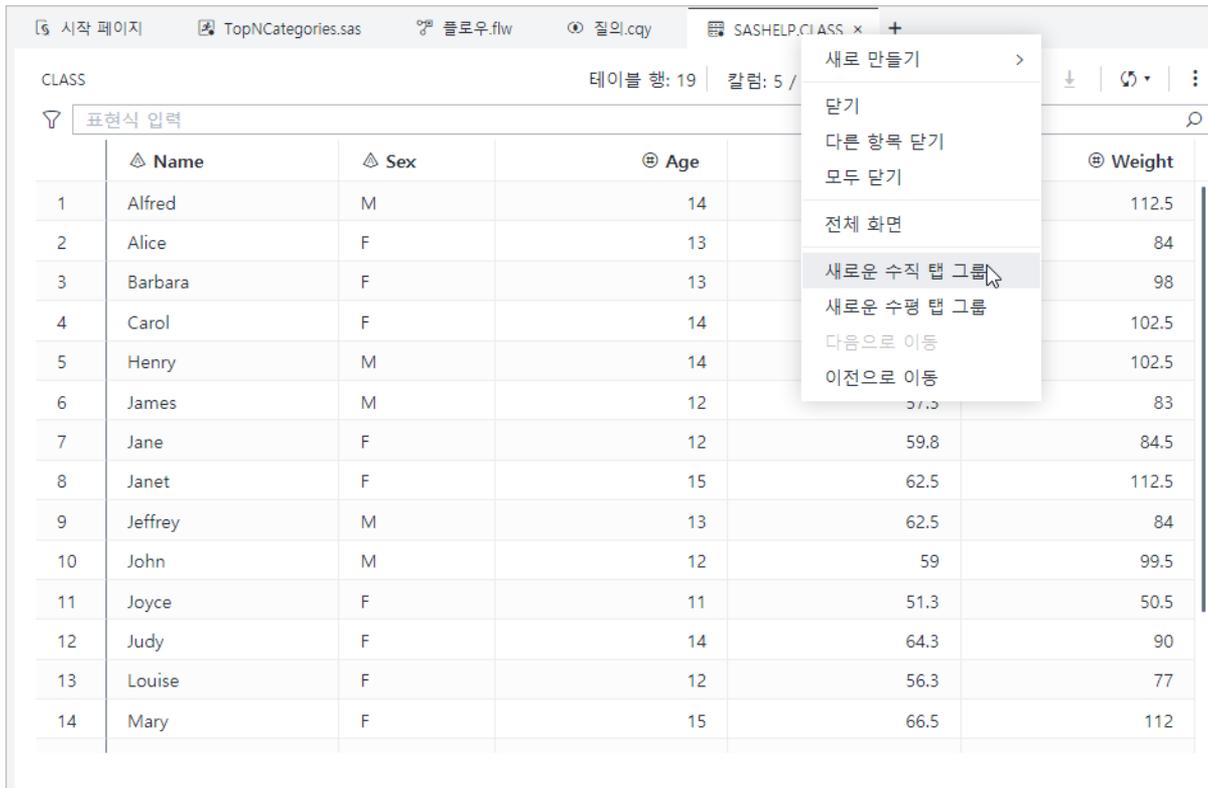
작업 영역을 최대화하려면 **보기** ⇨ **전체 화면**을 선택합니다.

주: 탐색 영역을 다시 열려면 탭 오른쪽에 있는 **복원**을 클릭합니다. 작업 영역에서 탭을 더블 클릭하거나 탭을 마우스 오른쪽 버튼으로 클릭해도 **복원**을 선택할 수 있습니다.

작업 영역에 열려 있는 탭 중 일부 또는 모두를 닫을 수도 있습니다. 작업 영역에 열려 있는 모든 탭을 닫으려면 임의 탭을 마우스 오른쪽 버튼으로 클릭하고 **모두 닫기**를 선택합니다. 활성 탭을 제외한 모든 탭을 닫으려면 활성 탭을 마우스 오른쪽 버튼으로 클릭하고 **다른 항목 닫기**를 선택합니다. 저장되지 않은 프로그램 또는 작업이 있으면 저장할지 묻는 메시지가 표시됩니다.

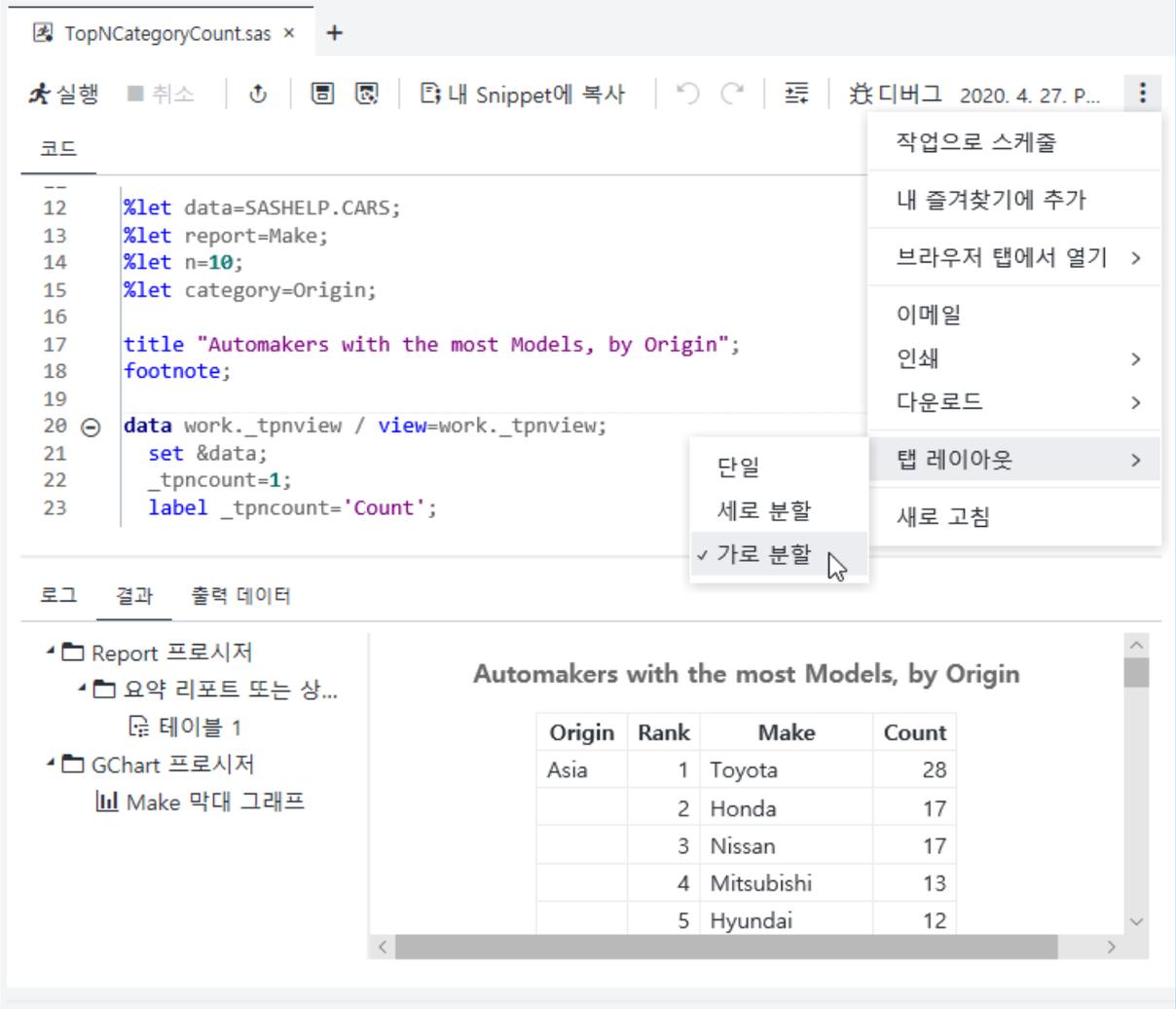
작업 영역에서 탭 재정렬

작업 영역에서 탭을 왼쪽 또는 오른쪽으로 끌어서 놓는 방법으로 탭을 재정렬할 수 있습니다. 별도의 수직 및 수평 그룹을 생성하고 그룹 간에 탭을 이동할 수도 있습니다. 탭을 재정렬하려면 이동할 탭을 마우스 오른쪽 버튼으로 클릭하고 적절한 옵션을 선택합니다.



표준 영역에서 **로그**, **결과** 및 **출력 데이터** 하위 탭을 **코드** 하위 탭이 세로 또는 가로로 분할되어 있는 단일 그룹에 표시하여 프로그램 탭의 레이아웃을 변경할 수 있습니다. 레이아웃을 변경하려면 **과 탭 레이아웃**을 차례로 클릭합니다. 그런 다음 적절한 옵션을 선택합니다.

대화식 영역에서 **로그**, **결과** 및 **출력 데이터** 하위 탭을 별도의 세로 또는 가로 탭 그룹으로 표시할 수 있습니다. 레이아웃을 변경하려면 **을**을 클릭하고 알맞은 옵션을 선택합니다.



TopNCategoryCount.sas × +

실행 취소 | 내 Snippet에 복사 | 실행 | 디버그 2020. 4. 27. P...

코드

```
--
12 %let data=SASHELP.CARS;
13 %let report=Make;
14 %let n=10;
15 %let category=Origin;
16
17 title "Automakers with the most Models, by Origin";
18 footnote;
19
20 data work._tpnview / view=work._tpnview;
21 set &data;
22 _tpncount=1;
23 label _tpncount='Count';
```

로그 결과 출력 데이터

Report 프로시저

- 요약 리포트 또는 상...
 - 테이블 1
- GChart 프로시저
 - Make 막대 그래프

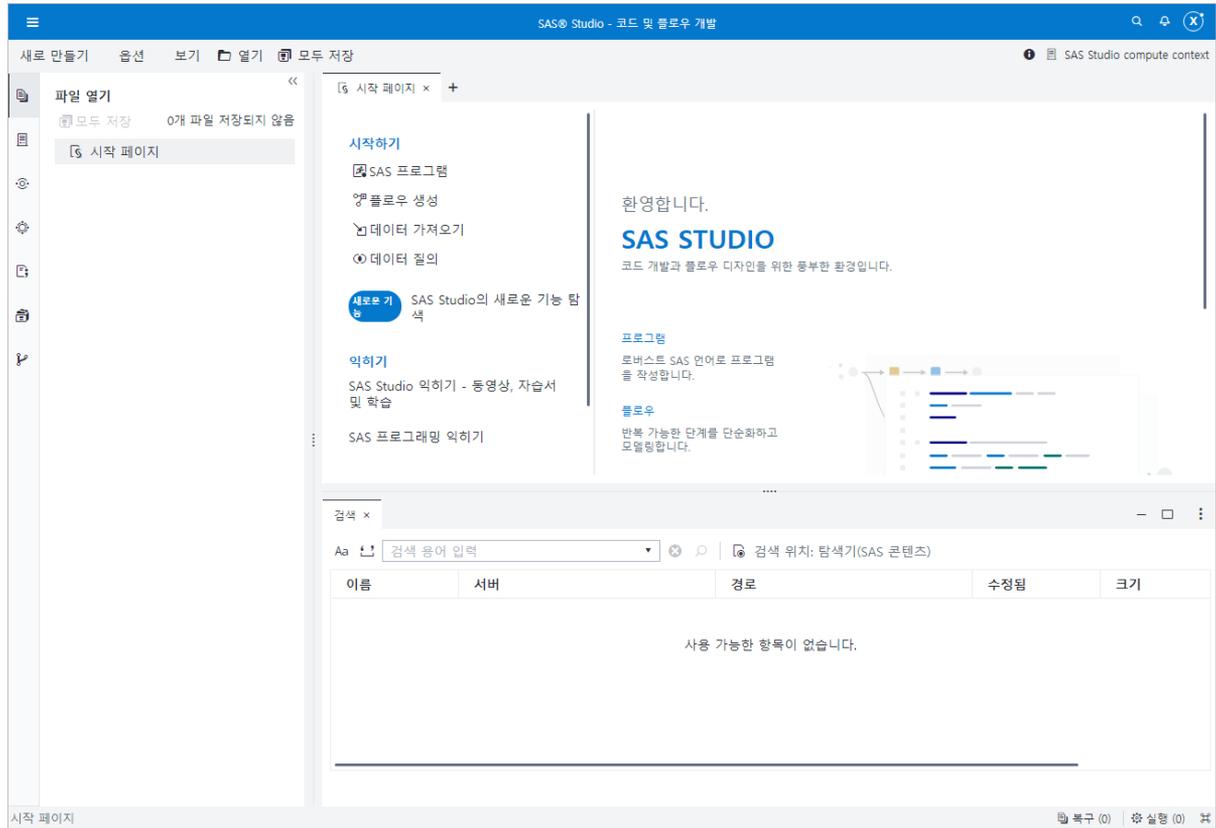
Automakers with the most Models, by Origin

Origin	Rank	Make	Count
Asia	1	Toyota	28
	2	Honda	17
	3	Nissan	17
	4	Mitsubishi	13
	5	Hyundai	12

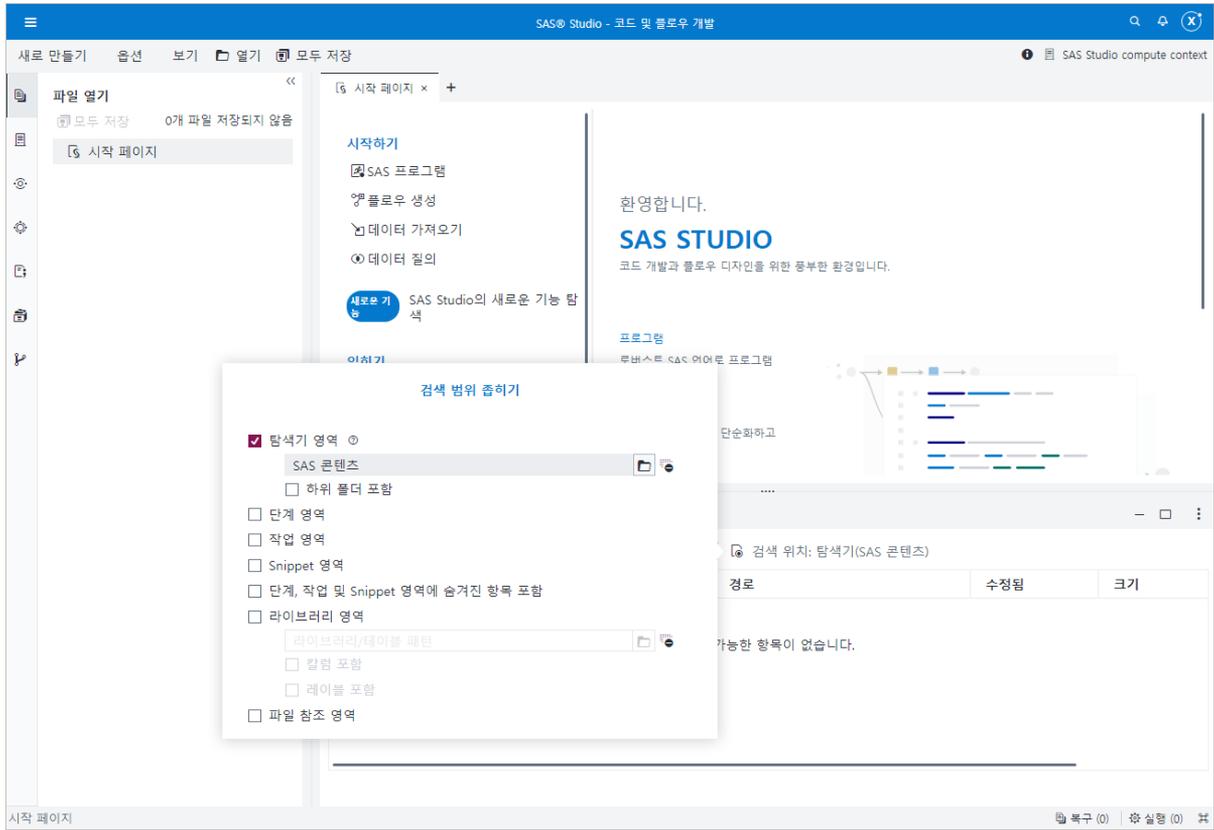
SAS Studio에서 검색하기

검색 기능을 사용하여 SAS 콘텐츠 및 SAS 서버 폴더를 포함하여 탐색 창의 모든 섹션을 검색할 수 있습니다. 검색에 포함할 섹션과 항목 유형을 지정할 수 있습니다. 검색 시 대소문자를 구분할지 여부와 전체 단어가 일치하는 항목을 검색할지 여부를 지정할 수도 있습니다.

검색 기능에 액세스하려면 [보기](#) ⇒ [검색](#)을 선택합니다.



항목을 검색하려면 검색 용어 상자에 검색할 텍스트를 입력합니다. 탐색 영역에서 검색할 영역을 지정하려면  을 클릭합니다. 기본적으로 SAS Studio는 탐색 영역의 **탐색기** 섹션에서 SAS 콘텐츠 사용자 디렉터리를 검색합니다.



검색 시 대소문자를 구분하려면 **Aa**를 클릭합니다. 검색 용어와 전체 단어가 일치하는 항목을 검색하려면 **[Enter]**을 클릭합니다.

검색 기준을 지정했으면 **O**를 클릭합니다. 검색 결과에서 항목을 더블 클릭하여 열 수 있습니다.

이름	서버	경로	수정됨	크기	유형
Age		SASHELP.BMIMEN	2022. 4. 8. 오후 7:17:16		칼럼
Age		SASHELP.CLASS	2022. 4. 8. 오후 6:44:59		칼럼
Age		SASHELP.CLASSFIT	2022. 4. 19. 오전 9:09:38		칼럼

콘솔 사용

콘솔 창에는 프로그램 및 작업을 실행할 때 생성된 오류 메시지와 경고 같은 모든 메시지의 리스트가 표시됩니다. 콘솔을 열려면 **보기** ⇒ **콘솔**을 선택합니다. 콘솔 탭을 최소화했으면, 작업 영역 오른쪽 하단에 있는 **[maximize icon]**을 클릭하여 콘솔을 복원하거나 **[close icon]**를 클릭하여 콘솔을 최대화할 수 있습니다.

창 맨 위의 버튼을 사용하여 표시되는 메시지 유형을 필터링할 수 있습니다. 기본적으로 모든 유형의 메시지가 표시됩니다. SAS Studio에서 로그아웃하면 메시지가 기본적으로 지워집니다. **모두 지우기**를 클릭하여 콘솔에서 메시지를 수동으로 지울 수도 있습니다.

The screenshot shows the SAS Studio interface. At the top, there are tabs for '시작 페이지', 'datastep_horsepower.sas', and '* 데이터 리스트.ctlk'. Below the tabs, there are buttons for '실행', '취소', and '내 작업에 복사'. The main area is divided into three panes: '데이터' (Data), '역할' (Roles), and '결과' (Results). The '데이터' pane shows 'SASHELP.BASEBALL' selected. The '역할' pane shows a list of variables: Name, Team, nAtBat, and nHits. The '결과' pane shows a table titled 'SASHELP.BASEBALL에 대한 데이터 리스트' with columns 'OBS', 'Player's Name', and 'Team at the'. The table contains 9 rows of data. Below the main panes, there is a '작업 콘솔(0)' window. At the bottom, there is a '콘솔' window with a '모두 지우기' button and a list of messages.

OBS	Player's Name	Team at the
1	Allanson, Andy	Cleveland
2	Ashby, Alan	Houston
3	Davis, Alan	Seattle
4	Dawson, Andre	Montreal
5	Galarraga, Andres	Montreal
6	Griffin, Alfredo	Oakland
7	Newman, Al	Montreal
8	Salazar, Argenis	Kansas City
9	Thomas, Andres	Atlanta

콘솔 메시지:

- ① 프로그램.sas이(가) 실행되었습니다.
- ① 프로그램.sas을(를) 완료했습니다.
- ① 프로그램.sas 경과 시간 5.488초
- ① classfit.sas이(가) 실행되었습니다.
- ① classfit.sas을(를) 완료했습니다.
- ① classfit.sas 경과 시간 2.789초
- ① SQL_reports.sas이(가) 실행되었습니다.

백그라운드 실행 기능 사용

백그라운드 실행 기능 정보

저장된 프로그램, 질의, 작업 또는 플로우를 백그라운드로 실행할 수 있습니다. 즉, SAS Studio를 계속 사용하는 동안에는 파일을 실행할 수 있습니다. 백그라운드에서 실행된 파일의 상태를 볼 수 있으며 현재 백그라운드에서 실행 중인 파일을 취소할 수 있습니다.

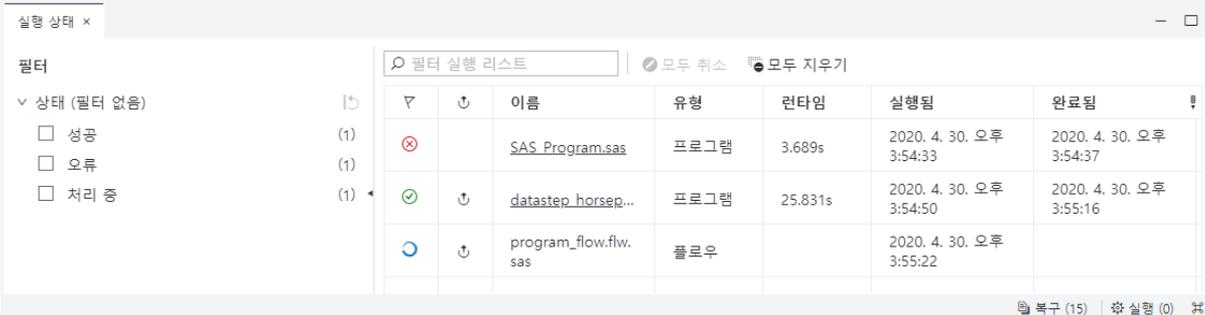
백그라운드 실행으로 파일 실행

백그라운드 실행으로 파일을 실행하려면 탐색 영역에서 프로그램을 마우스 오른쪽 버튼으로 클릭하고 **백그라운드 실행**을 선택하거나 도구 모음에서 을 클릭합니다. 파일을 실행하기 전에, 백그라운드 프로세스는 현재 작업 중인 디렉터리를 프로그램이 있는 디렉터리로 변경합니다.

기본적으로 알림 메시지는 파일이 실행될 때와 파일 실행이 끝날 때를 표시됩니다. 파일이 실행되는 동안에는 SAS Studio에서 로그아웃하더라도 파일이 계속 실행되지만 파일이 완료되는 시기를 나타내는 알림 메시지는 표시되지 않습니다.

주: 백그라운드 실행은 별도의 컴퓨팅 서버를 사용하기 때문에 실행에서 생성한 라이브러리 또는 테이블은 SAS Studio 탐색 영역의 **라이브러리** 섹션에 나타납니다.

백그라운드 실행 상태 및 속성을 보려면 **보기**를 클릭하고 **실행 상태**를 선택합니다. 백그라운드 실행은 으로 표시됩니다. 자세한 내용은 **“실행 상태 창 사용” (26페이지)**에서 확인하십시오.



실행 상태 창은 필터, 실행 목록, 그리고 실행 상태 선택 옵션을 포함합니다. 실행 목록은 다음과 같습니다:

필터	이름	유형	런타임	실행됨	완료됨
<input type="checkbox"/> 성공 <input type="checkbox"/> 오류 <input type="checkbox"/> 처리 중	SAS_Program.sas	프로그램	3.689s	2020. 4. 30. 오후 3:54:33	2020. 4. 30. 오후 3:54:37
	datastep_horsep...	프로그램	25.831s	2020. 4. 30. 오후 3:54:50	2020. 4. 30. 오후 3:55:16
	program_flow.flw.sas	플로우		2020. 4. 30. 오후 3:55:22	

백그라운드 실행 사용자 정의

우선 설정 창에서 백그라운드 실행 처리 방법을 사용자 정의할 수 있습니다.

백그라운드 실행을 다시 수행할 때 기존 로그 및 출력 파일을 삭제할지 또는 덮어쓸지를 변경하려면 **옵션** ⇨ **우선 설정**을 선택합니다. **백그라운드 실행**을 클릭합니다.

각 옵션에 대한 자세한 내용은 **“백그라운드 실행 우선 설정 지정” (184페이지)**에서 참조하십시오.

실행 상태 창 사용

실행 상태 창에는 SAS Studio에서 실행하는 프로그램, 작업, 질의 및 플로우에 대한 정보가 표시됩니다. **실행 상태** 창을 열려면 **보기** ⇨ **실행 상태**를 선택하거나 SAS Studio 브라우저 창의 오른쪽 하단에 있는 **실행**을 클릭합니다.

현재 진행 중인 실행을 취소하려면 항목을 마우스 오른쪽 버튼으로 클릭하고 **취소**를 선택합니다. 현재 실행 중인 모든 프로그램, 작업, 질의 및 플로우를 취소하려면 실행 상태 창 도구 모음에서 **모두 취소**를 클릭합니다.

주: 백그라운드 실행은 취소할 수 없습니다.

실행 상태 창에는 프로그램, 작업, 질의 또는 플로우를 실행할 때마다 엔트리가 포함됩니다. 실행 상태 창을 사용하여 이전 버전의 프로그램, 작업, 질의 또는 플로우로 돌아갈 수 있습니다. 실행 상태 창에서 항목을 클릭할 때 해당 시간에 실행되는 항목 버전과 관련 로그가 열립니다. 프로그램 또는 자동으로 생성된 작업, 질의 또는 플로우는 변경 사항을 적용하고 저장한 새로운 탭에서 열립니다.

주: 실행 상태 창에서 연 코드는 SAS Studio가 실행한 코드의 본사본으로 원래 프로그램, 작업, 질의 또는 플로우와 더 이상 연결되지 않습니다. 이 코드를 편집해도 원래 프로그램, 작업, 질의 또는 플로우에 영향을 주지 않습니다.

필터	이름	유형	런타임	실행됨	완료됨	서버
<input type="checkbox"/> 성공 <input type="checkbox"/> 오류 <input type="checkbox"/> 취소됨	Sales_Study.flw.sas	플로우	12.601s	2020. 4. 30. 오후 3:02:03	2020. 4. 30. 오후 3:02:16	SAS Studio compute context
	class_analysis.sas	프로그램	46.977s	2020. 4. 30. 오후 3:02:54	2020. 4. 30. 오후 3:03:41	SAS Studio compute context
	SummaryStatistic...	작업	5.083s	2020. 4. 30. 오후 3:03:45	2020. 4. 30. 오후 3:03:50	SAS Studio compute context
	Longprogram.sas	프로그램	5.252s	2020. 4. 30. 오후 3:04:13	2020. 4. 30. 오후 3:04:18	SAS Studio compute context

여러 방법으로 실행 상태 창에 표시된 항목을 사용자 정의할 수 있습니다.

- **필터** 영역을 사용하여 실행 상태로 표시된 항목을 서브셋합니다.
- **필터 실행 리스트** 상자를 사용하여 **이름** 칼럼에 대한 필터 기준을 지정합니다.
- 실행 리스트에서 개별 항목을 제거하려면 항목을 오른쪽 버튼으로 클릭하고 **지우기**를 선택합니다.
- 실행 리스트에서 모든 항목을 제거하려면 **실행 상태 창 도구 모음**에서 **모두 지우기**를 클릭합니다. 실행 프로세스에 있는 항목은 리스트에서 지워지지 않습니다.

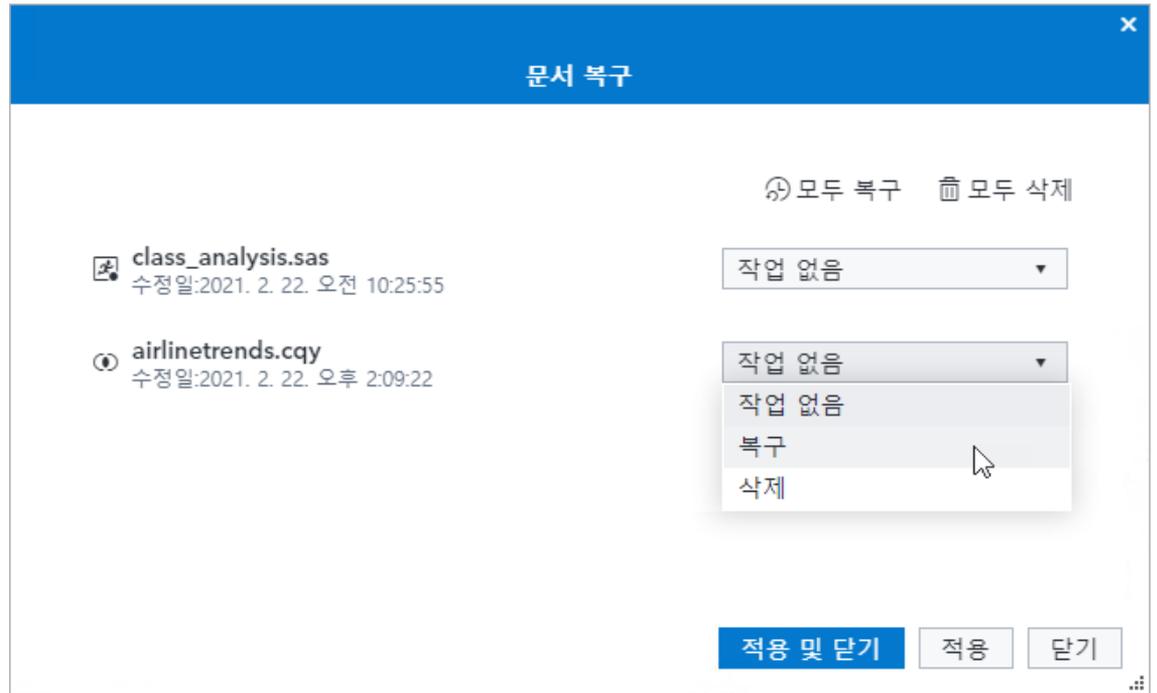
주: SAS Studio 세션이 종료되면 백그라운드 실행을 제외한 실행 상태 창의 모든 항목이 제거됩니다. 실행 상태 창에서 제거하려면 백그라운드 실행 항목을 수동으로 지워야 합니다.

문서 복구 창 사용

문서 복구 창에서는 SAS Studio가 자동으로 저장한 파일을 복구하거나 삭제할 수 있습니다. 기본적으로 부재 중 SAS Studio가 시간 초과되거나 정전과 같은 상황으로 인해 예기치 않게 종료되는 경우 SAS 프로그램, 플로우, 질의 및 작업 등 열려 있는 파일의 가장 최근에 저장된 버전을 복구할 수 있습니다.

문서 복구 창을 열려면 **보기** ⇨ **문서 복구**를 선택하거나 SAS Studio 브라우저 창의 오른쪽 하단 모서리에서 **복구**를 클릭합니다.

주: **문서 복구** 창은 문서를 복구한 경우에만 사용할 수 있습니다. **문서 복구 사용** 옵션을 사용하여 문서 복구 기능을 관리할 수 있습니다. 자세한 내용은 **"일반 우선 설정 지정" (175페이지)**에서 참조하십시오.



문서 복구 창의 각 파일에 대해 다음 옵션 중에서 선택할 수 있습니다.

- **작업 없음** - 복구된 문서 리스트의 파일을 유지합니다. 파일을 복구하거나 삭제할 때까지 문서 복구 창에서 파일을 계속 사용할 수 있습니다.
- **복구** - 작업 영역의 새로운 탭에서 파일을 엽니다. 복구된 문서 리스트에서 파일이 삭제됩니다.
- **삭제** - 복구된 문서 리스트에서 파일을 삭제합니다.

주: **모두 복구** 또는 **모두 삭제**를 클릭하여 모든 문서를 복구하거나 삭제할 수 있습니다.

복구된 파일에 대한 작업을 지정한 경우 다음 옵션 중 하나를 선택합니다.

- **적용 및 닫기** - 선택한 작업을 파일에 적용하고 **문서 복구** 창을 닫습니다.
- **적용** - 선택한 작업을 파일에 적용하지만 **문서 복구** 창은 닫지 않습니다.
- **닫기** - 파일에 작업을 적용하지 않고 **문서 복구** 창을 닫습니다. 다음에 **문서 복구** 창을 열 때 파일을 계속 사용할 수 있습니다.

프로그램, 작업, 질의 또는 플로우를 작업으로 스케줄링

지정된 시간과 빈도로 실행되도록 프로그램, 작업, 질의 및 플로우를 포함한 저장된 항목을 스케줄링할 수 있습니다. SAS Studio를 사용하지 않더라도 스케줄 항목은 계속 실행됩니다.

프로그램, 작업, 질의 또는 플로우 스케줄링

지정된 시간에 항목이 실행되도록 스케줄링하려면 다음을 수행합니다.

- 1 탐색 영역의 **탐색기** 섹션에서 항목을 마우스 오른쪽 버튼으로 클릭하고 **작업으로 스케줄**을 선택합니다.

주: 항목 도구 모음에서 을 클릭하고 **작업으로 스케줄**을 선택해서도 항목을 스케줄링할 수 있습니다.

- 2 **새로운 트리거** 창에서 작업 실행 빈도를 지정합니다. 작업 실행에 대해 실행 시간, 시작일과 종료일을 지정할 수도 있습니다.
- 3 **저장**을 클릭합니다. 스케줄 작업 창을 사용하여 스케줄 작업을 보고 관리할 수 있습니다.

스케줄 작업 창 사용

스케줄 작업 창에는 SAS Studio의 모든 스케줄 작업에 대한 정보가 표시됩니다. 저장된 프로그램, 작업, 질의 및 플로우를 스케줄링하고 SAS Viya 플랫폼 작업을 생성 및 스케줄링할 수도 있습니다. SAS Viya 플랫폼 작업은 프로그램과 그 정의로 구성되며 작업에 사용자 인터페이스를 제공하는 HTML 폼 또는 작업 프롬프트를 포함할 수 있습니다.

스케줄 작업 창을 열려면 **보기** ⇨ **스케줄 작업**을 선택합니다. SAS Viya 플랫폼 작업에 대한 자세한 내용은 [SAS Studio 개발자 가이드: 작업하기의 "작업 관리"](#)에서 참조하십시오.

스케줄 작업 창에는 다음번 작업 실행 스케줄과 작업의 최신 제출 및 완료 시간을 포함하여 스케줄링한 각 작업에 대한 정보가 있는 엔트리가 포함됩니다.



스케줄 작업이 다음과 같은 방식으로 실행되는 시기를 변경할 수 있습니다.

- 스케줄 작업을 즉시 실행하려면 작업을 선택하고 **지금 실행**을 클릭합니다. 작업이 별도의 브라우저 창에서 열립니다.
- 스케줄 작업을 비활성화하려면 해당 작업의 행에서 **🔴**을 클릭합니다. 작업을 다시 활성화하려면 **🟢**을 클릭합니다.
- 작업 스케줄 시간을 변경하려면 **🕒** 버튼을 클릭합니다. 작업 실행 시기를 변경할 수 있도록 트리거 편집 창이 열립니다.
- 스케줄에서 작업을 제거하려면 작업을 선택하고 **🗑️**를 클릭합니다. 작업은 스케줄 작업 창에 남아 있지만 더 이상 실행이 스케줄링되지 않습니다. 작업을 다시 스케줄링하려면 작업을 마우스 오른쪽 버튼으로 클릭하고 **스케줄**을 선택합니다.
- 작업을 삭제하려면 작업을 마우스 오른쪽 버튼으로 클릭하고 **작업 삭제**를 선택합니다.

제출된 모든 작업 버전 리스트를 보려면 **🕒**을 클릭합니다. 리스트에서 해당 버전을 클릭하여 작업과 관련된 코드, 로그 및 결과를 열 수 있습니다. 특정 버전의 작업을 삭제하려면 알맞은 버전 옆에 있는 **🗑️**을 클릭합니다.

주: 스케줄 작업 창에서 연 코드는 SAS Studio가 실행한 코드의 복사본으로 원래 프로그램, 작업, 질의 또는 플로우와 더 이상 연결되지 않습니다. 이 코드를 편집해도 원래 프로그램, 작업, 질의 또는 플로우에 영향을 주지 않습니다.

여러 가지 방법으로 스케줄 작업 창을 사용자 정의할 수 있습니다.

- **스케줄 작업 필터링** 상자를 사용하여 이름 칼럼에 대한 필터 기준을 지정합니다.
- **시간별 필터링** 드롭다운 리스트를 사용하여 보려는 특정 기간을 선택합니다.
- 선택한 유형의 실행 상태를 모두 볼지 아니면 선택한 유형만 볼지를 지정합니다.
- **🔍**을 클릭하여 칼럼 관리 창을 열고 창에 표시할 칼럼을 지정합니다.

주: SAS Studio 스케줄링 기능은 단일 사용자를 위한 기본적인 스케줄링 기능을 제공합니다. 관리자는 SAS Environment Manager를 사용하여 고급 스케줄링 옵션에 액세스할 수 있습니다. SAS Environment Manager를 통해 관리자는 여러 사용자의 작업을 보고 관리할 수 있습니다. 자세한 내용은 [SAS Environment Manager: User's Guide](#)의 "Jobs and Flows Page"에서 확인하십시오.

영역 이해

SAS Studio는 조직 내의 다양한 사용자 및 그룹이 사용할 수 있으므로 사용자의 필요에 가장 적합한 특정 기능 또는 영역만 표시되도록 선택할 수 있습니다. 영역은 다양한 사용자 유형의 필요에 맞게 사용자 정의된 기능 집합입니다. SAS Studio에는 표준 영역과 대화식 영역의 두 가지 영역이 포함되어 있습니다.

표준 영역은 SAS 프로그램을 작성하고 사전 정의된 작업을 사용하며 플로우를 생성하는 사용자를 위한 영역입니다. 표준 영역에서는 SAS 프로그램 파일이 모두 한꺼번에 실행됩니다. 기본적으로 표준 영역은 **시작 페이지** 탭에서 열리며 여기에는 파일 열기, 탐색기, 스텝, 작업, Snippet, 라이브러리 및 Git 섹션이 포함됩니다.

대화식 영역은 주로 대화식 SAS 프로시저를 사용하는 SAS 프로그램의 작성 및 편집 작업에 SAS Studio를 사용하는 사용자를 위한 영역입니다. 기본적으로 대화식 영역은 **시작 페이지** 탭에서 열리며 **로그** 및 **결과** 탭은 별도의 가로 탭 그룹으로 표시됩니다. 또한 모든 프로그램 탭의 로그 데이터 및 결과가 하나의 로그 탭과 하나의 결과 탭에 함께 추가됩니다. 표준 영역에는 각 프로그램마다 개별 로그 및 결과 탭이 있습니다. 자세한 내용은 **“대화식 영역 작업” (54페이지)**에서 확인하십시오.

The screenshot shows the SAS Studio interface with two tabs: 'classfit.sas' and '* SQL_reports.sas'. The main area is divided into '코드' (Code) and '로그 결과' (Log Results). The code editor shows the following SAS code:

```

1 proc sql;
2   select * from sashelp.cars;
3
4   select * from sashelp.class;
5   quit;

```

The '로그 결과' (Log Results) pane displays the following tables:

Print 프로시저					
데이터셋 SASHELP.CLASS					
Volvo	S80 2.5T 4dr	Sedan	Europe	All	
Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	
Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	
Volvo	S80 T6 4dr	Sedan	Europe	Front	
Volvo	V40	Wagon	Europe	Front	
Volvo	XC70	Wagon	Europe	All	

Name	Sex	Age	Height	Weight
Alfred	M	14	69	112.5
Alice	F	13	56.5	84
Barbara	F	13	65.3	98

다음 표에서는 두 영역의 차이점을 볼 수 있습니다.

SAS Studio의 요소	표준 영역에서 사용 가능 여부	대화식 영역에서 사용 가능 여부
탐색 영역 색션		
■ 파일 열기	예	예
■ 탐색기	예	예
■ 스텝	예	아니요
■ 작업	예	아니요
■ Snippet	예	예
■ 라이브러리	예	예
■ Git	예	아니요
■ 파일 참조	예, 기본으로 표시되지 않음	예, 기본으로 표시되지 않음
질의 생성 및 열기 기능	예	아니요
플로우를 생성하고 실행하는 기능	예	아니요
작업 정의 생성 및 실행 기능	예	아니요
코드 호출 및 실행 시 코드 지우기 옵션	아니요	예
PDF, Word, RTF, Excel 및 PowerPoint 출력을 생성하는 옵션	예	아니요
라이브러리에 대한 연결을 테스트하는 기능	예	아니요
데이터 가져오기 기능	예	아니요
데이터 내보내기 기능	예	아니요
데이터 인쇄 기능	예	아니요
테이블 행 필터링 창의 표현식 확인 기능	예	아니요
autoexec.sas 파일 확인 기능	예	아니요
사용자 정의 코드 삽입 기능	예	아니요
칼럼 속성 업데이트 기능	예	아니요

SAS Studio를 시작한 후에는 **옵션** ⇒ **영역 변경**을 선택하고 사용하려는 영역을 선택하여 사용 중인 영역을 변경할 수 있습니다.

보기 ⇒ **탐색 영역**을 선택하여 SAS Studio에 표시되는 탐색 영역 섹션을 지정할 수 있습니다.

자동실행 파일 편집

autoexec.sas 파일에는 SAS Studio를 시작하고 SAS 서버에 연결할 때마다 실행할 SAS 문이 포함됩니다. 예를 들어 SAS Studio를 표준 및 대화식 영역에서 사용할 때마다 사용할 수 있도록 할 라이브러리를 할당하기 위해 autoexec.sas 파일을 사용할 수 있습니다.

주: 새로운 라이브러리 창을 사용하여 라이브러리를 생성하는 경우, **현재 세션 이후에도 이 라이브러리 유지** 옵션을 선택하여 LIBNAME 문을 autoexec.sas 파일에 자동으로 추가합니다. 자세한 내용은 **"라이브러리 작업" (11페이지)**에서 확인하십시오.

autoexec.sas 파일을 편집하려면 다음을 수행합니다.

- 1 **옵션** ⇒ **자동실행 파일**을 선택합니다.
- 2 autoexec.sas 파일에 포함하려는 코드를 입력합니다.
- 3 **실행**을 클릭하여 구문을 확인합니다. **로그** 탭을 열어 로그를 볼 수 있습니다.

주: 자동실행 파일을 실행하는 옵션은 표준 영역에서만 사용할 수 있습니다. 자세한 내용은 **"영역 이해" (31페이지)**에서 확인하십시오.

- 4 **저장**을 클릭하면 자동실행 파일을 저장하고 닫습니다.

사용자 정의 SAS 코드 삽입

프로그램, 작업, 질의 및 가져오기에 대해 코드의 앞 또는 뒤에서 실행될 SAS 코드를 지정할 수 있습니다. 이 사용자 정의 코드는 항상 실행되며 세션 간에 유지됩니다.

사용자 정의 SAS 코드를 포함하려면 다음을 수행합니다.

- 1 **옵션** ⇒ **사용자 정의 코드**를 선택합니다.
- 2 프리앰블 및 포스트앰블 SAS 코드를 지정합니다. **실행**을 클릭하여 구문을 확인합니다.
- 3 **옵션** 탭에서 사용자 정의 코드를 삽입할 위치를 지정합니다. 프로그램, 작업, 질의 및 가져오기 중에서 선택할 수 있습니다. 백그라운드 실행을 실행 중이면 코드를 포함할지 여부도 지정할 수 있습니다.
- 4 **저장**을 클릭합니다.

SAS 세션 재설정

SAS 세션 재설정 옵션을 사용하여 현재 SAS 세션을 종료하고 작업 공간 세션을 생성할 수 있습니다. **SAS 세션 재설정** 옵션은 ENDSAS 문을 실행하여 현재 작업 공간 세션을 종료한 다음 새로운 작업 공간 세션을 생성합니다. 새로운 작업 공간 세션이 시작되면 SAS Studio는 자동실행 파일을 실행하여 세션을 초기화합니다. SAS 세션을 재설정하려면 **옵션** ⇨ **SAS 세션 재설정**을 선택합니다.

주: SAS 세션을 재설정하면 Work 라이브러리의 모든 파일이 삭제됩니다.

바로 가기 키 관리

SAS Studio는 작업을 수행하는 데 사용할 수 있는 많은 바로 가기 키를 포함하고 있습니다. 바로 가기 키 리스트는 *SAS Studio: 접근성 기능의 “바로 가기 키”*에서 참조하십시오.

일부 바로 가기 키를 사용자 정의하고 기본적으로 바로 가기 키가 없는 일부 기능에 바로 가기 키를 할당할 수 있습니다. 바로 가기 키를 관리하려면 **옵션** ⇨ **바로 가기 키 관리**를 선택합니다. 새로운 바로 가기 키를 입력하면 변경 사항을 확인하라는 새 창이 나타납니다.

주: 응용 프로그램의 바로 가기 키를 보려면 SAS Studio 브라우저 창의 오른쪽 하단 모서리에서 를 클릭합니다.

컴퓨팅 컨텍스트 변경

둘 이상의 컴퓨팅 컨텍스트에 액세스할 수 있으면 SAS Studio에서 사용하는 컨텍스트를 변경할 수 있습니다. 컴퓨팅 컨텍스트를 변경하려면 응용 프로그램 창의 오른쪽 상단에 표시되는 현재 컴퓨팅 컨텍스트를 클릭합니다. 사용할 컨텍스트를 선택합니다. 컴퓨팅 컨텍스트를 변경할 때 다음 사항을 참고하십시오.

- 생성한 모든 라이브러리 및 파일 바로 가기가 삭제됩니다.
- 스케줄링된 작업 및 백그라운드 실행은 취소되거나 지워지지 않습니다.
- 열려 있는 모든 탭은 열려 있습니다.

자세한 내용은 *SAS Viya Platform: Server Contexts*의 “[Server Contexts: Concepts](#)”에서 참조하십시오.

주: 한 번에 하나의 컴퓨팅 컨텍스트에서만 작업할 수 있습니다.

프로그램 작업

SAS 코드 편집기 정보	38
Python 코드 편집기 정보	38
프로그램 열기 및 생성	40
프로그램 열기	40
프로그램 생성	41
프로그램 실행	41
프로그램 저장	43
프로그램에서 작업 생성	43
자동 완성 기능 사용	43
구문 도움말 사용	46
괄호 일치	48
텍스트 칼럼 선택	49
여러 줄 편집 사용	49
테이블 이름 및 칼럼 이름 추가	49
경로 이름 삽입	49
작업 또는 질의에서 코드 편집	50
SAS 프로그램 패키지 생성	51
프로그램 요약 생성	51
매크로 변수 사용	51
실행 히스토리 사용	53
SAS 코드 출력형식 자동 지정	53
대화식 영역 작업	54
DATA 스텝 디버거 사용	55
DATA 스텝 디버거 정보	55
DATA 스텝 디버거 시작하기	55
DATA 스텝 디버거 사용	56
Snippet 작업	62
Snippet을 사용하는 이유	62
Snippet 생성	72
코드 Snippet을 삽입하는 방법	72
코드 편집기 사용자 정의	73

SAS 코드 편집기 정보

SAS Studio에는 새로운 SAS 프로그램 또는 기존 SAS 프로그램을 편집할 수 있는 색상 코딩 구문 검사 편집기가 포함되어 있습니다. 이 편집기에는 자동 완성, 자동 서식 지정 및 팝업 구문 도움말 같은 다양한 기능이 포함되어 있습니다. 코드 편집기를 사용하여 SAS 프로그램을 작성, 실행 및 저장할 수 있습니다. 작업을 실행할 때 자동으로 생성되는 코드를 수정하고 저장할 수도 있습니다.

SAS Studio에는 일반적인 작업을 보다 쉽게 프로그래밍하는 데 사용할 수 있는 여러 샘플 SAS 코드 Snippet도 포함되어 있습니다.

The screenshot shows the SAS Studio interface. On the left is the code editor with the following SAS code:

```

24
25 /* summarize the data across a category and store */
26 /* the output in an output data set */
27 proc means data=&data &stat noprint;
28   var &measure;
29   class &category &report;
30   output out=summary &stat=&measure &category /levels;
31 run;
32
33 /* store the value of the measure for ALL rows and
34 /* the row count into a macro variable for use */
35 /* later in the report */
36 proc sql noprint;
37   select &measure, _FREQ_ into :overall, :numobs
38   from summary where _TYPE_=0;
39   select count(distinct &category) into :categorycount fro
40   quit;
41
42 /* sort the results so that we get the TOP values */
43 /* rising to the top of the data set */
44 proc sort data=work.summary out=work.topn;
45   where _type_>2;
46   by &category descending &measure;
47 run;
48
49 /* Pass through the data and output the first N */
50 /* values for each category */
51 data topn;
52

```

On the right is the results window titled 'Top Models by MPG_City for each region of Origin'. It contains the following table:

Origin	Rank	Model	MPG (City)
Asia	1	Insight 2dr (gas/electric)	60
	2	Prius 4dr (gas/electric)	59
	3	Civic Hybrid 4dr manual (gas/electric)	46
	4	Civic HX 2dr	36
	5	Echo 2dr manual	35
	6	Echo 4dr	35
	7	Echo 2dr auto	33
	8	Civic DX 2dr	32
	9	Civic EX 4dr	32
	10	Civic LX 4dr	32
Europe	1	Jetta GLS TDI 4dr	38
	2	Cooper	28
	3	Cooper S	25
	4	GTI 1.8T 2dr hatch	24
	5	Golf GLS 4dr	24
	6	Jetta GL	24
	7	New Beetle GLS 1.8T 2dr	24

Python 코드 편집기 정보

SAS Studio에는 신규 또는 기존 Python 프로그램(*.py) 편집을 위한 색상 코딩 편집기가 포함되어 있습니다. Python 코드 편집기를 사용하면 PROC PYTHON을 명시적으로 사용하지 않고도 Python 프로그램을 작성, 실행 및 저장할 수 있습니다. PROC PYTHON 옵션은 SAS Environment Manager에 저장되며 Python 프로그램에 포함할 필요가 없습니다. 환경에서 사용 가능한 모든 Python 패키지를 사용할 수 있습니다. SAS Studio는 Python 코드를 자동으로 사용하여 PROC PYTHON으로 SAS 프로그램을 생성합니다.

주: 자세한 내용은 *Base SAS Procedures Guide*의 “PYTHON Procedure”에서 확인하십시오.

The screenshot shows a Python code editor window with the following code:

```

1 import os, sys
2 sys.path
3
4 # define our data refereces
5 input_table = 'SASHELP.CLASS'
6 output_table = 'WORK.PYTHONOUT'
7
8 # set input data to pipe from SAS
9 dfin = SAS.sd2df(input_table)
10
11 # output to the log details about the table
12 print("input data shape is:", dfin.shape)
13
14 # call the python transpose
15 dfout = dfin.transpose()
16
17 # output to the log details about the table
18 print("output data shape is:", dfout.shape)
19
20 #set output data using call to pipe to SAS
21 SAS.df2sd(dfout, output_table)
22
23 SAS.submit('proc print data=work.pythonout;');
24
25

```

The output window shows the following table:

OBS	0	1	2	3	4	5	6
1	Alfred	Alice	Barbara	Carol	Henry	James	Jan
2	M	F	F	F	M	M	F
3	14.0	13.0	13.0	14.0	14.0	12.0	12.0
4	69.0	56.5	65.3	62.8	63.5	57.3	59.1
5	112.5	84.0	98.0	102.5	102.5	83.0	84.1

Python 프로그램을 스케줄링하고 Python 프로그램을 작업 또는 백그라운드 실행으로 실행하는 기능을 포함하여 SAS 프로그램에서 사용할 수 있는 많은 기능을 Python 프로그램에서도 사용할 수 있습니다. 편집기 우선 설정을 사용하여 Python 코드 편집기의 동작과 모양을 사용자 정의할 수도 있습니다.

팁 Python 프로그램 내에서 SAS 코드를 실행하려면 SAS.submit 메소드를 사용하십시오. 이 방법은 SAS 옵션을 지정해야 하거나 Python 프로그램에서 SAS 프로시저를 실행하려는 경우에 유용합니다. 예를 들어 다음 코드 줄을 Python 프로그램 시작 부분에 추가하여 VALIDVARNAME 옵션을 실행할 수 있습니다.

```
SAS.submit('options validvarname=any;')
```

다음 코드 예제에서는 SAS.submit 메소드를 사용하여 PROC PRINT 문을 실행합니다.

```
SAS.submit('proc print data=sashelp.class;')
```

프로그램 열기 및 생성

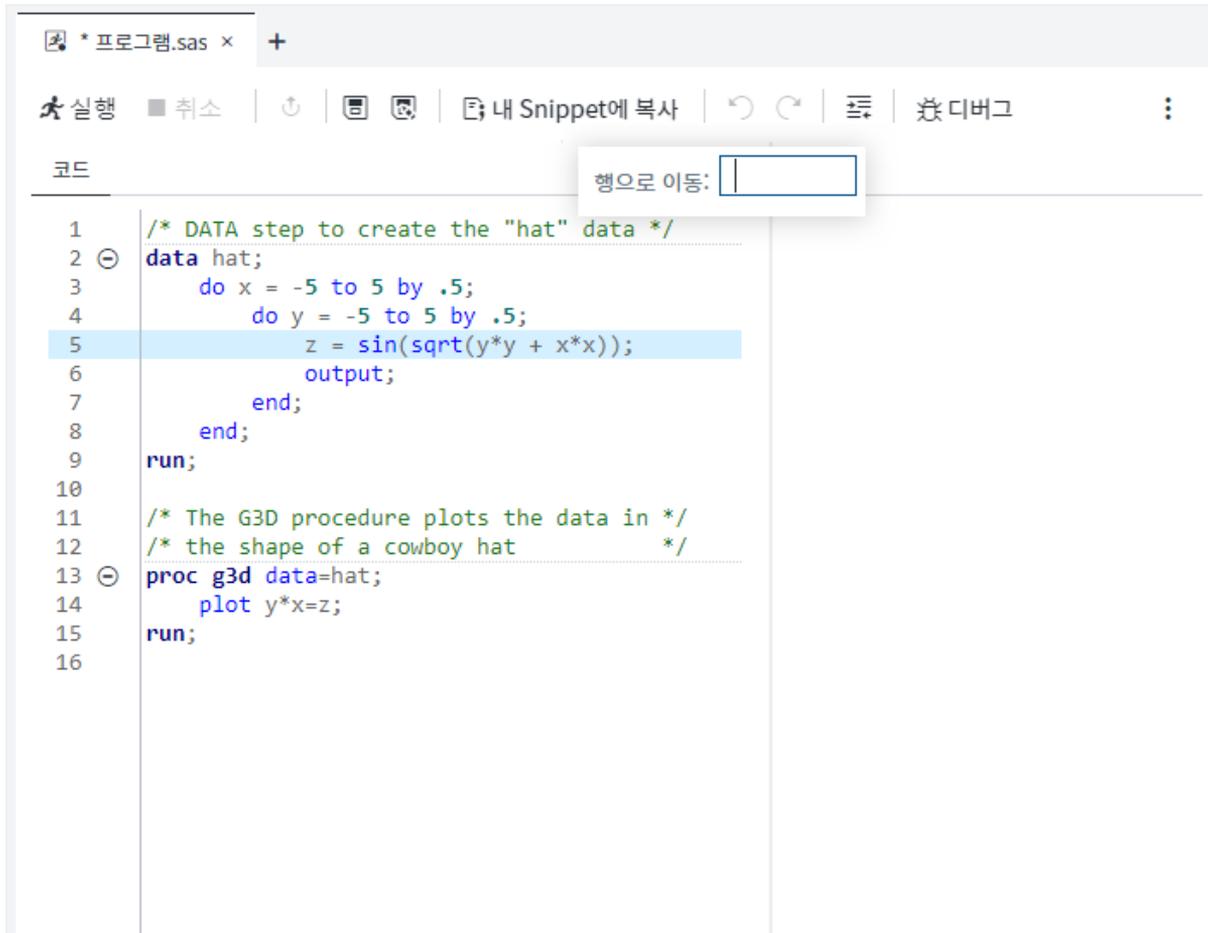
프로그램 열기

탐색 영역의 **탐색기** 섹션에서 SAS 프로그램을 열 수 있습니다. 프로그램을 열려면 해당 폴더를 펼치고 열려는 프로그램을 더블 클릭하거나, 작업 영역에서 프로그램을 마우스 오른쪽 버튼으로 클릭하고 **열기**를 선택합니다. 그러면 해당 프로그램이 작업 영역의 새로운 탭에 열립니다.

주: 도구 모음에서 **열기**를 클릭하고 사용 가능한 폴더에서 파일을 선택하여 파일을 열 수도 있습니다. SAS 프로그램 또는 텍스트 파일을 열 때에는 **인코딩으로 열기**를 선택한 다음 **열기**를 클릭하여 사용할 텍스트 인코딩을 지정할 수 있습니다. **인코딩 선택** 창을 사용하여 원하는 인코딩을 지정합니다.

주: 매우 큰 프로그램 파일을 열면 성능에 영향을 미칩니다. 기본적으로 10MB보다 큰 프로그램 파일을 열면 파일 열기를 계속할지를 확인하는 창이 나타납니다. **텍스트 파일이 지정된 크기보다 크면 경고 표시** 옵션을 사용하여 경고를 표시할 파일 크기를 변경할 수 있습니다. 자세한 내용은 **"일반 우선 설정 지정" (175페이지)**에서 참조하십시오.

코드 편집기의 왼쪽 아래 모서리에서 커서의 현재 행 및 칼럼 위치를 볼 수 있습니다. *.sas 또는 *.txt 파일을 볼 때는 텍스트 인코딩도 표시됩니다. 프로그램에서 특정 행으로 이동하려면 프로그램을 마우스 오른쪽 버튼으로 클릭하고 **행으로 이동**을 선택합니다. **행으로 이동** 상자에 행 번호를 입력합니다. 프로그램을 마우스 오른쪽 버튼으로 클릭하고 **인코딩**을 선택하여 다른 문자 집합 인코딩 옵션을 선택할 수도 있습니다.



프로그램 생성

SAS 프로그램을 생성하려면 **새로 만들기** ⇨ **SAS 프로그램**을 선택합니다.

Python 프로그램을 생성하려면 **새로 만들기** ⇨ **Python 프로그램**을 선택합니다.

주: **새로 만들기** 메뉴를 사용해서도 파일 가져오기를 시작할 수 있을 뿐 아니라 플로우, 질의, 태스크, 작업, 텍스트 파일 및 XML 파일을 생성할 수도 있습니다.

프로그램 실행

프로그램을 작성한 후 프로그램 전체, 코드 영역을 실행하거나 실행할 특정 코드 행을 선택할 수 있습니다.

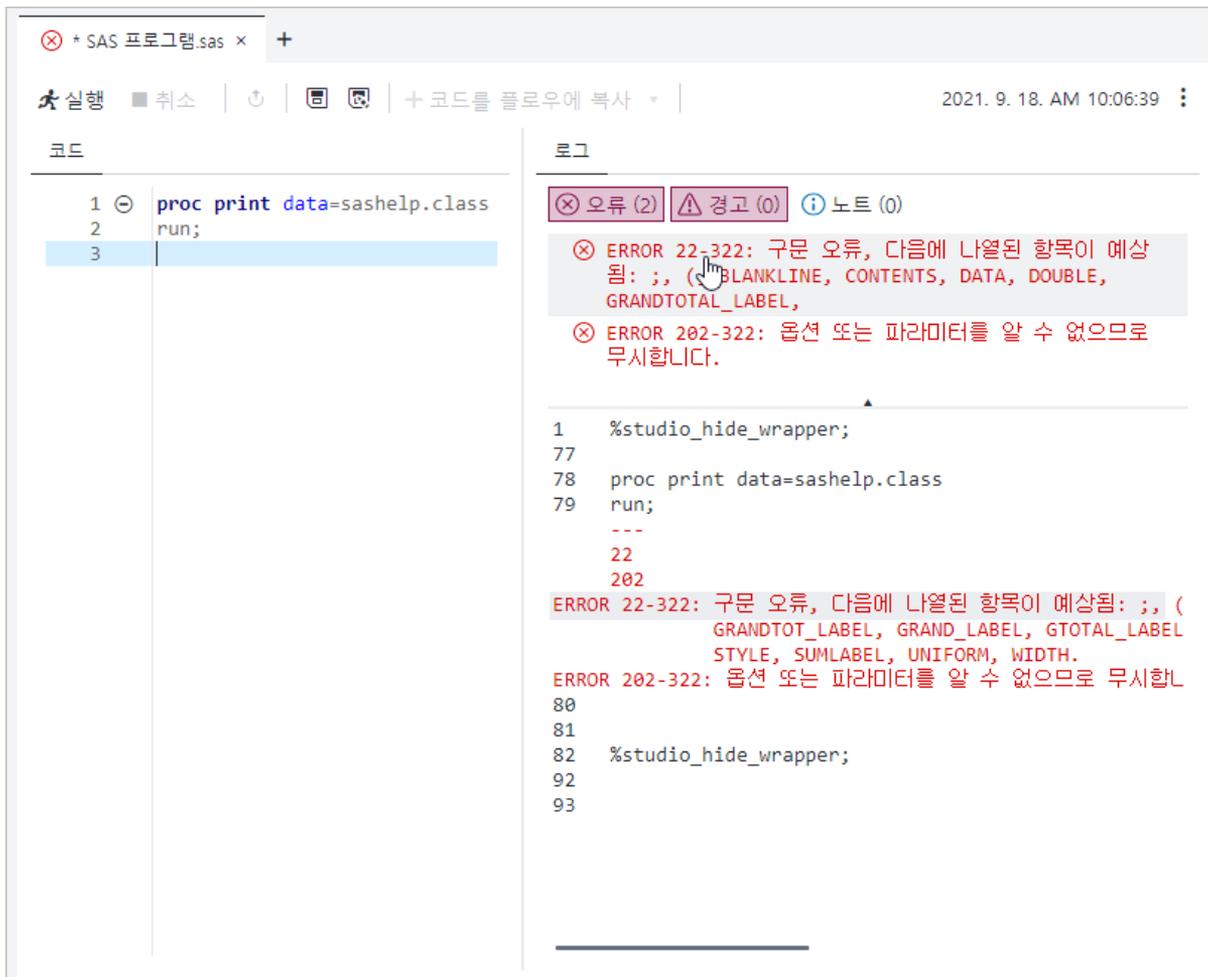
- 전체 프로그램을 실행하려면 도구 모음에서 **실행**을 클릭합니다.
- 코드 영역을 실행하려면 실행할 코드 영역을 마우스 오른쪽 버튼으로 클릭하고 **영역 실행**을 선택합니다. 코드 영역은 코드 편집기에서 접거나 펼칠 수 있는 코드 블록으로 식별

됩니다. 코드 영역을 실행하려면 **코드 접기 사용** 옵션을 선택해야 합니다. 자세한 내용은 **“일반 코드 편집기 우선 설정 지정” (180페이지)**에서 참조하십시오.

- 프로그램의 일부를 실행하려면 실행할 코드 행을 선택하고 **실행**을 클릭합니다.

기본적으로, 오류가 없으면 결과가 자동으로 열리고, 오류가 있으면 **로그** 탭이 열립니다. 하나 이상의 **오류**, **경고** 및 **노트** 섹션을 클릭하여 메시지를 볼 수 있습니다. 메시지를 클릭하면 메시지가 로그에서 발생하는 정확한 위치를 볼 수 있도록 SAS Studio가 해당 메시지를 로그에서 강조 표시합니다.

주: 일반 우선 설정을 사용하여 프로그램 또는 작업 실행 후에 표시할 탭을 지정할 수 있습니다. 자세한 내용은 **“일반 우선 설정 지정” (175페이지)**에서 확인하십시오.



주: 서버 환경에서 작업 중이기 때문에 SAS 프로그램에 ENDSAS 문을 포함하지 마십시오. ENDSAS를 포함하는 프로그램을 실행할 때는 **옵션** ⇒ **SAS 세션 재설정**을 선택하여 SAS 세션을 재설정하십시오.

프로그램 저장

현재 위치에서 기존 프로그램에 대한 업데이트 내용을 저장하려면 도구 모음에서 을 클릭합니다. 전체 프로그램을 새로운 이름으로 또는 새로운 위치에 저장하려면 을 클릭합니다. 프로그램 또는 텍스트 파일을 저장할 때 텍스트 인코딩을 지정하려면 파일을 저장하기 전에 파일을 마우스 오른쪽 버튼으로 클릭하고 **인코딩**을 선택한 다음 새로운 인코딩 옵션을 지정합니다.

주: 프로그램을 저장할 때 SAS 프로그램 패키지 또는 프로그램 요약을 생성할 수도 있습니다. 자세한 내용은 ["SAS 프로그램 패키지 생성" \(51페이지\)](#) 및 ["프로그램 요약 생성" \(51페이지\)](#)에서 확인하십시오.

프로그램에서 작업 생성

저장된 프로그램에서 작업을 생성하려면 탐색기 영역을 엽니다. 저장된 프로그램의 위치로 이동합니다. 프로그램 이름을 마우스 오른쪽 버튼으로 클릭하고 **작업 생성**을 선택합니다. 새로운 작업은 작업 공간에서 작업 정의로 열립니다. 자세한 내용은 [SAS Studio 개발자 가이드: 작업하기의 "새로운 작업 정의 생성"](#)에서 참조하십시오.

프로그램을 작업으로 배포하려면 탐색기 영역을 엽니다. 저장된 프로그램의 위치로 이동합니다. 이름을 마우스 오른쪽 버튼으로 클릭하고 **작업으로 배포**를 선택합니다. 자세한 내용은 [SAS Studio 개발자 가이드: 작업하기의 "플로우 또는 프로그램을 작업으로 배포 및 재배포"](#)에서 참조하십시오.

탐색기 영역에서 작업을 스케줄링할 수도 있습니다. 프로그램 이름을 마우스 오른쪽 버튼으로 클릭하고 **스케줄 작업**을 선택합니다. 자세한 내용은 [SAS Studio 개발자 가이드: 작업하기의 "작업 스케줄링"](#)에서 참조하십시오.

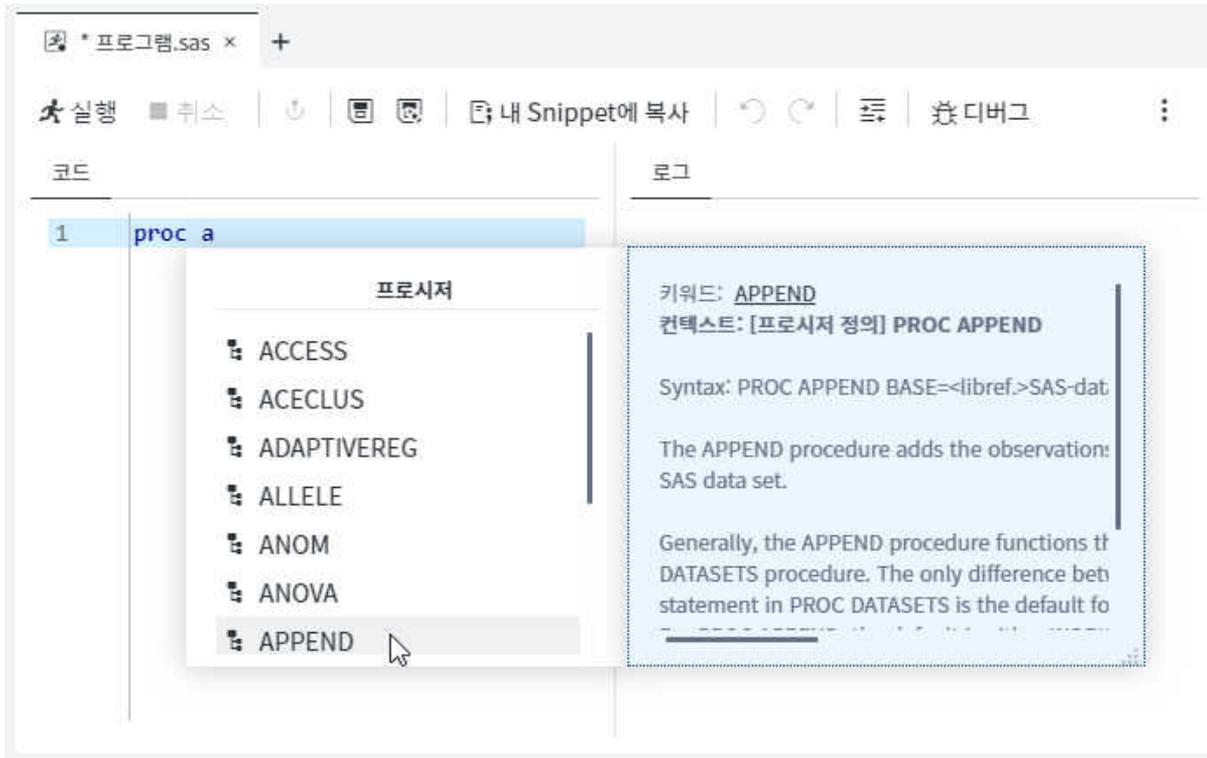
자동 완성 기능 사용

자동 완성 기능 정보

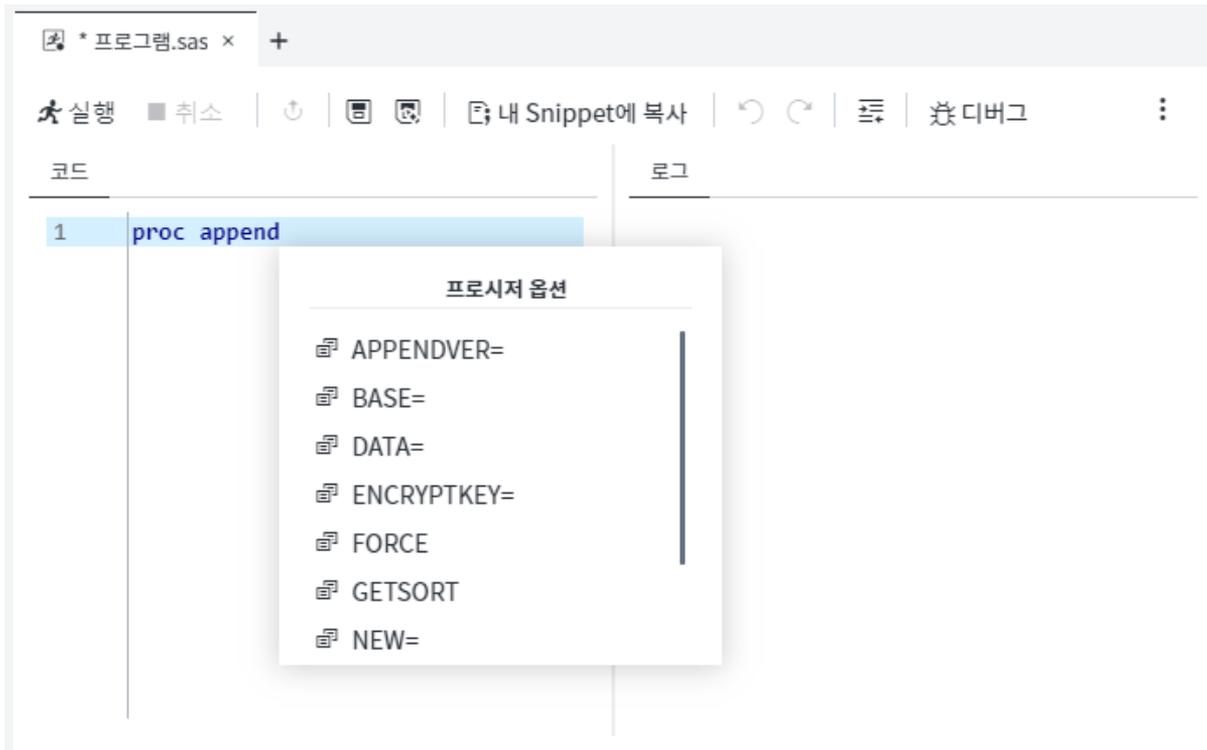
코드 편집기의 자동 완성, 즉 코드 완성 기능은 사용자가 실제로 단어를 모두 입력하기 전에 SAS 프로그램에 입력하려고 하는 다음 단어를 예측하여 완성해 주는 기능입니다. 자동 완성 기능으로 SAS 프로시저, 문장, 매크로, 함수, CALL 루틴, 출력형식, 입력형식, 매크로 변수, SAS 색상, 스타일 요소, 스타일 속성 및 통계 키워드에 관련된 키워드와 다양한 SAS 문 및 프로시저 옵션을 완성할 수 있습니다. 자동 완성 기능은 라이브러리 참조 및 테이블 이름도 완성할 수 있습니다.

주: 자동 완성 기능은 SAS 프로그램을 편집할 때만 사용할 수 있습니다.

이 예에서는 코드 편집기에 proc a를 입력하면 키워드 및 도움말이 표시됩니다.



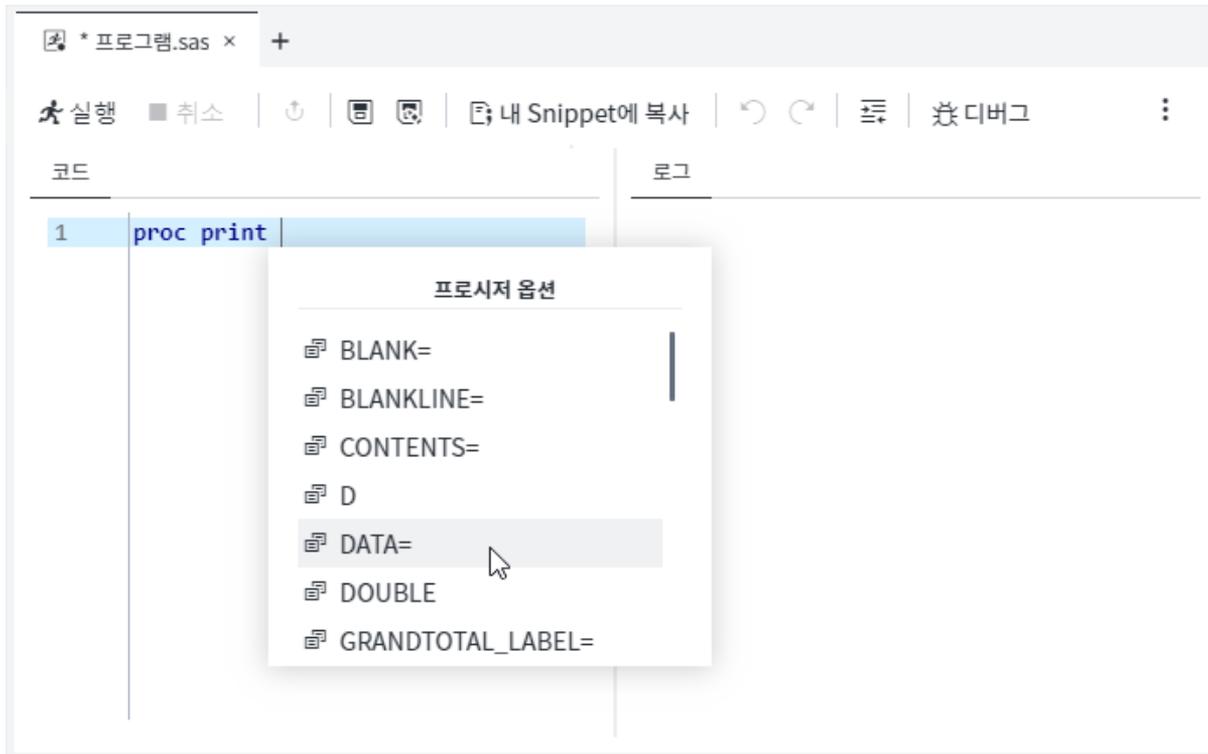
이 예에서는 프로시저 리스트에서 **추가**를 선택하여 proc append가 코드 편집기에 나타나도 록 합니다. 빈 칸을 입력하면 코드 편집기는 APPEND 프로시저의 옵션 리스트를 표시합니 다.



자동 완성 기능을 사용하는 방법

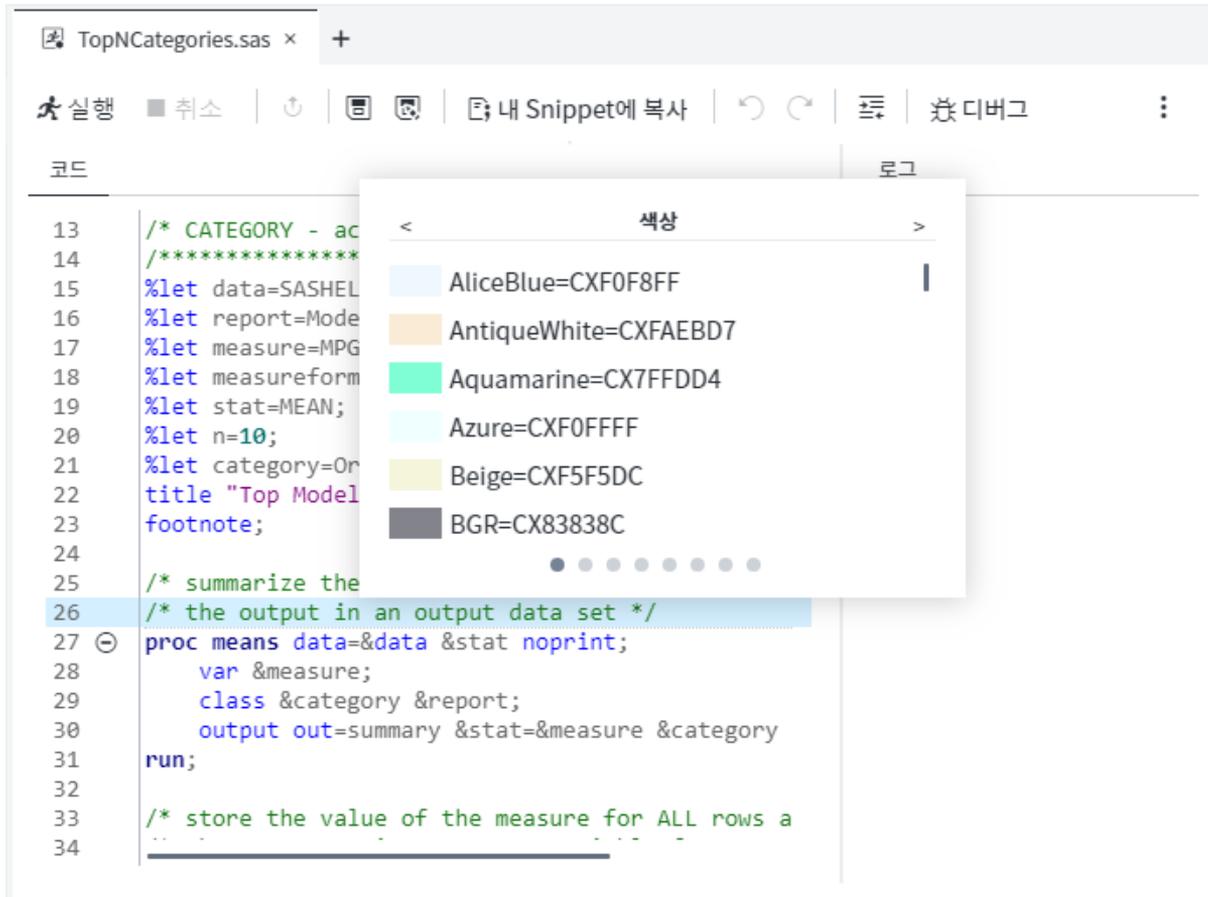
자동 완성 기능을 사용하려면 다음을 수행합니다.

- 1 자동 완성 리스트를 여는 방법은 추가할 키워드에 따라 다릅니다.
 - 글로벌 문장, DATA 스텝 문, CALL 루틴, 프로시저, 매크로 문장 또는 자동 매크로 변수를 추가하려면 사용할 키워드의 처음 문자를 하나 이상 입력합니다.
해당 문자로 시작하는 추천 키워드 리스트가 있는 창이 열립니다.



- 색상, 출력형식, 입력형식, 매크로 함수, SAS 함수, 통계량 키워드, 스타일 요소 또는 스타일 속성을 지정하려면, 마우스 포인터를 주석에 올려 놓고 Ctrl+스페이스바를 누릅니다. 옵션 리스트를 뒤로 탐색하려면, Ctrl+Shift+스페이스바를 누르거나 팝업 창 상단에 있는 화살표 키를 사용합니다.

주: 우선 설정 창에서 자동 완성 사용 옵션을 선택 취소해도 바로 가기는 작동합니다. 자세한 내용은 “코드 편집기 사용자 정의” (73페이지)에서 확인하십시오.



2. 사용하려는 키워드를 다음과 같은 방식으로 탐색할 수 있습니다.
 - 올바른 키워드가 선택될 때까지 계속 입력합니다(입력을 계속할수록 원하는 단어와 가까워짐).
 - 위쪽 또는 아래쪽 화살표나 Page Up 및 Page Down 키를 사용하여 리스트를 스크롤합니다.
3. 선택한 키워드를 더블 클릭하거나 Enter 키를 눌러 프로그램에 키워드를 추가할 수 있습니다.

구문 도움말 사용

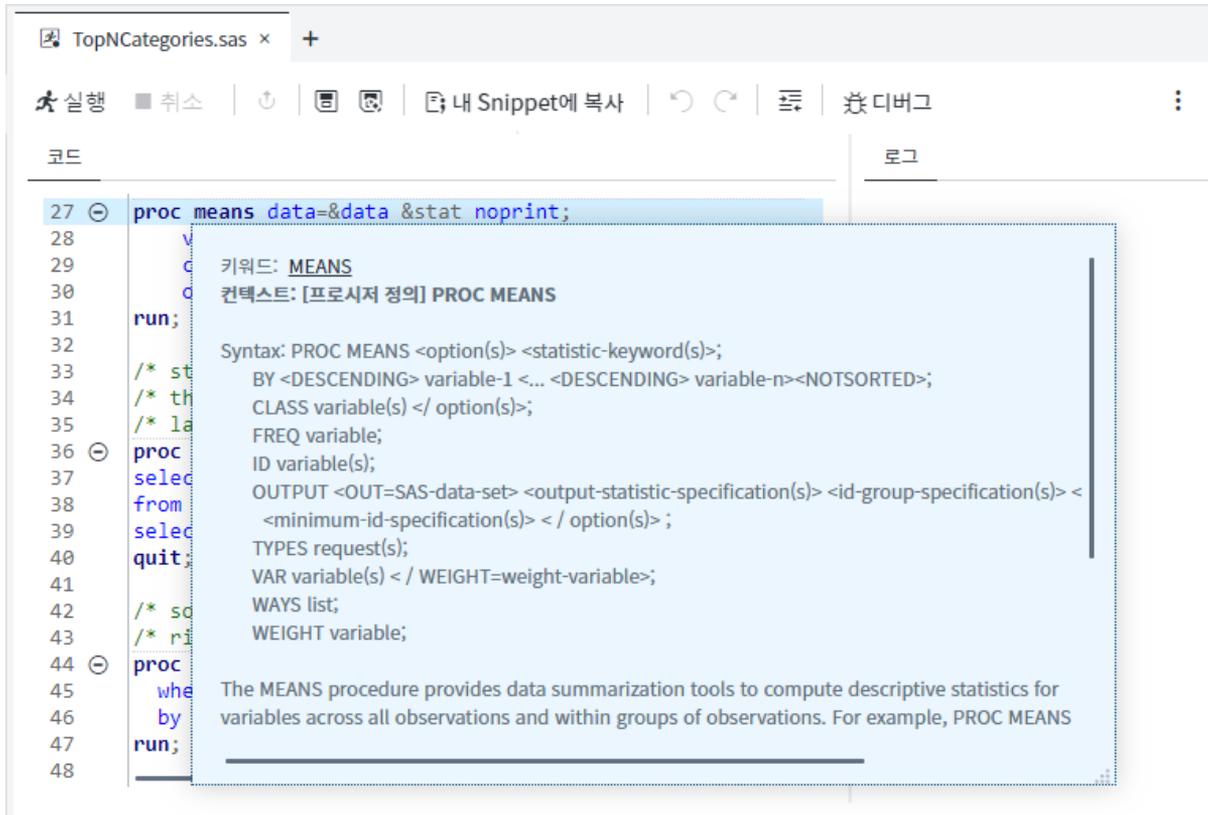
코드 편집기에서 SAS 프로그램을 작성 및 편집할 때는 간략한 SAS 구문 문서가 표시됩니다. 다음과 같은 방법으로 도움말을 표시할 수 있습니다.

- 프로그램에서 키워드를 마우스 오른쪽 버튼으로 클릭하고 **구문 도움말**을 선택합니다.
- 올바른 SAS 키워드를 입력한 다음 자동 완성 창에서 추천 키워드를 클릭하십시오.
- 프로그램에서 올바른 SAS 키워드 위로 마우스 포인터를 가져갑니다. 이 작업은 편집기 우선 설정에서 **키워드에 마우스를 올리면 힌트를 표시합니다**. 옵션을 선택했을 때만 작동합니다. 자세한 내용은 **"코드 편집기 사용자 정의" (73페이지)**에서 확인하십시오.

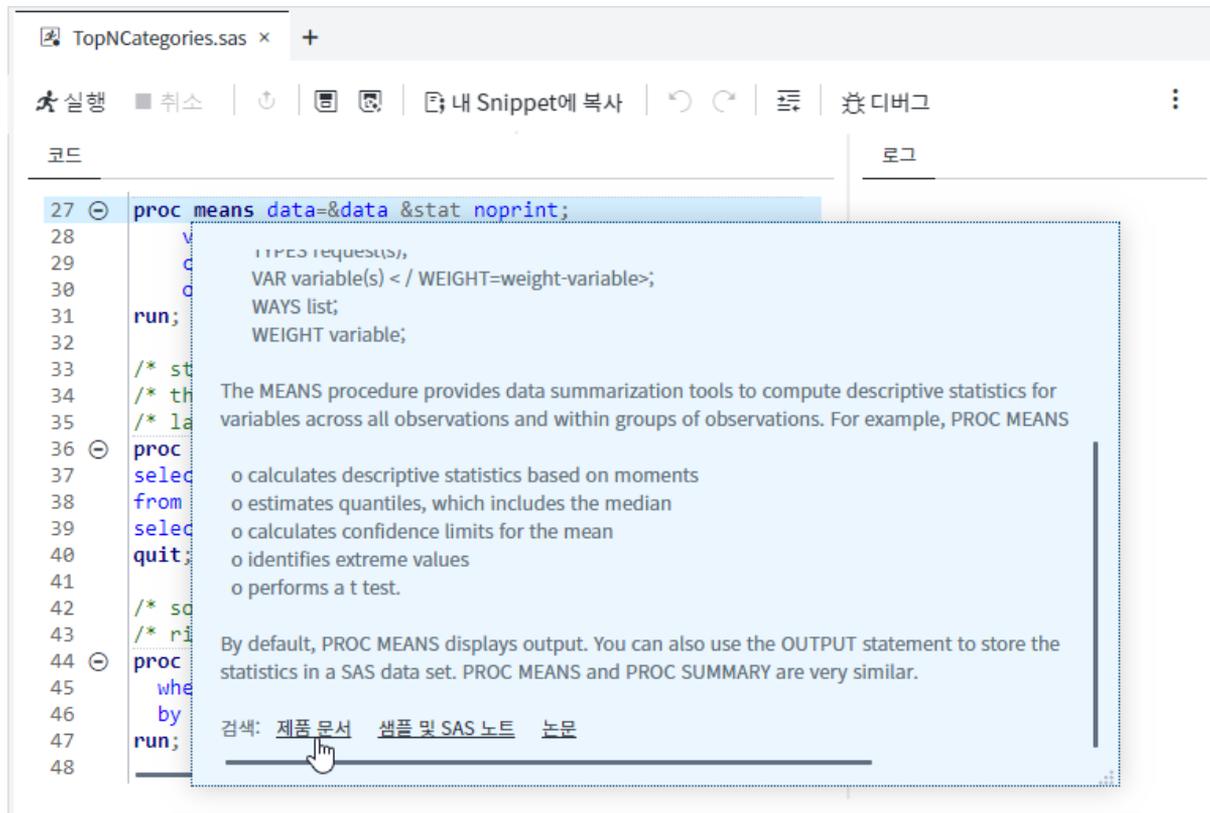
SAS 제품 문서에는 SAS 언어에 대한 보다 포괄적인 사용 정보가 나와 있지만 코드 편집기의 구문 도움말에 나오는 구문에 대한 힌트와 키워드에 대한 간략한 설명을 참조하여 작업을 시

작할 수도 있습니다. 다음과 같은 방식으로 구문 도움말의 링크를 클릭하여 추가 도움말을 얻을 수 있습니다.

- 창 위쪽에 있는 키워드 링크를 클릭하여 support.sas.com 웹 사이트에서 해당 키워드를 검색할 수 있습니다.



- 창 아래쪽에 있는 링크를 클릭하여 SAS 제품 문서, 샘플 및 SAS 노트, SAS 기술 문서 등에서 키워드를 검색할 수 있습니다.



괄호 일치

괄호 일치 기능을 사용하여 프로그램 내에서 중첩된 괄호를 추적할 수 있습니다. 코드 편집기는 열기 및 닫기 괄호를 모두 강조 표시합니다. 하나의 괄호만 강조 표시되면, 다른 하나의 괄호가 없음을 알 수 있습니다. 이 기능은 괄호, 대괄호, 각쇠 괄호 및 중괄호를 일치시키는데 사용할 수 있습니다.

```

do y = -5 to 5 by .5;
  z = sin(sqrt(y*y + x*x));
  
```

괄호를 일치시키려면 원하는 여는 괄호 앞 또는 닫는 괄호 바로 뒤에 커서를 놓습니다. 괄호 및 일치하는 해당 괄호가 강조 표시됩니다. 괄호가 일치하지 않으면 강조 표시되지 않습니다.

주: 코드 편집기 설정 페이지를 사용하여 괄호 일치 강조 표시 옵션을 해제할 수 있습니다. 닫는 대괄호 또는 괄호를 자동으로 삽입하는 옵션을 선택할 수도 있습니다. 자세한 내용은 "일반 코드 편집기 우선 설정 지정" (180페이지)에서 확인하십시오.

텍스트 칼럼 선택

텍스트의 전체 가로 행을 선택할 필요는 없습니다. 텍스트 칼럼 또는 수직 블록을 선택할 수 있습니다.

텍스트 칼럼 또는 수직 블록을 선택하려면 다음을 수행하십시오.

- Windows 환경에서는 마우스 왼쪽 버튼으로 텍스트를 선택한 상태에서 Alt 키를 누릅니다.
- Apple OS X 환경에서는 마우스 왼쪽 버튼으로 텍스트를 선택한 상태에서 Option 키를 누릅니다.

여러 줄 편집 사용

동시에 여러 줄의 코드를 편집할 수 있습니다. 이 기능은 여러 위치의 코드에서 동일한 추가 또는 삭제를 수행해야 할 때 유용합니다.

여러 줄 편집을 사용하려면 다음을 수행하십시오.

- Windows 환경에서는 Ctrl 키를 누르고 편집하려는 각 위치를 마우스 왼쪽 버튼으로 클릭합니다. 여러 줄 편집을 종료하려면 코드 편집기 창에서 다른 위치를 클릭합니다.
- Apple OS X 환경에서는 Command 키를 누르고 편집하려는 각 위치를 마우스 왼쪽 버튼으로 클릭합니다. 여러 줄 편집을 종료하려면 코드 편집기 창에서 다른 위치를 클릭합니다.

테이블 이름 및 칼럼 이름 추가

탐색 영역의 **라이브러리** 섹션에서 끌어서 놓기 작업으로 테이블 이름 및 칼럼 이름을 SAS 코드로 이동할 수 있습니다. 예를 들어 Sashelp.Cars 테이블을 PRINT 프로시저의 DATA 옵션으로 이동할 수 있습니다. 마우스를 놓으면 해당 테이블의 전체 이름이 코드에 나타납니다.

.....
 주: 테이블 또는 칼럼 이름이 데이터베이스의 예약어이면 프로그램에서 이름이 올바르게 평가될 수 있도록 SAS Studio가 따옴표와 소문자 n을 자동으로 추가하여 이름 리터럴을 생성합니다.

경로 이름 삽입

탐색 영역의 **탐색기** 섹션에서 폴더 또는 *.sas 파일을 마우스 오른쪽 버튼으로 클릭하고 **경로로 삽입**을 선택하여 프로그램에 폴더의 경로 이름 또는 파일의 경로 이름과 파일 이름을 추가합니다. 예를 들어

/products/salesdata/와 같은 폴더를 마우스 오른쪽 버튼으로 클릭하고 **경로로 삽입**을 선택하여 해당 폴더를 LIBNAME 문에 추가할 수 있습니다.

주: 이 옵션은 독립 실행형 프로그램에만 사용할 수 있습니다. 플로우의 SAS 프로그램 노트에는 사용할 수 없습니다.

작업 또는 질의에서 코드 편집

작업 또는 질의를 실행한 다음 수정 사항을 적용해서 실행할 때 자동으로 생성되는 코드를 편집할 수 있습니다. 코드를 편집하면 해당 코드가 별도의 프로그램 창으로 열립니다. 코드는 원래 작업 또는 질의와 더 이상 연결되지 않습니다.

작업 또는 질의에서 생성된 프로그램을 편집하려면 다음을 수행합니다.

- 1 작업 영역의 해당 작업 또는 질의 탭에서 **코드** 하위 탭을 클릭하여 작업 또는 질의와 연결된 코드를 표시합니다.
- 2 도구 모음에서 **코드 편집**을 클릭합니다. 코드가 새로운 프로그램 창에서 열립니다.

The screenshot shows the SAS Studio interface. On the left, there is a '데이터' (Data) pane with 'SASHELP.CLASS' selected. Below it, '역할' (Roles) are listed, including '분석변수:' (Analysis Variables) with 'Weight' and '분류변수:' (Classification Variables) with 'Age'. At the bottom, there is an '작업 콘솔(0)' (Job Console) pane showing '항목 없음' (No items).

The main area is a '코드' (Code) editor. The toolbar at the top right of the code editor includes a '코드 편집' (Code Edit) button, which is highlighted with a red box and a mouse cursor. The code in the editor is as follows:

```

0
1
2
3
4
5
6
7 * 생성 서버: 'sas-launcher-81833ce0-9852-4dbd-960e-4537e
8 * SAS 플랫폼 "Linux LIN X64 3.10.0-957.1.3.el7.x86_64"에
9 * SAS 버전 "V.04.00M0P041920"에서 생성됨
10 * 브라우저 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) Ap
11 * 웹 클라이언트 "http://d17209.scnylz-cg1-m1.dsgproject.s
12 */
13
14 ods noproctitle;
15 ods graphics / imagemap=on;
16
17 proc means data=SASHELP.CLASS chartype mean std min max r
18 var Weight;
19 class Age;
20 output out=work.Means_stats mean=std=min=max=n= / aut
21 run;
22
23 proc univariate data=SASHELP.CLASS vardef=df noprint;
24 var Weight;
25 class Age;
26 histogram Weight / normal(noprint) kernel;
27 run;
28
29 proc sort data=SASHELP.CLASS out=WORK.TempSorted2236;
30 by Age;
31 run;
32
33 proc boxplot data=WORK.TempSorted2236;
34 plot (Weight)*Age / boxstyle=schematic;
35 inset mean stddev min max nobs / position=nw;
36 run;
37
38 proc datasets library=WORK noprint;
39 delete TempSorted2236;
40 run;

```

SAS 프로그램 패키지 생성

SAS 프로그램 패키지는 로그 및 HTML 결과와 함께 제공되는 프로그램의 스냅샷을 포함하는 파일입니다. 작업을 실행할 때 작성한 코드뿐 아니라 자동으로 생성된 코드에서도 프로그램 패키지를 생성할 수 있습니다. SAS Studio에서 프로그램 패키지를 열었을 때 프로그램을 다시 실행하지 않아도 코드, 로그 및 결과에 액세스할 수 있습니다. 코드를 변경하여 다시 실행하면 패키지는 자동으로 업데이트되지 않습니다. 변경 사항을 유지하려면 패키지를 다시 저장해야 합니다.

주:

- 프로그램 패키지는 PDF 또는 RTF 결과를 포함하지 않습니다.
- 대화식 영역을 사용 중이면 프로그램 패키지를 생성할 수 없습니다.

프로그램 패키지 파일을 생성하려면 사용할 코드를 열고 을 클릭합니다. 파일 위치 및 이름을 지정하고 **유형** 옵션으로 **패키지**를 선택합니다.

주: 프로그램 패키지 파일을 열고 프로그램 또는 로그를 개별적으로 저장하거나 결과를 HTML, PDF 또는 RTF 파일로 다운로드하려면 프로그램 패키지 파일을 열고 프로그램을 다시 실행해야 합니다.

프로그램 요약 생성

작성한 코드뿐 아니라 자동으로 실행할 때 자동으로 생성된 코드에 대한 요약 페이지를 생성할 수 있습니다. 프로그램 요약 페이지는 별도의 브라우저 탭에서 열리는 HTML 파일입니다. 여기에는 프로그램 실행, 전체 소스 코드, 전체 로그 및 결과에 대한 정보가 포함됩니다. 프로그램에 대한 프로그램 요약 페이지를 보려면 을 클릭하고 **브라우저 탭에서 열기** ⇨ **요약**을 선택합니다.

주: 프로그램 요약은 프로그램을 실행한 후에만 사용할 수 있습니다.

을 클릭하고 **다른 유형으로 저장** 옵션으로 **요약**을 선택하여 프로그램 요약 파일을 지정한 폴더에 저장할 수도 있습니다. 프로그램 요약은 HTML 파일로 저장됩니다.

매크로 변수 사용

매크로 변수는 응용 프로그램의 이름이나 버전 번호 등 프로그램 또는 작업이 실행될 때 얻는 정보를 추가하는 데 사용할 수 있습니다. 이러한 항목은 코드, 제목 또는 각주 앞에 "&"를 붙여 참조할 수 있습니다.

예를 들어 다음과 같이 각주에서 매크로를 사용할 수 있습니다.

```
Generated with &_CLIENTAPP &_CLIENTVERSION
```

주: 다음 매크로 변수뿐 아니라 이 코드를 실행하여 사용 가능한 기타 사용자 생성 및 자동 매크로 변수를 볼 수 있습니다.

```
%PUT _ALL_;
```

SAS 매크로 함수 및 변수에 대한 자세한 내용은 *SAS Macro Language Reference*를 참조하십시오.

매크로 변수	설명
_CLIENTAPP	클라이언트 응용 프로그램의 이름
_CLIENTAPPABBREV	클라이언트 응용 프로그램의 축약된 이름
_CLIENTMACHINE	클라이언트 컴퓨터의 노드 이름
_CLIENTMODE	SAS Studio 모드 유형: viya (표준 모드) 또는 interactive (대화식 모드).
_CLIENTVERSION	응용 프로그램 버전입니다.
_SASHOSTNAME	서버 노드 이름(IP 주소 또는 DNS 이름)
_SASPROGRAMFILE	현재 실행 중인 SAS 프로그램의 전체 경로 및 파일 이름입니다. 매크로 변수는 SAS Studio 코드가 실행 중인 동일한 서버에 저장된 SAS 프로그램 파일에서만 사용할 수 있습니다.
_SASPROGRAMFILEHOST	현재 SAS 프로그램이 실행되고 있는 서버 노드 이름
_SASWORKINGDIR	현재 작업 중인 디렉터리
SASWORKLOCATION	Work 라이브러리의 위치
SYSPROCESSNAME	현재 SAS 프로세스의 이름
SYSPROCESSMODE	현재 SAS 세션 실행 모드 또는 서버 유형 이름
SYSVLONG4	SAS 소프트웨어 릴리스 번호, 유지보수 레벨 및 4자리 년도

주: SAS 프로그램에 %put_all_ 또는 %put_global_을 지정하면 출력에는 특수 문자가 포함되지 않습니다. 예를 들어 슬래시는 디렉터리 경로에 포함되지 않습니다. 특수 문자를 포함한 출력을 보려면 %put &_sasprogramfile;과 같이 이름을 기준으로 개별 매크로 변수를 지정해야 합니다.

실행 히스토리 사용

실행 상태 창을 사용하여 이전 버전의 실행 코드에 액세스할 수 있습니다. 도구 모음에서 **보기** ⇨ **실행 상태**를 선택하거나 SAS Studio 브라우저 창의 오른쪽 하단 모서리에서 **실행**을 클릭합니다. 실행 상태 탭에는 프로그램을 실행할 때마다 엔트리를 포함합니다. 실행 상태 창에서 프로그램을 클릭하면 실행 당시의 프로그램을 열게 됩니다. 프로그램이 작업 영역의 새로운 탭에서 열리고 변경 사항을 적용 및 저장할 수 있습니다.

주: SAS Studio에서 로그아웃하면 실행 히스토리가 자동으로 지워집니다.

자세한 내용은 ["실행 상태 창 사용" \(26페이지\)](#)에서 확인하십시오.

SAS 코드 출력형식 자동 지정

코드 편집기에서 프로그램을 읽기 쉽도록 코드 출력형식을 자동으로 지정할 수 있습니다. 코드 출력형식을 자동으로 지정하면 줄 바꿈이 추가되고 중첩 수준에 따라 각 줄에 적절하게 들여쓰기가 적용됩니다. 코드 편집기에서 코드 출력형식을 지정하려면 도구 모음에서 **꺾기**를 클릭합니다. 선택한 코드 섹션의 출력형식을 지정하도록 선택할 수도 있습니다.

예를 들어 다음 코드는 들여쓰기와 논리적 줄 바꿈이 없으므로 읽기가 어렵습니다.

```
data topn;
length rank 8; label rank="Rank";
set topn; by &category descending &measure;
if first.&category then rank=0; rank+1;
if rank le &n then output;
run;
```

코드 출력형식 자동 지정 기능을 사용한 후에는 프로그램이 다음과 같이 표시됩니다.

```
data topn;
  length rank 8;
  label rank="Rank";
  set topn;
  by &category descending &measure;

  if first.&category then
    rank=0;
    rank+1;

  if rank le &n then
    output;
run;
```

대화식 영역 작업

대화식 영역이란?

일부 SAS 프로시저는 대화식으로 실행되므로 사용자가 QUIT 문을 실행하거나 새로운 PROC 또는 DATA 스텝을 실행할 때까지 활성 상태로 유지됩니다. SAS Studio에서는 코드 편집기를 사용하여 이러한 프로시저뿐 아니라 다른 SAS 프로시저도 대화식 영역에서 실행할 수 있습니다.

대화식 영역을 사용하면 SAS 프로그램에서 선택한 코드 행을 실행하고 그 결과에 따라 다음 스텝을 결정할 수 있습니다. 예를 들어 SAS/OR의 OPTMODEL 프로시저를 사용하면 산술적 프로그래밍 모델을 모델링하고 해를 구할 수 있습니다. 이 프로시저를 대화식으로 실행하면 전체 프로그램을 실행할 필요 없이 프로그램의 일부에 대한 결과를 빠르게 검사하여 수정이 필요한지 여부를 결정할 수 있습니다.

대화식 영역에서 프로그램 실행

프로그램을 대화식 영역에서 실행하면 ODS 및 %LET 문 같은 자동 생성 코드는 프로그램에 추가되지 않습니다. 또한 결과가 HTML로만 생성됩니다. 대화식 영역에서는 로그와 결과가 기존 로그 및 결과에 추가됩니다. 이전에 실행된 코드는 사용자가 종료할 때까지 활성 상태로 유지됩니다.

주: 을 클릭하고 **로그 지우기** 또는 **결과 지우기**를 선택하여 기존 로그 및 결과를 지울 수 있습니다.

예를 들어 다음과 같은 프로그램이 있다고 가정합니다.

```
proc sql;
  select * from sashelp.cars;

  select * from sashelp.class;
quit;
```

표준 영역에서는 코드의 처음 두 행을 선택하여 실행하면 코드가 성공적으로 실행됩니다. 그런 다음 코드의 마지막 두 행을 선택하여 실행하면 PROC SQL 문이 없기 때문에 코드가 실패합니다.

대화식 영역으로 전환하고 동일한 스텝을 따르면 PROC SQL 문이 여전히 활성 상태이므로 코드의 마지막 두 행이 성공적으로 실행됩니다.

주: 특정 프로시저에 대한 문서는 support.sas.com에서 SAS Programmer's Bookshelf를 참조하십시오.

DATA 스텝 디버거 사용

DATA 스텝 디버거 정보

DATA 스텝 디버거는 DATA 스텝 프로그램에서 논리 오류를 찾을 수 있는 도구입니다. DATA 스텝 디버거를 사용하면 프로그램 실행 시 프로그램 변경에서 변수값을 볼 수 있습니다. 프로그램을 행 단위로 실행할 수 있고 프로그램에 특정 중단점을 설정할 수도 있습니다.

주: DATA 스텝 디버거는 DATA 스텝 프로그램에서만 작동하며 프로그램에 구문 오류가 없는 경우에만 작동합니다.

DATA 스텝 디버거 시작하기

디버깅이란?

*디버깅*은 프로그램에서 논리 오류를 제거하는 과정입니다. 구문 오류와 달리 논리 오류는 프로그램의 실행을 중지하지 않습니다. 대신 논리 오류가 있으면 프로그램에서 예기치 않은 결과가 생성됩니다. 예를 들어 인벤토리를 추적하는 DATA 스텝을 생성하고 웨어하우스가 가득 차 있을 때 프로그램에서 재고가 없음이 표시되면, 프로그램에 논리 오류가 있는 것입니다.

DATA 스텝 디버거가 없으면 PUT 문과 같은 코드를 프로그램에 추가하고 결과 및 로그를 검사하여 프로그램을 디버깅할 수 있습니다. SAS 로그는 데이터 오류를 식별하는 데 도움이 되지만, DATA 스텝 디버거를 사용하면 논리 오류 그리고 DATA 스텝의 데이터 오류까지 손쉽게 대화식으로 찾아낼 수 있습니다.

디버깅의 작동 방식

DATA 스텝 디버거에서 DATA 스텝을 열면 SAS에서 스텝을 컴파일하고, DATA 스텝 디버거 창에 코드를 표시하고, 사용자가 실행을 시작할 때까지 일시 중지합니다. 실행이 일시 중지될 때마다 각 변수의 현재 값이 변수 영역에 표시됩니다. 자세한 내용은 ["변수 값 검사" \(60 페이지\)](#)에서 확인하십시오.

다음 몇 가지 방법으로 DATA 스텝 디버거에서 프로그램을 실행할 수 있습니다.

- ▶을 클릭하여 실행을 시작하면 SAS는 DATA 스텝에서 각 문장을 실행합니다.

- DATA 스텝 프로그램의 특정 행에서 실행을 일시 중지하려면 중단점을 설정할 문장을 하나 이상 선택합니다. ▶을 클릭하면 SAS는 중단점에 도달할 때까지 프로그램을 실행합니다. 자세한 내용은 “중단점 설정 및 지우기” (59페이지)에서 확인하십시오.
- DATA 스텝을 한 번의 하나의 문장씩 실행하려면 ☐을 클릭합니다. 프로그램의 각 스텝 뒤에서 변수 값을 검사할 수 있습니다. 자세한 내용은 “프로그램을 스텝별로 실행하기” (59페이지)에서 확인하십시오.
- 특정 변수의 값이 변경될 때까지 DATA 스텝을 실행하려면 관찰할 변수를 선택하고 ▶을 클릭합니다. 그러면 SAS는 지정된 변수의 값이 변경될 때까지 프로그램을 실행합니다. 자세한 내용은 “변수 값 관찰” (60페이지)에서 확인하십시오.

주: DATA 스텝 디버거를 사용할 때는 다음 사항에 주의하십시오.

- DATA 스텝 디버거는 표준 영역에서만 사용할 수 있습니다. 자세한 내용은 “영역 이해” (31페이지)에서 확인하십시오.
- DATA 스텝을 디버깅하면 DATA 스텝의 문장은 디버깅 세션 외부에서와 마찬가지로 여러 번 반복될 수 있습니다. 마지막 반복이 끝나면 디버그 콘솔에 메시지가 나타납니다.
- DATALINES 및 CARDS 문이 포함된 DATA 스텝은 디버깅할 수 없습니다. 다음 방법으로 이 제한 사항을 해결할 수 있습니다.
 - 데이터를 플랫폼 파일에 저장하고 FILENAME 문을 사용하여 데이터를 DATA 스텝으로 읽습니다.
 - 두 개의 개별적인 DATA 스텝을 생성합니다. 하나는 DATALINES 또는 CARDS 문을 포함하고 다른 하나는 데이터를 처리하기 위한 디버깅 가능한 DATA 스텝을 포함합니다.
- 한 번의 하나의 DATA 스텝만 디버깅할 수 있습니다. DATA 스텝 디버거는 DATA 스텝에만 사용할 수 있고 PROC 스텝에는 사용할 수 없습니다.

DATA 스텝 디버거 사용

DATA 스텝 디버거 창 정보

DATA 스텝 디버거 창에는 DATA 스텝 프로그램의 디버깅을 용이하게 해주는 여러 기능이 포함되어 있습니다.

- 현재 실행 중인 행이 파란색으로 강조 표시됩니다.
- 창 오른쪽의 변수 영역에는 프로그램의 각 변수와 해당 변수의 현재 값이 나열됩니다.
- 필터 및 검색 상자를 사용하여 변수 영역에서 변수의 리스트 범위를 좁힐 수 있습니다.
- 프로그램의 스텝을 진행할 때 변경되는 변수 값은 빨간색으로 표시됩니다.
- 창 하단의 디버그 콘솔에는 프로그램 실행에 따른 로깅 정보가 표시됩니다.
- 창 하단의 명령행에 직접 디버거 명령을 입력할 수 있습니다. 올바른 명령 리스트를 보려면 ⓘ을 클릭합니다.

DATA 스텝 디버거 닫기

▶ 계속 ■ 중지 🔍 ● ⚙️

```

1  data horsepower;
2  set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
3  format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
4  retain runninghorses;
5  by make;
6
7  if _n_ eq 1 then
8    runningHorses=0;
9  else runninghorses=runninghorses+horsepower;
10 runningaveragehorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;

```

콘솔

DATA 스텝 디버거가 시작되었습니다.
정지 : 행 2 칼럼 2

> 명령 입력

🔍 필터

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make		<input type="checkbox"/>
Model		<input type="checkbox"/>
Type		<input type="checkbox"/>
Origin		<input type="checkbox"/>
DriveTrain		<input type="checkbox"/>
MSRP	.	<input type="checkbox"/>
Invoice	.	<input type="checkbox"/>
EngineSize	.	<input type="checkbox"/>
		—

DATA 스텝 디버거 열기

DATA 스텝이 포함된 모든 프로그램에 대해 DATA 스텝 디버거를 실행할 수 있습니다. DATA 스텝 디버거 사용을 시작하려면 먼저 DATA 스텝 디버거를 활성화하고 디버깅하려는 올바른 DATA 스텝 코드를 선택해야 합니다.

- 1 프로그램 편집기에서 프로그램을 열고 도구 모음에서 **디버거**를 클릭합니다. 프로그램에서 DATA 스텝 코드의 모든 섹션은 여백의 녹색 막대로 강조 표시되어 디버깅이 가능하다는 것을 나타냅니다.

```

1 proc sort data=sashelp.cars out=cars;
2   by make;
3   run;
4
5 data horsepower;
6   set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
7   format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
8   retain runninghorses;
9   by make;
10
11   if _n_ eq 1 then
12     runningHorses=0;
13   else runninghorses=runninghorses+horsepower;
14   runningaveragehorsepower=runningHorses/_n_;
15   dollarsPerHorse=invoice/Horsepower;
16   run;
17
18 proc print;
19   run;

```

- 2 디버깅하려는 코드 섹션 옆의 여백에서 을 클릭합니다. DATA 스텝 코드가 DATA 스텝 디버거 창에서 열립니다.

주: DATA 스텝 코드를 마우스 오른쪽 버튼으로 클릭하고 **DATA 스텝 디버거**를 선택해서도 데이터 스텝 디버거 창을 열 수 있습니다.

DATA 스텝 디버거 닫기

▶ 계속 ■ 중지

```

1 data horsepower;
2 set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
3 format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
4 retain runninghorses;
5 by make;
6
7 if _n_ eq 1 then
8   runningHorses=0;
9 else runninghorses=runninghorses+horsepower;
10 runningaveragehorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;

```

필터

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make		<input type="checkbox"/>
Model		<input type="checkbox"/>
Type		<input type="checkbox"/>
Origin		<input type="checkbox"/>
DriveTrain		<input type="checkbox"/>
MSRP	.	<input type="checkbox"/>
Invoice	.	<input type="checkbox"/>
EngineSize	.	<input type="checkbox"/>

콘솔

DATA 스텝 디버거가 시작되었습니다.
정지 : 행 2 칼럼 2

> 명령 입력

중단점 설정 및 지우기

프로그램의 모든 실행 가능한 행에서 중단점을 설정하거나 지울 수 있습니다. 중단점은 지정된 행에서 프로그램 실행을 일시 중지합니다. 중단점을 설정하거나 지울 때마다 중단점에 대한 정보가 창 하단의 디버그 콘솔에 표시됩니다.

중단점을 설정하거나 지우려면 다음을 수행합니다.

- 행을 선택하고 도구 모음의 ●을 클릭하여 중단점을 켜거나 끌 수 있습니다.
- 프로그램의 모든 중단점을 지우려면 ✖을 클릭합니다.

DATA 스텝 디버거 닫기

▶ 계속 ■ 중지 | 🔄 | ● ✖

```

1  data horsepower;
2  set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
3  format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
4  retain runninghorses;
5  by make;
6
7  if n_eq 1 then
8      runningHorses=0;
9  else runninghorses=runninghorses+horsepower;
10 runningaveragehorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;
  
```

콘솔

DATA 스텝 디버거가 시작되었습니다.
 정지 : 행 2 칼럼 2
 DEBUG> break 7;
 중단점 1 설정 : 행 7

> 명령 입력

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make		<input type="checkbox"/>
Model		<input type="checkbox"/>
Type		<input type="checkbox"/>
Origin		<input type="checkbox"/>
DriveTrain		<input type="checkbox"/>
MSRP	.	<input type="checkbox"/>
Invoice	.	<input type="checkbox"/>
EngineSize	.	<input type="checkbox"/>

프로그램을 스텝별로 실행하기

프로그램의 문장을 한 번에 하나씩 실행하고 각 스텝에서 변수 값을 볼 수 있습니다. 프로그램의 스텝을 진행할 때 변경되는 변수 값은 빨간색으로 표시됩니다.

실행이 일시 중지된 지점부터 시작하여 문장을 실행하려면 🔄을 클릭합니다. 한 번에 하나의 문장만 실행됩니다.

프로그램의 실행 시작 및 중지

DATA 스텝 디버거에서 프로그램의 실행을 시작하면 프로그램은 중단점에 도달할 때까지 또는 관찰 중인 변수의 값이 변경될 때까지 실행됩니다. 중단점 또는 관찰되는 변수가 지정되지 않았다면 프로그램은 완료될 때까지 실행됩니다.

- 프로그램의 실행을 시작하거나 계속하려면 ▶을 클릭합니다.
- 프로그램의 실행을 중지하려면 ■을 클릭합니다.

주: 프로그램에서 적절한 행을 마우스 오른쪽 버튼으로 클릭하고 **건너뛰기**를 선택하여 실행을 다시 시작할 프로그램 행을 지정할 수 있습니다.

변수 값 검사

프로그램을 디버깅할 때 프로그램에서 각 변수의 값을 볼 수 있습니다. 변수 영역에서 전체 변수 리스트를 볼 수 있으며 필터 및 검색 상자를 사용하여 변수 리스트 범위를 좁힐 수 있습니다. 변경된 변수 값은 빨간색으로 표시됩니다.

DATA 스텝 디버거 닫기

▶ 계속 ■ 중지
🔍
⌂

```

1 data horsepower;
2 set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
3 format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
4 retain runningHorses;
5 by make;
6
7 if _n_ eq 1 then
8   runningHorses=0;
9 else runningHorses=runningHorses+horsepower;
10 runningAveragehorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;

```

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make	Audi	<input type="checkbox"/>
Model	A8 L Quattro 4dr	<input type="checkbox"/>
Type	Sedan	<input type="checkbox"/>
Origin	Europe	<input type="checkbox"/>
DriveTrain	All	<input type="checkbox"/>
MSRP	\$69,190	<input type="checkbox"/>
Invoice	\$64,740	<input type="checkbox"/>
EngineSize	4.2	<input type="checkbox"/>

콘솔

```

이동 : 행 12 칼럼 1
DEBUG> step;
이동 : 행 2 칼럼 2
DEBUG> step;
이동 : 행 7 칼럼 2

```

> 명령 입력

변수 값 관찰

프로그램의 모든 변수를 관찰할 수 있습니다. SAS는 관찰 중인 변수의 값이 변경될 때마다 프로그램의 실행을 일시 중지합니다. 변경된 변수 값은 변수 영역에서 빨간색으로 표시됩니다.

변수를 관찰하려면 변수 영역에서 변수의 체크 박스를 선택합니다.

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make	Cadillac	<input type="checkbox"/>
Model	Escalade	<input type="checkbox"/>
Type	SUV	<input type="checkbox"/>
Origin	USA	<input type="checkbox"/>
DriveTrain	Front	<input checked="" type="checkbox"/>
MSRP	\$52,795	<input type="checkbox"/>
Invoice	\$48,377	<input type="checkbox"/>
EngineSize	5.3	<input type="checkbox"/>
Cylinders	8	<input type="checkbox"/>
Horsepower	295	<input type="checkbox"/>
MPG_City	14	<input type="checkbox"/>
MPG_Highway	18	<input type="checkbox"/>

명령행으로 디버깅

명령행을 사용하여 디버거 명령을 직접 입력할 수 있습니다. 올바른 명령 리스트를 보려면 ⓘ을 클릭합니다.

주: DATA 스텝 디버거 명령에 대한 자세한 [SAS Code Debugger: User's Guide](#)의 “Debugger Commands Overview”에서 참조하십시오.

예를 들어 명령행을 사용하여 조건부 중단점을 설정할 수 있습니다. 그러면 지정된 조건이 충족되면 프로그램 실행이 중지됩니다. 다음 예에서는 이 디버거 명령을 사용하여 변수 `_n_`의 값이 10일 때 행 11에서 중단점을 설정합니다.

```
break 11 when _n_ eq 10
```

```

콘솔

DATA 스텝 디버거가 시작되었습니다.
정지 : 행 2 칼럼 2

> break 11 when _n_ eq 10;

```

중단점이 행 11에 설정되었으며 `_n_`의 값이 **10**이 되면 프로그램 실행이 중지됩니다.

DATA 스텝 디버거

▶ 계속 ■ 중지
🔍
⌂
⌵

```

1 data horsepower;
2 set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
3 format dollarsPerHorse dollar12.2 runningAveragehorsepower 6.1;
4 retain runninghorses;
5 by make;
6
7 if _n_ eq 1 then
8   runningHorses=0;
9 else runninghorses=runninghorses+horsepower;
10 runningaveragehorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;

```

변수	값	관찰
numobs	428	<input type="checkbox"/>
Make	Cadillac	<input type="checkbox"/>
Model	Escalade	<input type="checkbox"/>
Type	SUV	<input type="checkbox"/>
Origin	USA	<input type="checkbox"/>
DriveTrain	Front	<input type="checkbox"/>
MSRP	\$52,795	<input type="checkbox"/>
Invoice	\$48,377	<input type="checkbox"/>
EngineSize	5.3	<input type="checkbox"/>
Cylinders	8	<input type="checkbox"/>
Horsepower	295	<input type="checkbox"/>
MPG_City	14	<input type="checkbox"/>
MPG_Highway	18	<input type="checkbox"/>

콘솔

```

DATA 스텝 디버거가 시작되었습니다.
정지 : 행 2 칼럼 2
DEBUG> break 11 when _n_ eq 10;
중단점 1 설정 : 행 11
DEBUG> go;
중단 : 행 11 칼럼 2

```

> 명령 입력

Snippet 작업

Snippet을 사용하는 이유

Snippet은 공통으로 사용되는 코드 또는 텍스트 행이며, 저장해서 다시 사용할 수 있습니다. Snippet은 SAS 코드, XML 코드 및 텍스트를 포함할 수 있습니다. 프로그램 편집기, XML 편집기, 텍스트 편집기, 플로우, 작업 정의, 작업 품 및 작업 프롬프트를 포함한 위치에 Snippet을 삽입할 수 있습니다. SAS Studio에서는 여러 코드 Snippet이 제공됩니다.

Snippet 이름	설명
표준 > 데이터	
DS2 코드	DS2 프로그램에 대한 템플릿을 제공합니다. DS2는 고급 데이터 조작에 적절한 SAS 프로그래밍 언어입니다. DS2는 Base SAS와 함께 제공되며 핵심 기능은 SAS DATA 스텝과 같습니다. 하지만 DS2는 그 외에도 변수 범위 지정, 사용자 정의 메소드, ANSI SQL 데이터 유형 및 사용자 정의 패키지가 포함되어 있어 DATA 스텝보다 뛰어납니다. DS2 SET 문에는 FedSQL 구문이 포함될 수 있으며 런타임에 생성되는 질의는 DS2와 지원되는 데이터베이스 간에 데이터를 대화식으로 교환할 수 있습니다. 그러면 입력 테이블의 SQL 사전 처리가 가능하므로 두 언어의 기능이 효과적으로 결합됩니다. 자세한 내용은 <i>SAS DS2 Language Reference</i> 에서 참조하십시오.
DS2 패키지	DS2 패키지에 대한 템플릿을 제공합니다. 패키지는 DS2 프로그램과 유사합니다. 패키지 본문은 일련의 글로벌 선언과 메소드 리스트로 구성됩니다. 구문상 주요 차이점은 PACKAGE 및 ENDPACKAGE 문입니다. 이러한 문장은 글로벌 범위를 가진 블록을 정의합니다. 자세한 내용은 <i>SAS DS2 Language Reference</i> 에서 참조하십시오.
DS2 스레드	DS2 스레드 프로그램에 대한 템플릿을 제공합니다. 일반적으로 DS2 코드는 순차적으로 실행됩니다. 즉, 한 프로세스의 실행이 완료되어야 다음 프로세스가 시작됩니다. 스레드 처리를 사용하면 두 개 이상의 프로세스를 동시에 실행할 수 있습니다. 스레드 처리에서는 각 프로그램이 코드 섹션을 동시에 실행하는 것을 스레드에서 실행한다고 합니다. 자세한 내용은 <i>SAS DS2 Language Reference</i> 에서 참조하십시오.
CSV 파일 생성	FILENAME 문에 지정된 SAS 데이터셋의 CSV 버전을 생성합니다. 파일 이름은 전체 이름이어야 하며 .csv로 끝납니다.
PowerPoint 슬라이드 생성	PROC SGPLOT을 사용하여 FILENAME 문에 지정된 파일에서 Microsoft PowerPoint 출력을 생성합니다. 파일 이름은 전체 이름이어야 하며 .ppt로 끝납니다.
XML 파일 생성	FILENAME 문에 지정된 SAS 데이터셋의 XML 버전을 생성합니다. 디렉터리 이름은 전체 이름이어야 합니다.
CSV 파일 가져오기	FILENAME 문에 지정된 위치에서 심표로 구분된 파일을 가져오고 SAS 데이터셋에 출력을 씁니다.
XLSX 파일 가져오기	FILENAME 문에 지정된 위치에서 XLSX 파일을 가져오고 SAS 데이터셋에 출력을 씁니다.
선형 회귀 데이터 시뮬레이션	선형 회귀 분석에 사용할 수 있는 입력 데이터 소스를 생성합니다. 선형 회귀 분석에서는 최소제곱법을 사용하여 데이터에 선형함수를 할당하려고 시도합니다.

Snippet 이름	설명
일원분산분석 데이터 시뮬레이션	세 가지 처리 수준을 가진 하나의 처리 요인을 다루는 입력 데이터 소스를 생성합니다. 일원분산분석 작업을 사용하여 이 데이터를 분석하는 목적은 각 수준 간의 평균 차이를 검정하고 이러한 차이 값을 구하는 것입니다.
표준> 데이터 품질	
주: 데이터 품질 Snippet을 사용하려면 데이터 준비가 필요합니다. 자세한 내용은 SAS Data Quality: Getting Started 에서 참조하십시오.	
스키마 적용	DQSCHEMEAPPLY 함수, DQSCHEMEAPPLY CALL 루틴 또는 DQSCHEME 프로시저를 사용하여 SAS 출력형식으로 저장된 스키마를 적용하는 방법을 보여줍니다. 스키마를 사용하면 데이터의 불규칙한 엔트리를 표준화하여 이 데이터를 리포트에 사용할 수 있습니다.
매치코드 계산	DQMATCH 함수를 사용하여 문자 값에서 매치 코드를 생성합니다. 매치 코드는 데이터 값의 상대적 유사성을 반영합니다. 매치 코드는 지정된 로케일로 지정된 매치 정의를 기반으로 합니다. 매치 코드는 출력 SAS 데이터셋에 작성됩니다. 동일한 매치 코드를 생성하는 값은 변환 또는 표준화에 대한 후보 값입니다.
대소문자 데이터	DQCASE 함수를 사용하여 지정된 대/소문자 정의에 따라 문자 상수를 변환하고 문자 값을 표준 대문자로 반환합니다.
클러스터(엔터티 분석)- Proc DQMATCH	DQMATCH 프로시저를 실행하여 일치 코드를 생성하고 여러 기준 문장으로 클러스터 번호를 생성합니다. PROC DQMATCH는 표준화 또는 변환을 위한 기반으로 일치 코드를 생성합니다. 매치 코드는 데이터 값의 상대적 유사성을 반영합니다. 매치 코드는 지정된 로케일로 지정된 매치 정의를 기반으로 생성됩니다. 매치 코드는 출력 SAS 데이터셋에 작성됩니다. 동일한 매치 코드를 생성하는 값은 변환 또는 표준화에 대한 후보 값입니다.
데이터 추출	DQEXTRACT 및 DQEXTTOKENGET 함수를 사용합니다. DQEXTRACT 함수는 추출 정의에 의해 감지된 입력 문자 값에서 추출 토큰 값의 구분자로 구분된 문자열을 반환합니다. DQEXTTOKENGET 함수는 추출 토큰 값의 구분자로 구분된 문자열에서 추출 토큰 값을 반환합니다.
로케일 추측	DQLOCALEQUSS 함수를 사용하여 DQLOCALE= 시스템 옵션 또는 세션 옵션에 나열된 로케일 추측 정의에 입력 문자 값을 적용합니다. 함수는 로케일의 이름을 반환합니다.
성별 식별	DQGENDER 함수를 사용하여 개인의 이름에서 성별을 반환합니다.
데이터 파싱	DQPARSE 함수를 사용하여 파싱 토큰 값의 구분자로 구분된 문자열을 반환하고 QPARSETOKENGET 함수를 사용하

Snippet 이름	설명
	여 파싱 토큰 값의 구분자로 구분된 문자열에서 파싱 토큰 값을 반환합니다.
식별 분석 수행	DQIDENTIFYINFOGET, DQIDENTIFYMULTI 및 DQIDENTIFYIDGET 함수를 사용하여 식별 분석을 수행하는 방법을 보여줍니다. <ul style="list-style-type: none"> ■ DQIDENTIFYINFOGET 함수는 지정된 식별 분석 정의에서 지원하는 이름 리스트를 반환합니다. ■ DQIDENTIFYMULTI 함수는 문자 값의 모든 식별 분석을 반환합니다. ■ DQIDENTIFYIDGET 함수는 ID 이름 리스트를 반환합니다.
데이터 프로파일링 실행- Proc DATAMETRICS	IDENTITIES 문에 지정된 MULTIIDENTITY= 인수가 있거나 없는 DATAMETRICS 프로시저를 사용하여 데이터 프로파일링을 데이터셋에 적용하는 방법을 보여줍니다.
데이터 표준화	DQSTANDARDIZE 함수를 사용하여 입력 문자 값의 대/소 문자 구분, 간격 및 출력형식을 표준화하고 업데이트된 문자 값을 반환합니다.
표준 > 기술형	
사용자 정의 ODS 출력	SAS ODS(Output Delivery System)를 사용하여 HTML, PDF 및 RTF 출력 생성을 위한 템플릿을 제공합니다. 자세한 내용은 <i>SAS Output Delivery System: User's Guide</i> 에서 참조하십시오.
PROC SQL	SQL 질의 작성을 위한 템플릿을 제공합니다. 자세한 내용은 <i>SAS SQL Procedure User's Guide</i> 에서 참조하십시오.
표준 > 그래프	
주: SGPLOT, SGPANEL 및 SGSCATTER 프로시저에 대한 자세한 내용은 <i>SAS ODS Graphics: Procedures Guide</i> 에서 확인하십시오.	
막대 패널	SGPANEL 프로시저에서 VBAR 문을 사용하고 여러 막대 그래프를 생성할 수 있습니다.
상자 패널	SGPANEL 프로시저에서 VBOX 문을 사용하고 여러 박스 그래프를 생성할 수 있습니다.
비교 산점도	SGSCATTER 프로시저에서 COMPARE 문을 사용합니다. 이 코드 Snippet은 축을 공유하는 산점도 비교 패널을 생성합니다.
점 도표	SGPLOT 프로시저에서 DOT 문을 사용합니다. 점 도표에서는 범주변수의 값이 수평으로 요약됩니다. 기본적으로 각 점은 각 범주변수 값의 빈도를 나타냅니다.

Snippet 이름	설명
적합 도표	SGPLOT 프로시저에서 REG 문을 사용합니다. 이 코드 Snippet은 이차 적합을 사용한 회귀 도표를 생성하며, 신뢰 한계를 포함합니다.
HBar 도표	SGPLOT 프로시저에서 HBAR 문을 사용합니다. 이 코드 Snippet은 범주변수 값이 요약된 수평 막대 그래프를 생성합니다.
HighLow 도표	SGPLOT 프로시저에서 HIGHLOW 문을 사용합니다. High-Low 그래프는 한 변수의 여러 값이 다른 변수의 한 값과 어떤 관계에 있는지를 보여 줍니다. 일반적으로 수평 축의 각 변수 값마다 수직 축에 해당하는 값이 여러 개씩 있습니다.
히스토그램 도표	SGPLOT 프로시저에서 HISTOGRAM 문을 사용합니다. 이 코드 Snippet은 두 개의 밀도 도표가 있는 히스토그램을 생성합니다. 이 Snippet에서 한 밀도 도표는 정규 밀도함수 추정값을 사용하고 다른 밀도 도표는 커널 밀도함수 추정값을 사용합니다.
산점도 행렬	SGSCATTER 프로시저에서 MATRIX 문을 사용합니다. 이 코드 Snippet은 산점도 행렬을 생성합니다.
VBox 도표	SGPLOT 프로시저에서 VBOX 문을 사용합니다. 상자 도표에서는 데이터를 요약하고 중위수, 제1 사분위수/제3 사분위수, 최솟값/최댓값을 나타냅니다. 이 도표는 중심, 분포, 범위 및 이상치를 간단하게 보여 주는 빠른 시각적 요약을 제공합니다. SGPLOT 및 SGPPANEL 프로시저에는 수평 및 수직 상자 도표를 생성하기 위한 별도의 문장이 있습니다.
표준 > IML	
주: 이러한 Snippet은 SAS/IML에 라이선스가 있을 때만 사용할 수 있습니다.	
비선형 방정식의 근 계산	한 변수의 함수 근을 구할 수 있습니다. 함수의 근(또는 0)을 구하면 비선형 방정식의 해를 구할 수 있습니다.
최대우도를 사용한 적합	최대우도 추정을 사용하여 정규 밀도함수의 모수값을 추정합니다.
부트스트랩 분포 생성	IML 프로시저를 사용하여 표본 평균의 부트스트랩 분포를 생성하고 분석합니다.
함수 적분	SAS/IML 소프트웨어의 QUAD 서브루틴을 사용하여 1차 함수를 적분할 수 있습니다. QUAD 서브루틴은 유한, 반무한 또는 무한 영역에서 함수의 정적분 값을 구하는 데 사용합니다.
다변량 정규 데이터 시뮬레이션	지정된 평균과 공분산으로 다변량 정규분포의 데이터를 시뮬레이션합니다.
표준 > 매크로	

Snippet 이름	설명
주: SAS 매크로에 대한 자세한 내용은 <i>SAS Macro Language: Reference</i> 에서 참조하십시오.	
SAS 매크로	SAS 매크로 작업을 위한 기본 템플릿을 제공합니다. 매크로에서는 프로그램의 텍스트 대체 등을 포함한 많은 작업을 수행할 수 있습니다. SAS 프로그램은 매크로 수를 몇 개든 포함할 수 있고 하나의 프로그램에서 매크로를 여러 번 불러올 수 있습니다. 자세한 내용은 <i>SAS Macro Language: Reference</i> 에서 참조하십시오.
SAS 매크로 문자 함수	<p>문자값으로 작업하는 SAS 매크로의 다양한 예를 제공합니다.</p> <ul style="list-style-type: none"> ■ %EVAL 함수는 정수 산술을 사용하여 산술 및 논리 표현식을 계산합니다. 이 함수는 함수의 인수를 문자값에서 숫자 또는 논리 표현식으로 변환합니다. 표현식을 계산한 후 결과는 문자값으로 변환됩니다. <p>SAS 매크로 기능은 기본적으로 텍스트 생성이기 때문에 이 기능이 유용합니다. 결과적으로 산술 표현식은 숫자 표현식으로 처음 변환됩니다. 숫자 표현식을 계산한 후 산술 표현식으로 다시 변환됩니다.</p> <ul style="list-style-type: none"> ■ %INDEX 함수는 문자열의 첫 번째 문자의 위치를 반환합니다. ■ %LENGTH 함수는 문자의 길이를 반환합니다. ■ %SCAN 함수는 문자열에서 위치로 지정된 단어에 대해 검색합니다. ■ %SUBSTR 함수는 문자열의 하위 문자열을 생성합니다. ■ %UPCASE 함수는 값을 대문자로 변환합니다.
SAS 매크로 DO 문	<p>해당하는 %END 문이 나타날 때까지 한 단위로 처리되는 매크로 정의 섹션의 시작을 지정합니다. 이 매크로 섹션은 %DO 그룹이라고 합니다.</p> <p>간단한 %DO 문은 %IF 조건이 True 또는 False인지 여부에 따라 처리될 매크로 섹션을 지정하기 위한 %IF-%THEN-%ELSE 문과 함께 나타납니다.</p> <p>주: SAS에서는 이 Snippet에 의해 생성된 코드와 다른 %DO 반복 문을 제공합니다. 자세한 내용은 <i>SAS Macro Language: Reference</i>에서 참조하십시오.</p>
SAS 매크로 IF 문	매크로 일부를 조건부로 처리합니다. %IF-%THEN-%ELSE 문에 대한 조건인 표현식은 상수 텍스트 또는 텍스트를 생성하는 텍스트 표현식인 피연산자만 포함할 수 있습니다.
SAS 매크로 문장	매크로를 불러올 때 지정한 값을 가진 하나 이상의 로컬 매크로 변수의 이름을 지정합니다. 매크로 변수에는 위치 및 키워드라는 두 가지 유형이 있습니다. 파라미터는 파라미터를 정의하는 매크로로 한정됩니다. 각 파라미터의 이름을 제공해야 합니다. 텍스트 표현식을 사용하여 파라미터를 생성할 수 없습니다. 파라미터 리스트는 쉼표로 구분된 매크로 파라미터를 몇 개라도 포함할 수 있습니다. 파라미터 리스트의 매크로 변수는 대개 매크로를 참조합니다.

Snippet 이름	설명
SAS 매크로 인용	<p>매크로 프로세서가 특수 문자 및 니모닉을 매크로 언어의 일부가 아니라 텍스트로 해석하도록 지정하는 매크로 함수의 예를 제공합니다.</p> <ul style="list-style-type: none"> ■ %STR 함수는 매크로 컴파일에서 상수 텍스트의 특수 문자 및 니모닉 연산자를 암호화합니다. 이 함수는 다음과 같은 특수 문자 및 니모닉 연산자를 암호화합니다. + - * / < > = ~ ^ ~ ; , # blank AND OR NOT EQ NE LE LT GE GT IN 이 함수는 문자가 쌍으로 발생할 때, 일치하지 않으면 앞에 %로 표시해서 다음과 같은 문자를 암호화합니다. ' ") ■ %NRSTR 함수는 매크로 컴파일에서 상수 텍스트의 특수 문자 및 니모닉 연산자를 암호화합니다. 이 함수는 %STR 함수에 대해 나열된 모든 특수 문자와 니모닉 연산자를 마스킹합니다. 또한 %NRSTR 함수는 & % 문자를 마스킹합니다. ■ %BQUOTE 함수는 매크로 실행에서 확인된 값의 특수 문자 및 니모닉 연산자를 암호화합니다. 이 함수는 다음과 같은 특수 문자 및 니모닉 연산자를 암호화합니다. ' ") + - * / < > = ~ ^ ~ ; , # blank AND OR NOT EQ NE LE LT GE GT IN ■ %SUPERQ 함수는 매크로 실행에서 모든 특수 문자 및 니모닉 연산자를 암호화하지만 이후 값을 계산하지 않습니다. 이 함수는 다음과 같은 특수 문자 및 니모닉 연산자를 암호화합니다. & % ' ") + - * / < > = ~ ^ ~ ; , # blank AND OR NOT EQ NE LE LT GE GT IN ■ %QSCAN 함수는 단어를 검색하고 특수 문자 및 니모닉 연산자를 암호화합니다. ■ %QSUBSTR 함수는 하위 문자열을 생성하고 특수 문자 및 니모닉 연산자를 암호화합니다. ■ %QUPCASE 함수는 값을 대문자로 변환하고 특수 문자 및 니모닉 연산자를 암호화하는 결과를 반환합니다. ■ %UNQUOTE 함수는 매크로를 실행하는 동안 값을 해독하여 특수 문자 및 니모닉 연산자를 텍스트 대신 매크로 언어 요소로 해석합니다. <p>매크로 컴파일 및 매크로 실행에 대한 자세한 내용은 <i>SAS Macro Language: Reference</i>에서 참조하십시오.</p>
SAS 매크로 변수	<p>사용자 정의 글로벌 및 로컬 매크로 변수를 생성하는 방법의 예를 제공합니다. 매크로 변수는 기호 치환을 통해 SAS 프로그램의 텍스트를 동적으로 수정할 수 있는 도구입니다. 크거나 작은 양의 텍스트를 매크로 변수에 할당할 수 있습니다. 그 후에는 텍스트를 포함하는 변수를 단순히 참조하는 텍스트를 사용할 수 있습니다.</p> <p>매크로 프로그래머에 의해 정의된 매크로 변수는 사용자 정의 매크로 변수라고 합니다. 매크로 프로세서에 의해 정의된 매크로 변수는 사용자 정의 매크로 변수라고 합니다. 데이터 행 내를 제외한 SAS 프로그램 아무 곳에서도 매크로 변수를 정의하고 사용할 수 있습니다.</p>

Snippet 이름	설명
Viya Foundation > Cloud Analytic Services	
CAS 연결 생성	CAS 서버에 대한 연결을 생성합니다. CASHOST= 및 CASPORT= 시스템 옵션 값을 지정해야 합니다. CAS 문은 기본 세션을 지정된 CAS 서버 및 CAS 포트에 연결합니다.
Caslib 삭제	지정된 caslib을 삭제합니다.
caslib에서 테이블 또는 파일 삭제	caslib에서 테이블이나 파일을 삭제합니다. 또한 In-Memory 테이블을 제거할 수 있습니다.
CAS 세션 연결 해제	이름이 mySession인 CAS 세션에서 연결을 해제합니다. 연결을 해제하기 전에 타임아웃(단위: 초) 값을 지정할 수 있습니다. 타임아웃이 만료되기 전에 세션에 다시 연결할 수 있습니다. 그렇지 않으면, 세션이 종료됩니다.
Caslib에 대한 SAS 라이브러리 생성	기본 CAS 세션을 생성하고 라이브러리 영역에 라이브러리 참조가 표시되도록 기존 caslib에 대한 SAS 라이브러리 참조를 생성합니다.
CAS 세션 옵션 리스트	지정된 CAS 세션에 대한 세션 옵션을 나열합니다.
SAS 클라이언트에 대한 CAS 세션 나열	SAS 클라이언트에 의해 생성되었거나 SAS 클라이언트에 의해 다시 연결된 모든 CAS 세션 및 세션 속성을 나열합니다.
사용자 ID에 대한 CAS 세션 나열	세션 이름과 연결된 사용자 ID에 대해 CAS 서버에 알려진 모든 CAS 세션을 나열합니다.
Caslib으로 데이터 로드	지정된 caslib에 테이블을 로드합니다. Snippet에는 로드된 데이터를 모든 활성 세션에서 사용할 수 있도록 하는 PROMOTE 옵션이 포함되어 있습니다.
새로운 CAS 세션	기존 CAS 서버 연결을 사용하여 이름이 mySession인 새로운 CAS 세션을 시작합니다. 새로운 세션을 시작할 때 사용할 caslib, 세션에 대한 타임아웃(초) 및 세션의 로케일을 지정할 수 있습니다.
경로에 대한 새로운 Caslib	지정된 경로(<i>filePath</i>) 및 세션(mySession)에 대해 caslib(myCaslib)을 생성합니다. SESSREF= 옵션을 제거하면 현재 세션에 대해 caslib이 생성되어 활성화됩니다. SUBDIRS 옵션은 myCaslib의 범위를 확대하여 <i>filePath</i> 의 하위 디렉터리까지 포함합니다. LIBNAME 문은 caslib에 대해 SAS libref를 생성합니다.
CAS 세션 다시 연결	이름이 mySession인 CAS 세션에 다시 연결합니다.
Caslib에 테이블 저장	<i>sourceCaslib</i> 에서 In-Memory 테이블(<i>table-name</i>)의 영구 사본을 생성합니다. In-Memory 테이블은 지정된 이름(<i>filename</i>)을 사용하여 타겟 caslib <i>targetCaslib</i> 와 연결된 데이터 소스에 저장됩니다.

Snippet 이름	설명
	주: 라이브러리 영역에 있는 라이브러리 참조를 마우스 오른쪽 버튼으로 클릭하고 속성을 선택하여 CAS 엔진 라이브러리 참조와 연결된 Caslib를 결정할 수 있습니다. 서버 세션 CASLIB 필드는 Caslib를 표시합니다.
CAS 세션 종료	이름이 mySession인 CAS 세션을 종료합니다. 다시 연결을 사용할 수 없습니다.
Viya Foundation > Image Processing	
이 Snippet을 사용하려면 SAS Visual Data Mining and Machine Learning 라이선스를 구입하고 이를 설치해야 합니다.	
색상 변환	하나 이상의 이미지에서 색 공간을 변환합니다. TYPE 파라미터에 대해 다음 값 중 하나를 지정할 수 있습니다: 'BGR2HSV', 'BGR2RGB', 'BGR2YUV', 'COLOR2GRAY', 'GRAY2COLOR', 'HSV2BGR', 'RGB2BGR' 또는 'YUV2BGR'. 입력은 변환할 이미지가 포함된 CAS 테이블이고 출력은 변환된 이미지가 포함된 CAS 테이블입니다.
이미지 표시	SAS Studio에서 컴퓨터에 저장한 이미지를 표시합니다. 이 Snippet은 이미지에 대한 참조를 포함하는 주석 데이터셋을 생성하고 PROC SGPLOT를 사용하여 이미지를 표시합니다. IMAGE = 변수를 사용하여 표시할 이미지의 전체 이름과 위치를 지정합니다. SAS 서버에서 이 위치에 액세스할 수 있어야 합니다.
이미지 로드	지정된 경로에서 이미지 액션셋을 로드하고 CAS 테이블을 생성합니다. 경로 파라미터는 로드할 이미지가 있는 디렉터리를 가리킵니다. 경로 파라미터는 이미지가 저장된 caslib 디렉터리의 하위 디렉터리입니다. 일반적인 워크플로에는 이미지를 로드하는 작업과 그 후 다른 Snippet 중 하나를 사용하여 이미지를 크기 조정 또는 재조정하거나 변형하는 방식으로 이미지를 처리하는 작업이 포함됩니다. 한 Snippet의 출력을 다른 Snippet의 입력으로 사용할 수 있습니다. 추가 처리를 위해 이미지 저장 Snippet을 사용하여 이미지를 컴퓨터에 저장할 수 있습니다.
이미지 변형	다른 증강 기술을 사용하여 하나 이상의 이미지를 변형합니다. TYPE 파라미터에 대해 다음 값 중 하나를 지정할 수 있습니다: 'COLOR_JITTERING', 'COLOR_SHIFTING', 'DARKEN', 'HORIZONTAL_FLIP', 'INVERT_PIXELS', 'LIGHTEN', 'ROTATE_LEFT', 'ROTATE_RIGHT', 'SHARPEN' 또는 'VERTICAL_FLIP'. 입력은 변경할 이미지가 포함된 CAS 테이블이고 출력은 변형된 이미지가 포함된 CAS 테이블입니다.
이미지 재조정	지정된 옵션에 따라 하나 이상의 이미지의 깊이를 변경합니다. TYPE 파라미터에 대해 다음 값 중 하나를 지정할 수 있습니다: "TO_8U", "TO_32F" 또는 "TO_64F". ALPHA 및 BETA 파라미터를 사용하여 값을 조정할 수도 있습니다. 입

Snippet 이름	설명
	력은 재조정할 이미지가 포함된 CAS 테이블이고 출력은 재조정된 이미지가 포함된 CAS 테이블입니다.
이미지 크기 조정	지정된 HEIGHT 및 WIDTH 파라미터에 따라 하나 이상의 이미지의 크기를 조정합니다. HEIGHT 파라미터는 행 수에 해당하고 WIDTH 파라미터는 칼럼 수에 해당합니다. 입력은 크기를 조정할 이미지가 포함된 CAS 테이블이고 출력은 크기가 조정된 이미지가 포함된 CAS 테이블입니다.
이미지 저장	지정된 이미지를 CASUSER와 같이 이전에 정의된 Caslib의 지정된 하위 디렉터리에 저장합니다. Caslib 루트 디렉터리에 하위 디렉터리의 이름을 지정해야 합니다. 저장된 이미지의 이름은 PREFIX 파라미터의 값으로 시작합니다.
Viya Foundation > Machine Learning	
이 Snippet을 사용하려면 SAS Visual Statistics 라이선스를 구입하고 이를 설치해야 합니다.	
여러 개의 ML 알고리즘 비교	Hmeq 데이터셋의 이진 타겟을 예측하는 데 여러 개의 Machine Learning 알고리즘을 적합하고 비교합니다. 해당 데이터셋은 Machine Learning 샘플 데이터 라이브러리에 포함되어 있습니다. 이 Snippet을 실행하기 전에 데이터 로드 및 데이터 준비 및 탐색을 실행해야 합니다.
두 개의 ML 알고리즘 비교	Hmeq 데이터셋의 이진 타겟을 예측하는 데 두 개의 Machine Learning 알고리즘을 적합하고 비교합니다. 해당 데이터셋은 Machine Learning 샘플 데이터 라이브러리에 포함되어 있습니다. 이 Snippet을 실행하기 전에 데이터 로드 및 데이터 준비 및 탐색을 실행해야 합니다.
일반화선형모델	GENSELECT 프로시저를 사용하여 일반화선형모델 적합 및 평가를 나타냅니다. 이 Snippet을 실행하기 전에 데이터 로드 및 데이터 준비 및 탐색을 실행해야 합니다.
데이터 로드	로컬 데이터를 CAS에 로드하는 방법을 나타냅니다.
데이터 준비 및 탐색	모델링 전에 데이터를 수정하고 준비합니다. 이 Snippet은 Machine Learning 샘플 데이터 라이브러리에서 Hmeq 데이터셋을 사용합니다. Hmeq 데이터셋은 입력으로 사용되며 Hmeq_prepped 데이터셋을 생성합니다. Hmeq_prepped 데이터셋은 후속 예에 사용됩니다. 주: 이 Snippet을 실행할 때 결측값에 대한 경고 메시지를 볼 수 있습니다. 표본 데이터는 다른 프로시저에서 사용될 수 있는 일부 결측값을 포함합니다.
지도 학습	데이터 탐색 및 준비, 모델링 및 평가를 포함하는 전체 워크플로 프로세스를 보여 줍니다. 이 Snippet은 Sampleml.Hmeq 데이터셋을 사용하며 Machine Learning 샘플 데이터 라이브러리에 포함되어 있습니다.

Snippet 이름	설명
비지도 학습	데이터 준비, 분석 및 결과 시각화를 포함하는 전체 워크플로 프로세스를 보여 줍니다. 이 Snippet은 Sashelp.Iris 데이터셋을 사용합니다.

Snippet 생성

사용자 고유의 Snippet을 생성하려면 다음을 수행합니다.

- 1 SAS Studio에서 파일을 열고 Snippet으로 저장할 코드 또는 텍스트의 행을 선택합니다. 코드 또는 텍스트의 행을 선택하지 않으면 전체 파일이 Snippet으로 저장됩니다.
- 2 도구 모음에서 **내 Snippet에 복사**를 클릭합니다.

주: Snippet을 탐색 영역의 **탐색기** 섹션으로 복사할 수도 있습니다. 탐색 영역의 **Snippet** 섹션에서 기존 Snippet을 마우스 오른쪽 버튼으로 클릭하고 **Snippet을 다음으로 복사** ⇒ **탐색기**를 선택합니다. Snippet은 지정한 폴더에 *.csm 파일로 저장됩니다.

- 3 Snippet의 이름을 입력합니다.
- 4 프로그램 편집기에서 @*snippet-abbreviation*을 입력하여 Snippet에 빠르게 액세스하려면 Snippet의 축약어를 입력합니다. 설명을 추가할 수도 있습니다.
- 5 Snippet을 저장할 폴더를 선택하고 **확인**을 클릭합니다. 지정한 폴더에 Snippet이 추가됩니다.

주: 기존 Snippet 파일을 **내 Snippet** 폴더 및 하위 폴더로 업로드할 수도 있습니다. 파일 유형은 *.csm이어야 합니다. 사용할 폴더를 선택하고 **T**을 클릭합니다. **+**을 클릭하고 파일을 찾은 다음 **업로드**를 클릭합니다.

저장된 *.sas 파일을 **내 Snippet** 폴더에 추가하려면 탐색 영역의 **탐색기** 섹션에서 .sas 파일을 마우스 오른쪽 버튼으로 클릭하고 **내 Snippet으로 복사**를 선택합니다.

코드 Snippet을 삽입하는 방법

Snippet 삽입하려면 다음을 수행합니다.

- 1 Snippet을 삽입할 위치를 클릭합니다.
- 2 탐색 영역에서 **📄**을 클릭하여 Snippet 섹션을 엽니다.
- 3 다음과 같은 방법으로 파일에 Snippet을 추가할 수 있습니다.
 - 끌어서 놓기 작업을 사용하여 Snippet을 파일에 삽입합니다.
 - Snippet 이름을 마우스 오른쪽 버튼으로 클릭하고 **코드에 삽입**을 선택합니다.

- 파일을 마우스 오른쪽 버튼으로 클릭하고 **Snippet 삽입**을 선택한 다음 사용할 Snippet을 선택합니다.
- 코드에 `@snippet-name`을 입력하고 Enter을 누릅니다. @를 입력하고 팝업 창을 사용하여 내 Snippet 폴더에 있는 Snippet의 리스트에서 Snippet을 선택할 수도 있습니다.

주: 이 기능을 사용하려면 Snippet에 대한 축약어를 지정해야 합니다. Snippet에 축약어가 있는지 확인하려면 Snippet을 선택하고 **⌘**를 클릭하십시오. 축약어를 추가하거나 편집하려면 Snippet을 마우스 오른쪽 버튼으로 클릭하고 **편집**을 선택합니다. 자세한 내용은 [“Snippet 생성” \(72페이지\)](#)에서 확인하십시오. 또한 **Snippet 축약어 사용** 우선 설정을 선택해야 합니다. 자세한 내용은 [“편집기 우선 설정 지정” \(179페이지\)](#)에서 확인하십시오.

코드 편집기 사용자 정의

우선 설정 창에서 자동 완성 및 색상 코딩을 비롯한 코드 편집기 기능에 적용되는 여러 옵션을 변경할 수 있습니다.

편집기 옵션에 액세스하려면 **옵션** ⇒ **우선 설정**을 선택합니다. 그런 다음 **편집기**를 클릭합니다.

자세한 내용은 [“편집기 우선 설정 지정” \(179페이지\)](#)에서 참조하십시오.

질의 작업

질의란?	76
독립형 질의 및 플로우 질의의 차이점 이해	76
이전 버전의 SAS Studio에서 저장된 질의 마이그레이션	77
독립형 질의 생성	78
질의 생성	78
질의에 테이블 추가	79
조인 이해	79
테이블 조인	79
조인 생성	80
질의에서 테이블 속성 편집	81
조인 유형에 대한 이해	81
기존 조인 수정	82
데이터 선택	83
출력에서 칼럼 지정	83
계산된 칼럼 생성	86
요약 함수 사용	87
조건부 표현식 작업	88
데이터 필터링	89
필터 생성	89
필터 표현식 간 관계 변경	92
출력 관리	92
출력 정렬	92
출력에서 중복 행 제거	93
출력 그룹화	94
그룹화된 데이터 서브셋 지정	97
결과 저장	98
입력 처리에 대한 제한 지정	98
출력 처리에 대한 제한 지정	99
Pass-Through 및 테이블 옵션 지정	100
질의에 노트 추가	101
질의 실행	102
FedSQL 질의 생성	102

질의란?

질의를 사용하여 지정한 기준에 따라 하나 이상의 테이블에서 데이터를 추출할 수 있습니다. 테이블 한 개를 기반으로 하는 질의를 생성하거나 테이블을 조인할 수 있습니다. 질의를 생성할 때 SAS Studio는 보고 편집할 수 있는 SQL(Structured Query Language) 코드를 생성합니다. FEDSQL 프로시저를 사용하여 질의를 생성하도록 선택할 수도 있습니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.

주: 질의는 표준 영역에서만 사용할 수 있습니다. 자세한 내용은 [“영역 이해” \(31페이지\)](#)에서 확인하십시오.

독립형 질의 및 플로우 질의의 차이점 이해

개별적으로 실행할 수 있는 독립형 질의와 플로우의 일부로만 실행될 수 있는 플로우 내의 질의를 모두 생성할 수 있습니다. 플로우에서 질의를 실행하고 있으면 일부 질의 기능을 사용할 수 없습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “플로우란?”](#)에서 참조하십시오.

독립형 질의와 플로우 질의의 차이점은 다음 테이블에서 확인할 수 있습니다.

질의 기능	독립형 질의	플로우 질의
출력 옵션		
테이블	예	예
뷰	예. 질의가 PROC FEDSQL을 사용하지 않는 경우에 한 합니다.	아니요
리포트	예	아니요
라이브러리 및 테이블 이름을 지정하는 옵션	예	아니요. 라이브러리 및 테이블 이름을 지정하려면 플로우에 테이블 노드를 추가해야 합니다.
칼럼 영역		
테이블을 추가하고 제거하는 옵션	예	아니요

질의 기능	독립형 질의	플로우 질의
테이블 속성을 볼 수 있는 옵션	예	아니요. 테이블 별칭만 업데이트할 수 있습니다.
탭 선택		
칼럼에 대한 출력형식 지정 기능	예	예. 질의가 PROC FEDSQL을 사용하지 않는 경우에 한 합니다.
표현식 빌더		
칼럼 값 리스트를 표시하는 기능	예	아니요
구문을 검증하고 검증 로그를 보는 옵션	예	아니요
우선 설정		
질의에 우선 설정을 적용하는 기능	예	아니요
저장 옵션		
질의를 별도의 파일로 저장하는 기능	예. 질의는 * .cqy 파일로 저장될 수 있습니다.	아니요. 플로우의 일부로만 질의를 저장할 수 있습니다.

이전 버전의 SAS Studio에서 저장된 질의 마이그레이션

다음 기준이 충족되면 이전 버전의 SAS Studio에서 저장된 질의를 SAS Studio 2020.1에서 실행할 수 있습니다.

- 원래 질의에 사용된 데이터는 SAS Studio 2020.1 이상에서 사용할 수 있습니다.
- SAS Studio 2020.1 이상에서 사용할 수 있는 데이터에서는 원래 질의에 사용된 칼럼을 찾을 수 없습니다.
- 원래 질의에 포함된 모든 테이블에는 정의된 조인이 있어야 합니다.

주: SAS Studio 2020.1 이상에서 이전 버전의 SAS Studio에서 질의를 처음 저장하면 **다른 이름으로 저장** 창이 열립니다. 이전 버전과 동일한 이름으로 마이그레이션 질의를 저장하면 더 이상 이전 버전의 SAS Studio에서 질의를 열 수 없습니다.

저장된 질의를 마이그레이션할 수 없는 경우 SAS Studio는 질의를 프로그램 파일로 변환하고 질의를 마이그레이션할 수 없는 이유를 설명하는 메모를 프로그램에 포함시킵니다.

독립형 질의 생성

질의 생성

기본 SAS Studio 도구 모음에서 **새로 만들기** ⇨ **질의**를 선택합니다.

주: 탐색 영역의 **라이브러리** 섹션에서 하나 이상의 테이블을 마우스 오른쪽 버튼으로 클릭하여 **질의 생성**을 선택하여 질의를 생성할 수도 있습니다.

주: 자세한 내용은 *SAS Studio: 플로우 작업의 "질의"*에서 참조하십시오.

기본적으로 질의 탭에는 세 가지 주요 영역이 포함됩니다.

칼럼 영역	질의의 칼럼 및 테이블에 액세스할 수 있습니다.
질의 정의 영역	질의에서 칼럼을 지정 및 그룹화하고, 테이블 간에 조인을 생성하고, 질의에서 데이터를 필터링하고, 출력을 정렬하고, 출력 옵션을 지정할 수 있습니다.
코드, 로그 및 결과 영역	코드 탭에는 질의를 구성할 때 자동으로 생성된 코드가 표시됩니다. 로그 탭에는 질의를 실행할 때 질의 로그가 표시됩니다. 출력 유형이 테이블 또는 뷰 이고 PROC SQL을 사용하여 코드를 생성하면 질의를 실행할 때 이 영역에 출력 데이터 탭이 추가됩니다. 출력 유형이 리포트 이면 질의를 실행할 때 이 영역에 결과 탭이 추가됩니다.



질의에 테이블 추가

질의 탭의 칼럼 영역에서 **추가**를 클릭합니다. **테이블 선택** 창을 이용하여 사용할 라이브러리 및 하나 이상의 테이블을 찾은 다음 **확인**을 클릭합니다.

주: 탐색 영역의 **라이브러리** 섹션을 사용하여 질의에 테이블을 추가할 수도 있습니다. 탐색 영역의 을 클릭하여 **라이브러리** 섹션을 열고 적절한 라이브러리를 펼칩니다. 하나 이상의 테이블을 질의 탭의 **칼럼** 영역으로 끌어옵니다.

조인 이해

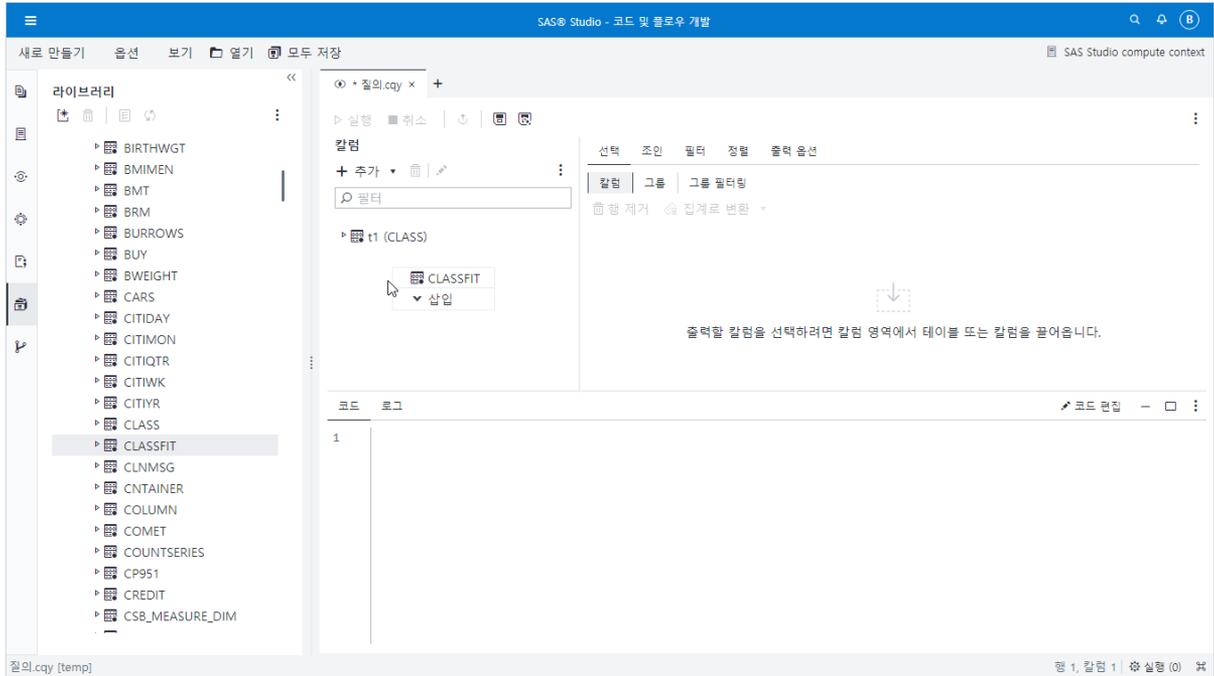
테이블 조인

질의를 생성하면 여러 테이블을 조인할 수 있습니다. SAS Studio는 자동으로 테이블을 조인하거나 사용자가 수동으로 조인을 생성할 수 있습니다. 두 테이블을 조인하면 SAS Studio는 이름과 유형이 같은 칼럼을 기준으로 테이블을 조인하려고 시도합니다. 칼럼 이름 및 유형에 대한 일치 항목이 발견되지 않으면 테이블이 각 테이블의 첫 번째 칼럼을 사용하여 조인됩니다. 질의에 대해 올바른 조인이 아니라면 조인을 편집하여 사용할 조인 기준을 지정할 수 있습니다.

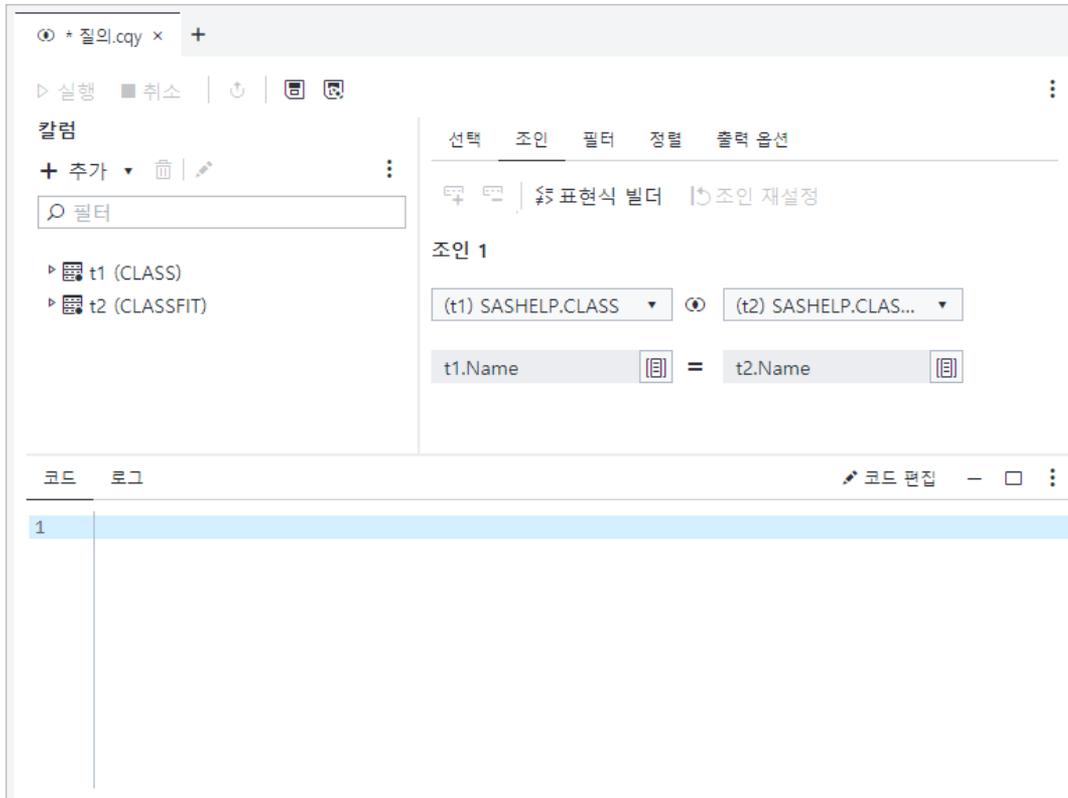
조인 생성

질의에서 기존 테이블에 테이블을 추가하면 SAS Studio에서 자동으로 조인을 생성합니다.

질의에 다른 테이블을 추가하려면 질의 탭의 칼럼 영역에서 **추가**를 클릭합니다. 테이블 선택 창을 사용하여 사용할 라이브러리 및 테이블을 찾은 다음 **확인**을 클릭합니다. 탐색 영역의 **라이브러리** 섹션에서 칼럼 영역으로 테이블을 끌어올 수도 있습니다.



조인 기준을 보려면 **조인** 탭을 클릭합니다. 다음 예에서 Classfit 테이블은 두 테이블에서 Name 칼럼을 사용하여 Classfit 테이블에 자동으로 조인됩니다.



알맞은 조인이 자동으로 생성되지 않으면 조인 조건을 수동으로 지정할 수 있습니다. 자세한 내용은 ["기존 조인 수정" \(82페이지\)](#)에서 확인하십시오.

질의에서 테이블 속성 편집

질의에서 테이블에 대한 별칭을 생성할 수 있습니다.

별칭 지정 방법 :

1. 질의 탭의 왼쪽 창에서 변경하려는 테이블을 마우스 오른쪽 버튼으로 클릭하고 **속성**을 선택합니다.

2. **별칭** 상자에 테이블의 URL을 입력합니다.

.....
 주: 별칭 이름은 질의 내에서 고유해야 합니다.

3. 속성 창에서 **확인**을 클릭하여 창을 닫고 변경 사항을 저장합니다. 별칭은 질의 탭의 테이블 리스트에 표시됩니다.

조인 유형에 대한 이해

SAS Studio는 조인에 대해 4가지 다른 유형을 지원합니다. 기존 조인을 수정하여 원하는 조인의 유형을 선택할 수 있습니다.

조인 창에서 사용하려는 조인 옵션을 선택할 수 있습니다.

SAS Studio 조인 유형	조인 아이콘	설명
이너 조인		출력 행은 칼럼 기준에 일치하는 첫 번째 테이블과 두 번째 테이블의 행만 포함합니다. 이너 조인이 기본값입니다.
왼쪽 조인		출력 행은 첫 번째 테이블의 모든 행과 두 번째 테이블에서 조인 조건과 일치하는 행을 포함합니다.
오른쪽 조인		출력 행은 두 번째 테이블의 모든 행과 조인 조건과 첫 번째 테이블에서 조인 조건과 일치하는 행을 포함합니다.
전체 조인		출력 행은 두 테이블에서 일치 및 일치하지 않는 모든 행을 포함합니다.

기본 조인 수정

조인의 다른 유형을 선택하거나 조인 조건에서 사용된 칼럼을 변경하여 기본 조인을 수정할 수 있습니다. 조인 조건을 추가 및 제거하거나 전체 조인을 제거할 수 있습니다.

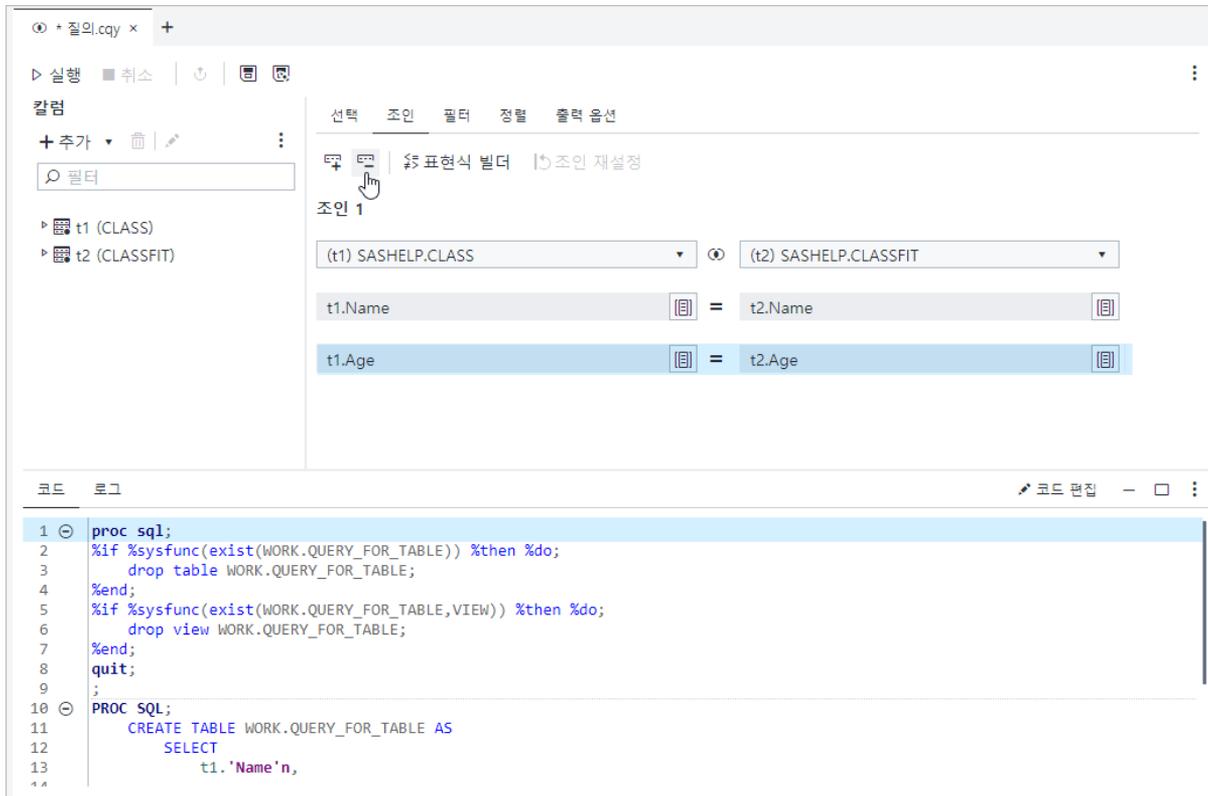
주: 질의에 추가하는 각 추가 테이블에 대해 최소 하나의 조인과 조인 조건이 있어야 합니다.

조인 수정하려면 다음을 수행합니다.

- 질의 창의 **조인** 탭을 클릭합니다. 다음과 같은 방법으로 조인을 편집할 수 있습니다.
 - 테이블 드롭다운 리스트를 사용하여 조인에 사용된 테이블을 변경할 수 있습니다.
 - 조인 유형을 편집하려면 테이블 이름 사이의 조인 지표를 클릭합니다. 기본 조인 유형은 **이너 조인**입니다. 자세한 내용은 [“조인 유형에 대한 이해” \(81페이지\)](#)에서 확인하십시오.
 - 새로운 연산자를 선택하려면 칼럼 이름 사이의 연산자를 클릭합니다. 기본 연산자는 =입니다.
 - 조인의 각 측면에서 을 클릭하여 조인에 사용되는 칼럼을 변경합니다.
- 새로운 조인 조건을 추가하려면 를 클릭합니다. 조인에 사용할 칼럼을 지정합니다. 조인 조건을 제거하려면 적절한 조인 조건을 선택한 다음 를 클릭합니다.

주: 다음 상황에서는 조인 조건을 삭제할 수 없습니다.

- 조인에는 하나의 조인 조건만 있습니다.
- 조인 조건은 조인의 첫 번째 조인 조건입니다. 그러나 첫 번째 조인 조건은 수정할 수 없습니다.



주: SQL 코드를 사용하여 조인을 편집하고 생성하려면 **조인** 탭에서 **표현식 빌더**를 클릭합니다. 자세한 내용은 **“표현식 구성” (187페이지)**에서 참조하십시오.

데이터 선택

출력에서 칼럼 지정

기본적으로 출력에서 포함하는 칼럼은 없습니다. 출력 테이블에 표시하려는 칼럼을 지정해야 합니다. 또한 출력 테이블에서 칼럼 이름 대신 사용할 별칭을 지정할 수도 있습니다.

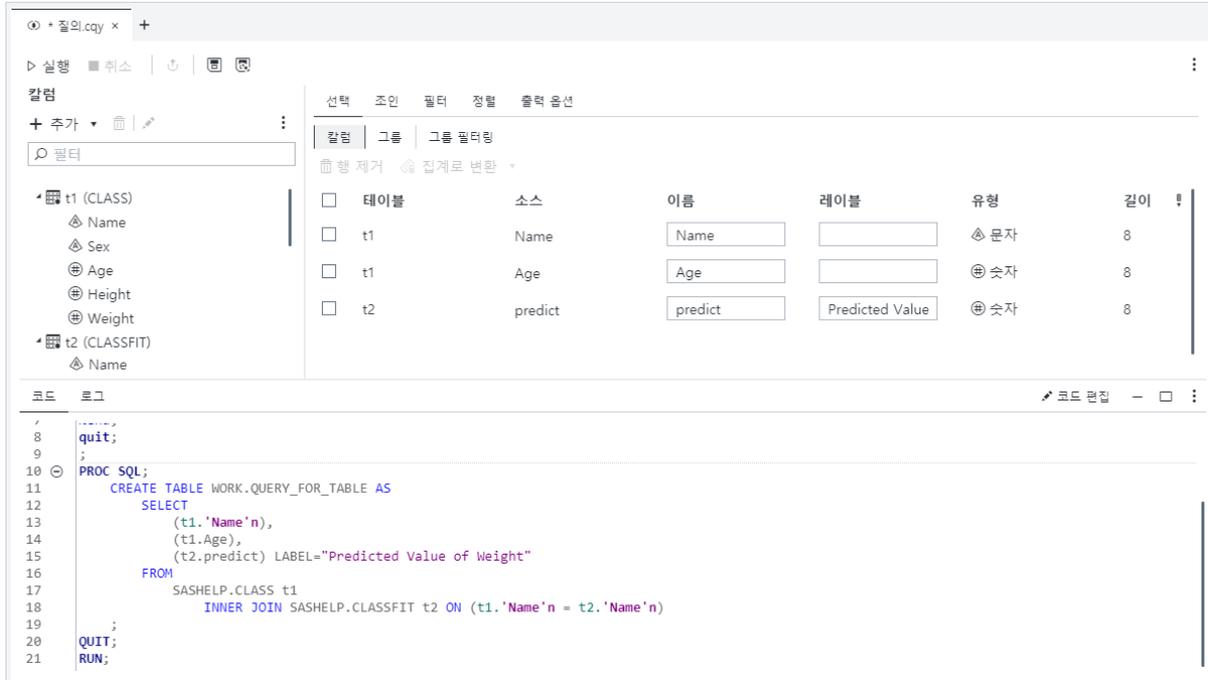
선택 탭에 나열된 칼럼의 순서는 출력 테이블에 나열된 순서와 같습니다. 칼럼을 마우스 오른쪽 버튼으로 클릭하고 알맞은 옵션을 선택하여 칼럼을 리스트의 위, 아래 및 맨 위 또는 맨 아래로 이동할 수 있습니다. 이동할 칼럼을 선택하고 칼럼 중 하나를 마우스 오른쪽 버튼으로 클릭한 다음 알맞은 옵션을 선택하여 리스트에서 이웃한 여러 칼럼을 이동할 수 있습니다.

주: 길이, 레이블 및 정보 칼럼은 FedSQL 질의를 생성하는 경우 **선택** 탭의 **칼럼** 하위 탭에서 사용할 수 없습니다. 플로우의 FedSQL 질의를 생성할 때에는 출력형식 칼럼을 사용할 수 없습니다. SQL 질의를 생성하는 경우, **☰**을 클릭하여 **칼럼** 하위 탭에 표시되는 칼럼을 관리할 수 있습니다. 칼럼 관리 창을 사용하여 칼럼을 숨기고 표시할 수 있습니다.

출력에 대해 개별적으로 칼럼을 선택하거나 다른 옵션을 사용하여 입력 테이블에서 모든 칼럼을 선택할 수 있습니다.

출력 테이블의 개별 칼럼을 선택하려면 다음을 수행합니다.

- 질의 탭의 칼럼 영역에서 테이블을 펼치고 출력에 포함할 칼럼을 선택 탭의 칼럼 하위 탭으로 끌어옵니다.



칼럼 이름을 사용하여 테이블에서 모든 칼럼을 선택하려면 다음을 수행합니다.

- 질의 탭의 칼럼 영역에서 테이블을 마우스 오른쪽 버튼으로 클릭하고 **칼럼 추가**를 선택합니다. 이 옵션은 모든 개별 칼럼 이름을 포함하는 SELECT 절을 생성합니다.

SELECT *table-alias.column-name1* <, *table-alias.column-name2*,...>

주: 테이블에 100개 이상의 칼럼이 있는 경우 처음 100개 칼럼만 SELECT 절에 추가됩니다. 나머지 칼럼을 추가하려면 **추가**를 클릭합니다. 나머지 칼럼을 선택하고 **확인**을 클릭합니다.

별표 (*) 표기를 사용하여 테이블에서 모든 칼럼을 선택하려면 다음을 수행합니다.

- 질의 탭의 칼럼 영역에서 테이블을 마우스 오른쪽 버튼으로 클릭하고 **모든 칼럼 선택**을 선택합니다. 이 옵션은 별표(*) 표기를 사용하여 테이블의 모든 칼럼을 나타내는 SELECT 절을 생성합니다.

SELECT *table-alias*.*

주: 이 옵션은 입력 테이블의 칼럼 수가 변경 될 수 있는 경우에 유용합니다.

별표(*) 표기를 사용하여 모든 테이블에서 모든 칼럼을 선택하려면 다음을 수행합니다.

- 질의 탭의 칼럼 영역에서 **모든 테이블에서 모든 칼럼 선택**을 선택합니다. 이 옵션은 별표(*) 표기를 사용하여 모든 칼럼과 모든 테이블을 나타내는 SELECT 절을 생성합니다.

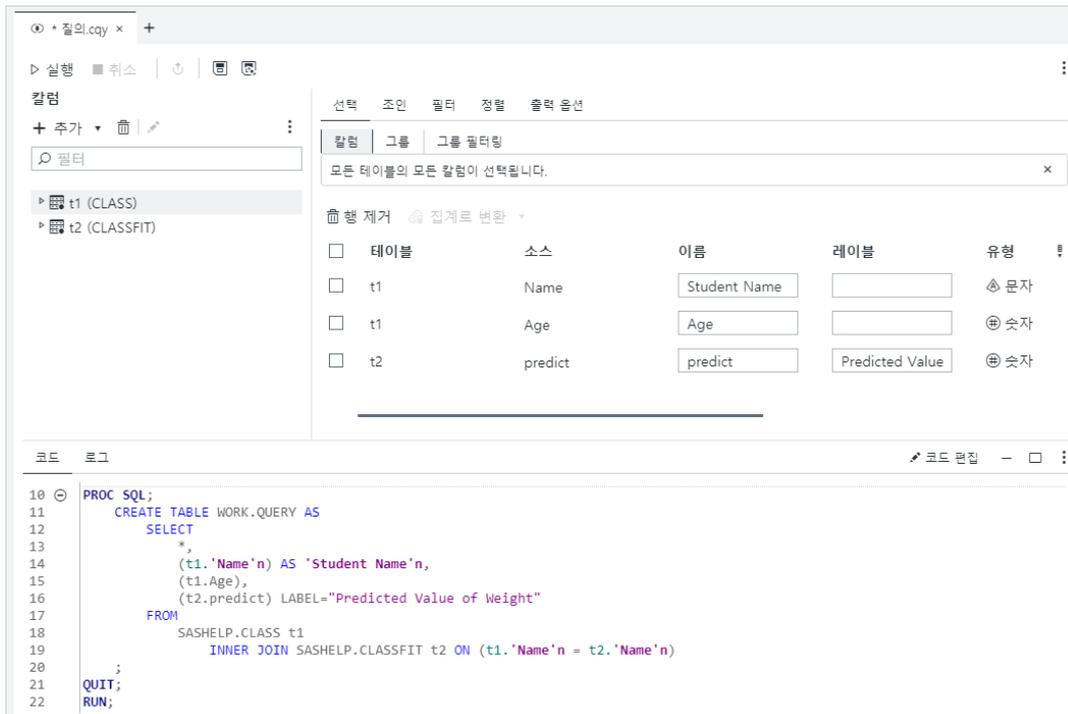
SELECT *

주: 이 옵션은 질의 테이블 수가 변경되거나 테이블의 칼럼 수가 변경될 때 유용합니다.



칼럼에 대한 별칭을 지정하려면 다음을 수행합니다.

- **선택** 탭에서 이름 칼럼을 사용하여 칼럼에 대해 사용할 별칭을 지정합니다. 별칭은 출력 데이터의 칼럼 머리글로 사용됩니다.



출력에서 칼럼을 제거하려면 다음을 수행합니다.

- **선택** 탭에서 제거할 칼럼에 대한 체크 박스를 선택하고 **행 제거**를 클릭합니다.

질의에서 모든 칼럼을 제거하려면 **선택** 탭에서 테이블에 대한 체크 박스를 선택하고 **행 제거**를 클릭합니다.

계산된 칼럼 생성

다른 칼럼 또는 값에서 계산된 새로운 칼럼을 질의에 삽입할 수 있습니다. 계산된 칼럼을 사용하여 값을 요약하고, 조건에 따라 값을 바꾸고, 문자열 및 숫자 계산을 수행할 수 있습니다. 계산 칼럼을 생성한 후 칼럼을 사용하여 출력을 필터링, 정렬 또는 그룹화할 수 있습니다.

계산된 칼럼을 생성하려면 다음을 수행합니다.

- 1 질의 탭의 칼럼 영역에서 **추가** ⇨ **계산된 칼럼**을 클릭합니다.
- 2 계산된 칼럼 창에서 표현식 빌더를 사용하여 칼럼을 생성합니다. 자세한 내용은 **“표현식 구성” (187페이지)**에서 참조하십시오.
- 3 칼럼에 대해 표현식을 생성한 다음 표현식 빌더 영역의 **속성** 탭을 사용하여 칼럼 이름을 지정합니다. 레이블, 데이터 유형, 길이, 출력형식 및 입력형식을 지정할 수도 있습니다.
- 4 계산된 칼럼을 선택 탭의 칼럼 리스트에 자동으로 추가하려면 **선택 탭에 새로운 계산 칼럼 추가**를 클릭합니다. 이 옵션은 계산된 칼럼을 생성할 때 기본적으로 선택됩니다.
- 5 **확인**을 클릭하여 계산된 칼럼을 생성합니다.

저장
취소

데이터 함수

필터

선택된 칼럼

- t1 (CARS)
 - Make
 - Model
 - Type
 - Origin
 - DriveTrain
 - MSRP
 - Invoice
 - EngineSize
 - Cylinders
 - Horsepower
 - MPG_City
 - MPG_Highway
 - Weight
 - Wheelbase
 - Length
- 계산된 칼럼

+ - * ÷ ** || | () ' " , % †

↶ ↷ ? | 검증

CASE WHEN THEN ELSE END | AND OR NOT | = ≠ < ≤ > :

1 t1.MSRP-t1.Invoice

속성 값 로그

칼럼 이름:

칼럼 레이블:

유형:

길이(바이트):

출력형식:

입력형식:

선택 탭에 새로운 계산 칼럼 추가

요약 함수 사용

질의의 칼럼에 대해 집계 또는 요약 함수를 실행할 수 있습니다. 칼럼을 집계하면 새롭게 계산된 칼럼이 자동으로 생성되어 질의에 추가됩니다. 집계 함수를 실행하려면 집계 기능을 실행할 칼럼을 선택합니다. **집계로 변환**을 클릭한 후 사용할 함수를 선택합니다. 다음 예제는 모든 학생의 평균 나이를 계산하는 계산된 칼럼으로 변환된 후의 Age 칼럼을 보여줍니다.

```

9 ;
10 PROC SQL;
11 CREATE TABLE WORK.QUERY AS
12 SELECT
13 (t1.'Name'n) AS 'Student Name',
14 (AVG(t1.Age)) LENGTH=8 AS AVG_Age,
15 (t2.predict) LABEL="Predicted Value of Weight"
16 FROM
17 SASHELP.CLASS t1
18 INNER JOIN SASHELP.CLASSFIT t2 ON (t1.'Name'n = t2.'Name'n)
19 GROUP BY
20 t1.'Name'n,
21 t2.predict
22 .
    
```

기본적으로 질의는 **출력 데이터** 탭에 결과를 표시하고 Work 라이브러리에 출력 테이블을 생성합니다.

1	AVG_Age
1	13.315789474

기본적으로 칼럼을 집계하면 출력은 집계 없는 모든 칼럼으로 그룹화됩니다. 자세한 내용은 [“출력 그룹화” \(94페이지\)](#)에서 확인하십시오.

조건부 표현식 작업

조건부 표현식 생성

CASE 함수를 사용하는 계산된 칼럼을 생성하여 질의에 조건부 표현식을 생성할 수 있습니다. 추가 표현식에서 "IF-THEN/ELSE"와 유사한 로직을 사용하여 칼럼을 생성하는 데 사용할 수 있는 값을 반환합니다.

주: FedSQL 질의를 생성할 때에는 조건부 표현식을 생성할 수 없습니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.

예를 들어 성의 마지막 문자를 추출한 다음 "A"에서 "F"로 시작하는 성을 "A - F" 값으로 바꾸고, "G"부터 "L"로 시작하는 성을 "G - L"로 바꾸는 식으로 진행할 수 있는 표현식을 생성할 수 있습니다.

조건부 표현식을 생성하려면 다음을 수행합니다.

- 1 질의 탭의 칼럼 영역에서 **추가** ⇨ **계산된 칼럼**을 클릭합니다.
- 2 표현식 상자에 CASE 표현식을 직접 입력하려면 CASE 표현식 구문에 대해 다음 템플릿을 사용합니다.

```
CASE <CaseOperand>
  WHEN <whenCondition> THEN <resultExpression>
  WHEN <whenCondition> THEN <resultExpression>
  ELSE <resultExpression>
END
```

- **<CaseOperand>** - 표현식에서 사용하는 칼럼의 이름
- **<whenCondition>** - 칼럼의 기존 값
- **<resultExpression>** - 칼럼에 할당할 값입니다.

- 3 칼럼에 대한 표현식을 생성한 후 표현식 빌더 창의 아래쪽에 있는 **속성** 탭을 사용하여 칼럼의 이름을 지정합니다. 레이블, 데이터 유형, 길이, 출력형식 및 입력형식을 지정할 수도 있습니다.

주: 자세한 내용은 *SAS SQL Procedure User's Guide*의 [“CASE Expression”](#)에서 확인하십시오.

샘플 조건부 표현식

다음 예제에서는 SUBSTR 함수를 사용하여 성의 첫 문자를 추출한 다음 이 문자에 따라 대체 값을 할당합니다.

```
CASE WHEN (SUBSTR(tablename.LastName,1,1) between "A" and "F") THEN "A - F"
      WHEN (SUBSTR(tablename.LastName,1,1) between "G" and "L") THEN "G - L"
      WHEN (SUBSTR(tablename.LastName,1,1) between "M" and "R") THEN "M - R"
      WHEN (SUBSTR(tablename.LastName,1,1) between "S" and "Z") THEN "S - Z"
END
```

다음 예제에서는 나이 값을 "Under 35," "35 - 50" 및 "Over 50"으로 바꿉니다.

```
CASE
  WHEN tablename.Age < 35 THEN "Under 35"
  WHEN tablename.Age between 35 and 50 THEN "35 - 50"
  ELSE "Over 50"
END
```

데이터 필터링

필터 생성

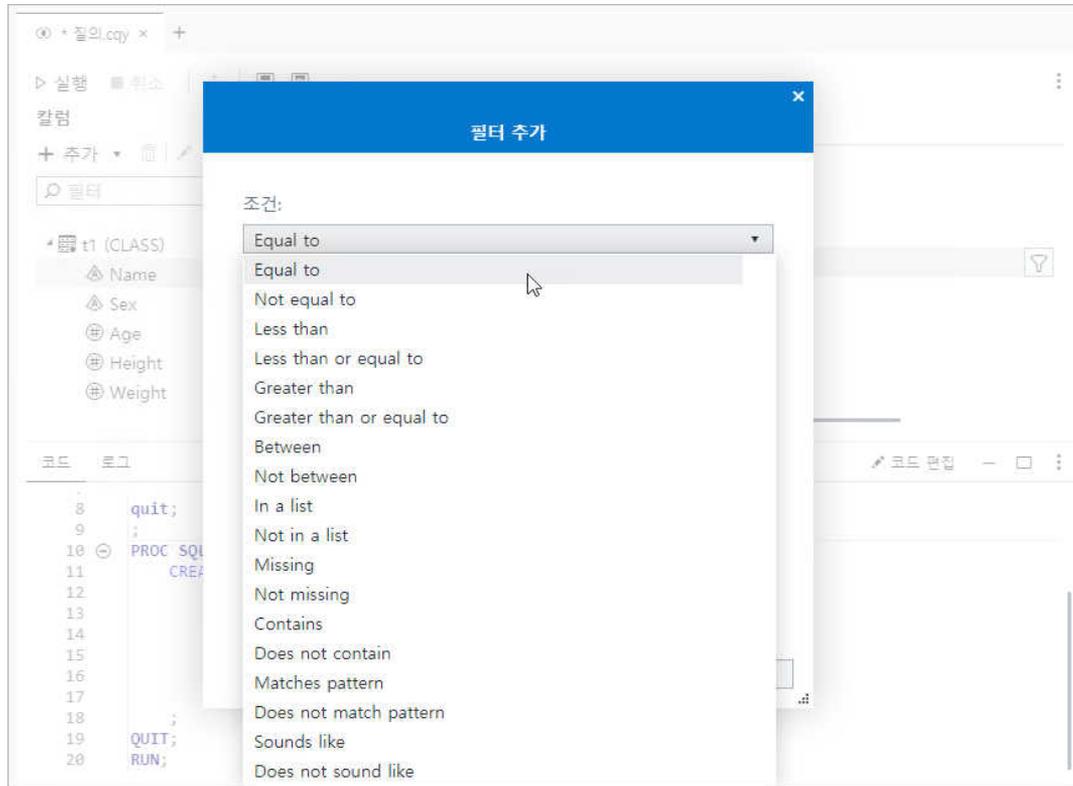
데이터를 질의할 때 데이터에서 칼럼의 값에 따라 특정 기준을 만족하는 행만 검색할 수 있습니다. SAS Studio에 검색할 행을 지정하는 과정을 필터 설정이라고 하며, 이 작업은 **필터** 탭에서 수행됩니다. SQL 질의에서 WHERE 절을 사용할 때 해당합니다.

표현식 빌더를 사용하여 필터 표현식을 생성하려면 도구 모음에서 **표현식 빌더**를 클릭합니다. 자세한 내용은 [“표현식 구성” \(187페이지\)](#)에서 참조하십시오.

- 1 **필터** 탭을 클릭한 다음 칼럼 영역에서 하나 이상의 칼럼을 끌어옵니다.

.....
 주: 출력으로 선택하지 않은 칼럼을 사용하여 출력 테이블을 필터링할 수 있습니다.

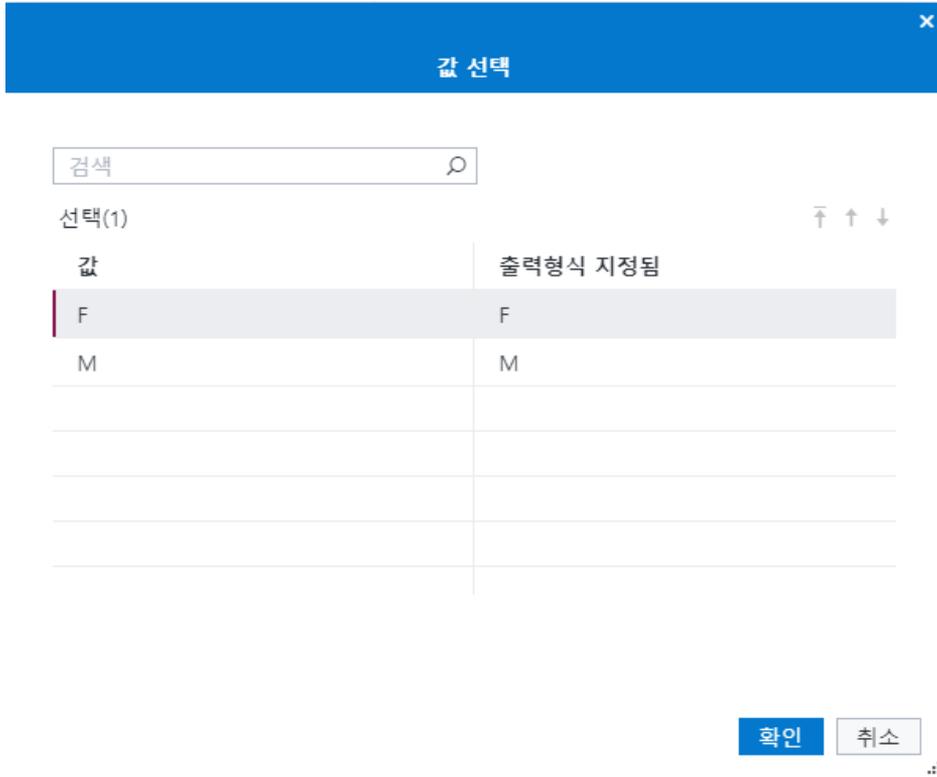
- 2 필터 기준을 지정할 칼럼 옆의  을 클릭합니다. 필터 창의 연산자 드롭다운 리스트에서 비교 연산자를 선택합니다. 기본값은 **Equal to**입니다.



3. 선택한 연산자에 값이 필요하다면 값 상자에서 값을 입력하거나 선택합니다. 값 리스트에서 선택하려면  을 클릭하고 값 선택 창에서 값 가져오기를 클릭합니다. 사용할 값을 선택하고 확인을 클릭합니다.

주: 값 창에서 값을 검색하려면 출력형식이 지정되지 않은 값을 사용하십시오.

주: **In a list** 또는 **Not in a list** 연산자를 사용할 때  을 클릭해서 값을 입력하거나 + 값 리스트에서 선택합니다.



- 4 필터에서 사용 중인 칼럼의 데이터 유형에 따라 다음 옵션 중에서 선택할 수 있습니다.
 - **대소문자 구분** - 지정하는 값의 대소문자와 일치하는 행만 검색합니다. 이 옵션이 선택되지 않으면 UPPER 함수가 표현식에 적용됩니다. 이 옵션은 기본적으로 선택되어 있지 않습니다.
 - **문자열 따옴표로 묶기** - 값을 작은따옴표로 묶습니다. 이 옵션은 기본적으로 선택되어 있습니다. 필터 실행 중 매크로 변수 또는 계산된 다른 값을 사용하려면 이 옵션을 취소해야 합니다.
 - **매크로 허용** - 숫자 칼럼에서 필터에 문자 값을 입력할 수 있습니다.
 - **원시 값 사용** - 출력형식이 지정된 값 대신 출력형식이 지정되지 않은 숫자, 날짜, 시간 및 날짜시간 값을 사용합니다. 이 옵션은 기본적으로 선택되어 있으며 숫자, 날짜, 시간 및 날짜시간 칼럼에 대해서만 사용할 수 있습니다. 출력형식 지정된 값을 기반으로 필터를 만들려면 이 옵션을 선택 취소합니다.

.....
주:

- 칼럼에 서식이 지정된 값이 포함되어 있지 않으면 필터에서 원시 값이 사용됩니다.
 - FedSQL 질의를 생성하는 경우 일부 출력형식은 출력형식이 지정된 값을 기반으로 하는 필터에서 사용할 수 없습니다.
-

팁 필터에 원시 값을 사용하는 것이 좋습니다. 출력형식이 지정된 값을 사용하는 필터가 올바르게 작동하지 않으면 출력형식이 지정되지 않은 원시 값을 필터링하십시오.

- 5 **필터**를 클릭하여 필터에 값을 추가합니다.

필터 표현식 간 관계 변경

필터 표현식에는 한 개의 칼럼만 사용할 수 있지만 여러 칼럼을 사용하여 여러 표현식을 생성할 수 있습니다. 필터에 한 개 이상의 비교 표현식을 생성하면, 이러한 필터 요소 간의 기본 관계는 AND입니다. 필터 요소 간의 관계를 AND에서 OR로 변경할 수 있습니다.

필터 간 관계를 변경하려면 다음을 수행합니다.

- 필터 탭에서 연산자 값을 클릭하고 새로운 값을 선택합니다.

The screenshot shows the SAS Query Builder interface. On the left, a list of columns includes 't1 (CLASS)', 't2 (CLASSFIT)', and '계산된 칼럼' (Calculated Column) with 'AVG_Age'. The main area shows filter expressions: 'UPPER(t1.Sex) = 'F'' and 't1.Age > 12'. A dropdown menu for the relationship operator is open, showing options for 'AND', 'AND', and 'OR'. The bottom pane shows the generated SQL code:

```

10 PROC SQL;
11 CREATE TABLE WORK.QUERY AS
12 SELECT
13 (t1.'Name'n) AS 'Student Name'n,
14 (AVG(t1.Age)) LENGTH=8 AS AVG_Age,
15 (t1.Weight) LABEL="Predicted Value of Weight"
16 FROM
17 SASHELP.CLASS t1
18 INNER JOIN SASHELP.CLASSFIT t2 ON (t1.'Name'n = t2.'Name'n)
19 WHERE
20 UPPER(t1.Sex) = 'F' AND
21 t1.Age > 12
22 ~
  
```

여러 개의 칼럼으로 데이터를 필터링하면 테이블이 처음 나열된 칼럼으로 먼저 필터링됩니다. 칼럼을 마우스 오른쪽 버튼으로 클릭하고 알맞은 옵션을 선택하여 칼럼을 리스트의 위, 아래 및 맨 위 또는 맨 아래로 이동할 수 있습니다. 이동할 칼럼을 선택하고 칼럼 중 하나를 마우스 오른쪽 버튼으로 클릭한 다음 알맞은 옵션을 선택하여 리스트에서 이웃한 여러 칼럼을 이동할 수 있습니다.

출력 관리

출력 정렬

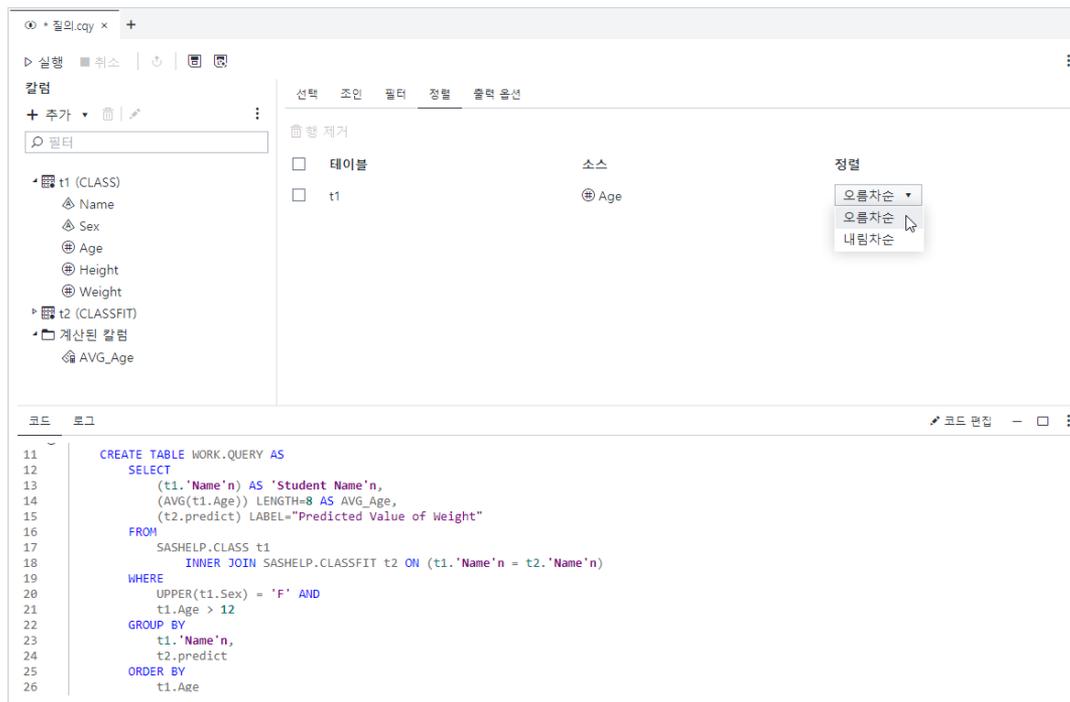
질의에서 사용한 테이블에서 한 개 이상의 칼럼을 사용하여 질의에서 출력을 정렬할 수 있습니다.

주: 출력으로 선택하지 않은 칼럼을 사용하여 출력 테이블을 정렬할 수 있습니다.

주: CAS 데이터를 사용하여 FedSQL 질의를 생성하는 경우 질의 출력을 정렬할 수 없습니다. 자세한 내용은 ["FedSQL 질의 생성" \(102페이지\)](#)에서 확인하십시오.

출력을 정렬하려면 다음을 수행합니다.

- 1 정렬 탭을 클릭하고 칼럼 영역에서 하나 이상의 칼럼을 끌어옵니다.
- 2 데이터를 정렬할 칼럼에 대해 정렬 상자를 클릭합니다. 드롭다운 리스트에서 **오름차순** 또는 **내림차순**을 선택합니다. 기본 정렬 방향은 **오름차순**입니다.



- 3 여러 개의 칼럼으로 정렬하면 출력 테이블이 리스트의 첫 칼럼으로 처음 정렬됩니다. 이어 첫 번째 칼럼의 각 레벨에서 리스트의 두 번째 칼럼을 기준으로 행이 정렬됩니다. 칼럼을 마우스 오른쪽 버튼으로 클릭하고 알맞은 옵션을 선택하여 칼럼을 리스트의 위, 아래 및 맨 위 또는 맨 아래로 이동할 수 있습니다. 이동할 칼럼을 선택하고 칼럼 중 하나를 마우스 오른쪽 버튼으로 클릭한 다음 알맞은 옵션을 선택하여 리스트에서 이웃한 여러 칼럼을 이동할 수 있습니다.

출력에서 중복 행 제거

일부 질의 유형은 여러 개의 동일 행을 생성합니다. SAS Studio를 사용하면 동일한 행을 하나만 남기고 불필요한 중복 행을 제거할 수 있습니다.

중복 행을 제거하려면 칼럼 영역에서 을 클릭하고 **고유 행만 선택** 체크 박스를 선택합니다.

출력 그룹화

요약된 칼럼을 생성하면 데이터를 칼럼 값에 기반한 그룹으로 분류할 수 있습니다. 이 작업은 SQL 질의에서 GROUP BY 절을 사용하는 것과 같습니다. 예를 들어 학생 그룹의 평균 키를 계산하면 각 나이 그룹에 대한 평균 키를 표시하도록 결과를 그룹화할 수 있습니다.

기본적으로 **선택** 탭의 **그룹** 하위 탭에서 **자동으로 데이터 그룹화** 옵션이 선택됩니다. 이 옵션이 선택되고 칼럼에 대해 요약 함수를 수행하면 질의는 요약 기능이 없는 모든 칼럼으로 자동 그룹화됩니다. 질의가 그룹화된 칼럼 리스트를 편집할 수 있습니다. 리스트에서 칼럼을 추가하거나 제거할 수 있으며 **자동으로 데이터 그룹화** 옵션을 선택 취소하여 리스트에서 모든 칼럼을 제거할 수 있습니다. 기본 그룹화 선택으로 돌아가려면 **자동으로 데이터 그룹화** 옵션을 선택합니다.

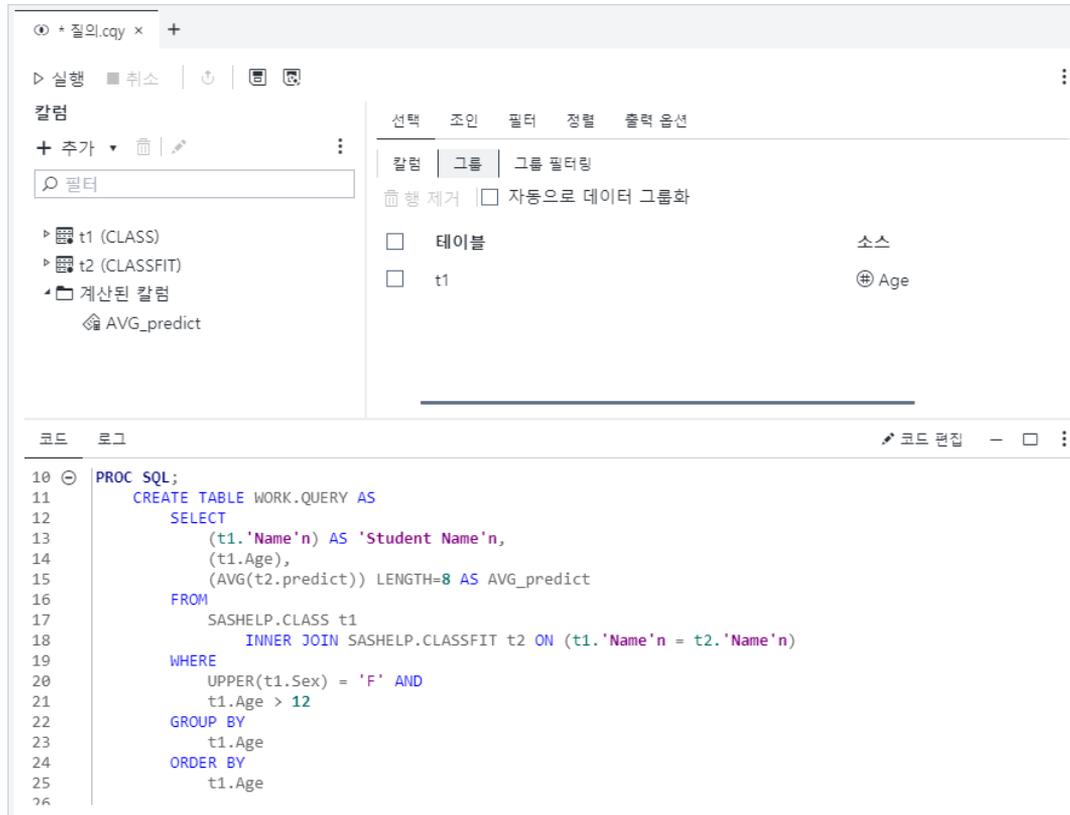
주: 자동으로 데이터 그룹화 옵션은 칼럼 이름을 사용하여 질의에 명시적으로 추가된 칼럼만 포함하고 별표(*) 표기법을 사용하여 질의에 추가된 칼럼은 포함하지 않습니다. 자세한 내용은 **“출력에서 칼럼 지정” (83페이지)**에서 확인하십시오.

출력을 그룹화하려면 다음을 수행합니다.

- 1 **선택** 탭의 **그룹** 하위 탭을 클릭한 다음 칼럼 영역에서 하나 이상의 칼럼을 끌어옵니다.

주: 출력에 대해 선택하지 않은 칼럼으로 출력을 그룹화할 수 있습니다.

주: **그룹** 하위 탭으로 칼럼을 추가하거나 탭에서 제거하면 **자동으로 데이터 그룹화** 옵션이 자동으로 선택 취소됩니다. **그룹** 하위 탭에 추가한 칼럼을 제거하려면 **자동으로 데이터 그룹화** 옵션을 다시 선택합니다. 제거할 칼럼을 선택하고 **행 삭제**를 클릭하여 개별 칼럼도 제거할 수 있습니다.



- 여러 개의 칼럼으로 데이터를 그룹화하면 테이블이 처음 나열된 칼럼으로 먼저 필터링됩니다. 칼럼을 마우스 오른쪽 버튼으로 클릭하고 알맞은 옵션을 선택하여 칼럼을 리스트의 위, 아래 및 맨 위 또는 맨 아래로 이동할 수 있습니다. 이동할 칼럼을 선택하고 칼럼 중 하나를 마우스 오른쪽 버튼으로 클릭한 다음 알맞은 옵션을 선택하여 리스트에서 이웃한 여러 칼럼을 이동할 수 있습니다.

다음 예는 각 나이 그룹에서 학생의 평균 몸무게를 찾는 방법을 보여줍니다. 먼저 Age 및 Weight 칼럼을 **선택** 탭에 추가한 다음 AVG 요약 함수를 사용하여 Weight 칼럼을 계산된 칼럼으로 변환합니다.

The screenshot shows the SAS Studio interface with a PROC SQL query in the main editor. The query is as follows:

```

10 PROC SQL;
11   CREATE TABLE WORK.QUERY AS
12   SELECT
13     (t1.Age),
14     (AVG(t1.Weight)) LENGTH=8 AS AVG_Weight
15   FROM
16     SASHELP.CLASS t1
17     INNER JOIN SASHELP.CLASSFIT t2 ON (t1.'Name'n = t2.'Name'n)
18   GROUP BY
19     t1.Age
20   ORDER BY
21     t1.Age
22 ;
23 QUIT;
24 RUN;
    
```

The interface also shows the 'Columns' panel on the left with 't1 (CLASS)', 't2 (CLASSFIT)', and '계산된 칼럼' (Calculated Columns) containing 'AVG_Weight'. The 'Groups' panel on the right shows 'Age' selected as the grouping variable.

나이별 학생 몸무게를 보려면 질의를 Age 칼럼으로 그룹화합니다. 결과에서 각 나이 그룹에 대한 평균 몸무게가 표시됩니다.

The screenshot shows the SAS Studio interface with the output of the PROC SQL query displayed in a table. The table has two columns: 'Age' and 'AVG_Weight'. The data is as follows:

	Age	AVG_Weight
1	11	67.75
2	12	94.4
3	13	88.66666667
4	14	101.875

The interface also shows the 'Columns' panel on the left with 't1 (CLASS)', 't2 (CLASSFIT)', and '계산된 칼럼' (Calculated Columns) containing 'AVG_Weight'. The 'Groups' panel on the right shows 'Age' selected as the grouping variable. The output table is displayed below the query editor.

주: 기본적으로 질의는 결과의 테이블을 생성합니다. 결과의 리포트(결과 탭에 표시됨)를 생성하려면 질의에 대한 출력 유형으로 리포트를 지정해야 합니다. 자세한 내용은 “결과 저장” (98페이지)에서 확인하십시오.

그룹화된 데이터 서브셋 지정

요약 함수가 포함된 계산된 칼럼 또는 요약된 칼럼을 생성했으면, 칼럼의 값에 따라 데이터를 그룹으로 분류할 수 있습니다. 출력을 하나 이상의 칼럼별로 그룹화하면, 지정된 조건에 따라 그룹화된 데이터의 서브셋을 지정하는 필터를 생성할 수 있습니다. 이 작업은 SQL 질의에서 HAVING 표현식을 사용하는 것과 같습니다.

주: 요약되지 않은 원시 데이터를 필터링하도록 선택할 수도 있습니다. 자세한 내용은 “필터 생성” (89페이지)에서 확인하십시오.

필터에 한 개의 칼럼만 사용하거나 여러 칼럼을 사용하여 여러 비교 표현식을 생성할 수 있습니다. 필터에 한 개 이상의 비교 표현식을 생성하면, 이러한 필터 요소 간의 기본 관계는 AND입니다. 필터 요소 간의 관계를 AND에서 OR로 변경할 수 있으며, 요소들을 함께 그룹화할 수 있습니다.

기본 필터를 생성하도록 선택하거나 표현식 빌더를 사용하여 고급 필터를 생성할 수 있습니다.

- 1 **선택** 탭의 **필터 그룹** 하위 탭을 클릭한 다음 칼럼 영역에서 하나 이상의 칼럼을 끌어옵니다.

주: 출력에 대해 선택하지 않은 칼럼으로 데이터를 그룹화할 수 있습니다.

- 2 필터 기준을 지정할 칼럼 옆의  을 클릭합니다. 필터 창의 **조건** 드롭다운 리스트에서 비교 연산자를 선택합니다. 기본값은 **Equal**입니다.
- 3 선택한 연산자에 값이 필요하다면 **값** 상자에서 값을 입력하거나 선택합니다. 값 리스트에서 선택하려면  을 클릭하고 값 선택 창에서 **값 가져오기**를 클릭합니다. 사용할 값을 선택하고 **확인**을 클릭합니다.

주: 값 창에서 값을 검색하려면 출력형식이 지정되지 않은 값을 사용하십시오.

- 4 필터에서 사용 중인 칼럼의 데이터 유형에 따라 다음 옵션 중에서 선택할 수 있습니다.
 - **대소문자 구분** - 지정하는 값의 대소문자와 일치하는 행만 검색합니다.
 - **문자열 따옴표로 묶기** - 값을 작은따옴표로 묶습니다. 이 옵션은 기본적으로 선택되어 있습니다. 필터 실행 중 매크로 변수 또는 계산된 다른 값을 사용하려면 이 옵션을 취소해야 합니다.
 - **매크로 허용** - 숫자 칼럼에서 필터에 문자 값을 입력할 수 있습니다.
- 5 **필터**를 클릭하여 질의에 필터 표현식을 추가합니다.

주: 하나 이상의 필터 표현식을 생성하면 표현식 간의 관계를 변경할 수 있습니다. 자세한 내용은 “필터 표현식 간 관계 변경” (92페이지)에서 확인하십시오.

결과 저장

결과를 리포트, 데이터 테이블 또는 데이터 뷰의 3가지 출력형식 중 하나로 생성할 수 있습니다.

결과를 데이터 테이블이나 데이터 뷰로 저장하면 사용할 라이브러리 및 테이블 이름을 지정할 수 있습니다. 라이브러리 및 테이블 이름을 지정하지 않으면 결과는 Work 라이브러리에 저장됩니다.

결과 출력형식을 지정하려면 다음을 수행합니다.

- 1 질의 창에서 **출력 옵션** 탭을 클릭합니다.
- 2 **출력 유형** 드롭다운 리스트에서 사용하려는 출력형식을 선택합니다.

리포트

질의 결과를 사용자가 다운로드할 수 있는 HTML, PDF 또는 RTF 파일의 리포트로 저장합니다. 질의 결과는 질의를 다시 실행해야만 업데이트됩니다. 이러한 출력형식의 질의 결과에 대해서는 작업을 실행할 수 없습니다.

테이블

작업을 실행할 수 있는 정적 데이터 테이블로 질의 결과를 저장합니다. 질의 결과는 질의를 다시 실행해야만 업데이트됩니다. 기본적으로 데이터 테이블은 Work 라이브러리에 저장됩니다.

뷰

작업을 실행할 수 있는 동적 데이터 뷰로 질의 결과를 저장합니다. 데이터 뷰 형식으로 질의 결과를 열 때마다 질의에 사용된 데이터의 변경 사항이 적용되어 결과가 업데이트됩니다. 기본적으로 데이터 뷰는 Work 라이브러리에 저장됩니다.

.....
 주: FedSQL 질의를 생성할 때에는 이 옵션을 사용할 수 없습니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.

출력 데이터를 특정 위치에 저장하려면 다음을 수행합니다.

- 1 질의 탭에서 **출력 옵션** 탭을 클릭합니다.
- 2 CAS 테이블을 사용하여 FedSQL 질의를 생성할 때 **세션 참조** 옵션을 이용해서 사용할 기본 CAS 세션 이름을 지정해야 합니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.
- 3 **라이브러리** 상자에서 결과를 저장할 라이브러리의 이름을 입력합니다.
- 4 출력 데이터의 이름을 지정하려면 **테이블 이름** 상자에 사용하려는 이름을 입력합니다.

입력 처리에 대한 제한 지정

질의를 생성할 때 질의가 각 소스 테이블에서 처리하는 행의 수를 제한할 수 있습니다. 이 옵션은 PROC SQL의 INOBS= 옵션에 해당하며, 질의를 테스트하기 위해 처리 시간을 줄이고 할 때 유용합니다.

주: FedSQL 질의를 생성할 때에는 이 옵션을 사용할 수 없습니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.

처리할 행 수에 제한 n 을 설정하면, SAS Studio는 질의에서 각 소스 테이블의 처음 n 개 행을 사용합니다. 질의에 필터를 추가했다면 SAS Studio는 각 테이블에서 필터 기준과 일치하는 처음 n 개 행을 사용합니다.

출력형식이 데이터 뷰이면 이 옵션이 적용되지 않습니다. 자세한 내용은 [“결과 저장” \(98페이지\)](#)에서 확인하십시오.

주: 합계 또는 평균처럼 테이블 전체에 대해 함수를 실행하는 컴퓨팅 칼럼이 질의 데이터에 있을 때 입력 행 수를 제한하면 해당 칼럼에 대해 계산되는 값에 영향을 미칩니다. 함수는 처리된 행에서만 수행됩니다.

처리할 행의 수를 제한하려면 다음을 수행합니다.

- **출력 옵션** 탭을 클릭합니다. **처리할 일치 행 수 제한(INOBS)**을 선택하고 질의를 처리할 행 수를 지정합니다.

주: 처리할 행 수를 제한하고 테이블을 조인했다면, 질의가 반환할 수 있는 최대 행수는 각 테이블 행 수의 Cartesian product입니다. 예를 들어 5개 행으로 제한을 설정하고 두 테이블을 제한하면, 질의가 반환할 수 있는 최대 행 수는 25 ($5 * 5$)입니다. 세 개의 테이블을 조인하면 질의가 반환할 수 있는 최대 행 수는 125 ($5 * 5 * 5$)입니다.

출력 처리에 대한 제한 지정

질의를 생성할 때 질의가 출력에 저장하는 행의 수를 제한할 수 있습니다. 이 옵션은 PROC SQL의 OUTOBS= 옵션에 해당하며, 테스트 목적으로 출력의 크기를 제한하고자 할 때 유용합니다.

주: FedSQL 질의를 생성할 때에는 이 옵션을 사용할 수 없습니다. 자세한 내용은 [“FedSQL 질의 생성” \(102페이지\)](#)에서 확인하십시오.

출력형식이 데이터 뷰이면 이 옵션이 적용되지 않습니다. 자세한 내용은 [“결과 저장” \(98페이지\)](#)에서 확인하십시오.

주: 출력의 크기를 제한하더라도 질의의 처리 시간이 감소하지 않습니다. 질의는 여전히 각 소스 테이블에서 모든 데이터를 처리합니다. 출력의 크기만 줄어든 뿐입니다.

- **출력 옵션** 탭을 클릭합니다. **출력에 저장할 일치 행 수 제한(OUTOBS)**을 선택하고 출력에 사용할 행 수를 지정합니다.

Pass-Through 및 테이블 옵션 지정

Oracle 데이터베이스의 테이블만 포함하는 질의에 대해 명시적 Pass-Through를 사용할지 여부를 지정할 수 있습니다. 이 옵션을 선택하면 처리를 위해 SQL 코드가 Oracle 데이터베이스로 전달됩니다. 데이터 파일이 매우 큰 경우 처리를 위해 파일을 SAS 서버에 복사할 필요가 없기 때문에 이 옵션을 사용하면 성능이 향상될 수 있습니다. 표현식 빌더를 사용하여 조인 또는 필터 표현식과 같은 표현식을 작성하는 경우 이 옵션을 사용하면 Oracle SQL 구문을 사용할 수 있습니다.

주: 이 옵션을 선택하지 않으면 SAS Studio는 SAS/ACCESS Interface to Oracle 데이터베이스 엔진을 사용하여 타겟 데이터베이스에 대한 코드를 생성합니다.

출력 테이블이 생성되거나 업데이트될 때 적용할 옵션을 지정할 수도 있습니다. SQL 질의를 생성하는 경우 SQL 문 테이블 옵션을 지정해야 합니다. FedSQL 질의를 생성하는 경우 FedSQL 문 테이블 옵션을 지정해야 합니다.

명시적 Pass-Through 처리를 사용할지 여부를 지정하려면 다음을 수행합니다.

- **출력 옵션** 탭을 클릭한 다음 **소스 테이블의 명시적 Pass-Through 처리** 옵션을 클릭합니다.

주: 질의에 Oracle 테이블이 아닌 다른 테이블이 포함되어 있거나 PROC FEDSQL을 사용하여 코드를 생성하는 경우에는 이 옵션을 사용할 수 없습니다.

출력 테이블 옵션을 지정하려면 다음을 수행합니다.

- **출력 옵션** 탭을 클릭합니다. **출력 테이블 옵션 입력(빈 칸으로 구분)** 상자에 옵션을 입력합니다.

주: 공백으로 여러 옵션을 구분하고 문자열 값에 필요한 모든 인용 부호를 포함해야 합니다.

선택
조인
필터
정렬
출력 옵션

출력 유형:

라이브러리:

테이블 이름:

처리할 일치 행 수 제한(INOBS):

출력에 저장할 일치 행 수 제한(OUTOBS):

▼ Pass-Through 및 테이블 옵션

소스 테이블의 명시적 Pass-Through 처리

출력 테이블 옵션 입력(빈 칸으로 구분):

▼ 노트

코드 편집

질의에 노트 추가

출력 옵션 탭에서 노트로 질의에 주석을 달 수 있습니다.

- **출력 옵션** 탭을 클릭합니다. **노트** 상자에 노트를 입력합니다.

질의 실행

질의에 대한 모든 기준을 지정한 후 질의 탭 도구 모음에서 **실행**을 클릭하여 결과를 생성할 수 있습니다. 출력 데이터 또는 결과가 질의 탭의 결과 영역에 추가됩니다.

주: 테이블만 지정되고 선택한 칼럼, 필터, 정렬 기준 및 출력 옵션이 제거된 초기 상태로 질의를 재설정할 수 있습니다. 질의를 재설정하려면 **> 재설정**을 클릭합니다.

FedSQL 질의 생성

기본적으로 SAS Studio의 질의는 PROC SQL을 사용하여 결과를 생성합니다. 그러나 PROC SQL 대신 PROC FEDSQL을 사용하여 질의를 생성하도록 선택할 수 있습니다. PROC SQL은 SAS 서버의 데이터를 사용하거나 단일 데이터베이스의 데이터를 사용하는 질의를 위해 고안되었습니다. PROC FEDSQL은 다중 데이터베이스 연결에서 테이블에 액세스하고 단일 결과 집합을 반환하는 통합 질의를 위해 고안되었습니다.

PROC FEDSQL은 SQL 프로시저에 제공된 것보다 더 많은 기능이 필요한 환경에서 작업 중일 때 이점이 더 많습니다.

- FedSQL은 ANSI SQL:1999 코어 표준을 준수합니다.
- FedSQL은 BIGINT, DATE, INTEGER 및 VARCHAR 등 SAS SQL 구현에서 사용 가능한 것보다 더 많은 데이터 유형을 지원합니다.
- FEDSQL 질의는 CAS 서버에서 실행할 수 있습니다.
- FedSQL은 통합 질의를 처리하고 다중 데이터베이스 연결에서 테이블을 조인할 수 있습니다.
- FedSQL 언어는 타겟 데이터 소스가 질의에 나타나지 않더라도 지원되는 모든 데이터 소스에서 데이터를 생성할 수 있습니다.

자세한 내용은 [Overview: PROC FEDSQL](#)에서 참조하십시오.

다음 테이블에는 PROC SQL 및 PROC FEDSQL에서 지원되는 데이터 유형이 나열되어 있습니다.

PROC SQL	PROC FEDSQL
CHARACTER	CHARACTER
DATE*	DATE
DECIMAL*	DECIMAL*
DOUBLE	DOUBLE

PROC SQL	PROC FEDSQL
FLOAT*	FLOAT*
INTEGER*	INTEGER
NUMERIC	NUMERIC
REAL*	REAL*
SMALLINT*	SMALLINT*
	BIGINT
	VARCHAR

주: * 해당 데이터 유형을 지정할 때에는 CHARACTER 또는 DOUBLE (NUMERIC) 데이터 유형으로 변환됩니다.

기본적으로 질의는 PROC SQL을 사용하여 생성됩니다. 모든 질의의 기본 프로시저를 PROC FEDSQL로 변경할 뿐만 아니라 개별 질의에서 사용되는 프로시저도 변경할 수 있습니다.

- 새로운 질의를 생성할 때 PROC FEDSQL을 자동으로 사용하려면 **옵션** ⇨ **우선 설정**을 선택하고 **질의**를 클릭합니다. **PROC FEDSQL**을 선택합니다.

주: 이 옵션은 옵션을 선택한 이후에 생성한 질의에 대해서만 적용됩니다.

- 개별 질의에서 사용되는 프로시저를 지정하려면 질의에서 **출력 옵션** 탭을 클릭합니다. **다음**을 사용하여 **코드 생성** 옵션을 사용하여 질의 생성을 시작하기 전에 **PROC SQL** 또는 **PROC FEDSQL**을 지정합니다. 질의 생성을 시작한 후에 이 옵션을 변경하면 질의는 오류를 포함할 수 있습니다.

주: CAS 테이블을 사용하여 FedSQL 질의를 생성할 때 **출력 옵션** 탭의 **세션 참조** 옵션을 이용하여 사용할 기본 CAS 세션 이름을 지정해야 합니다.

SQL 및 FedSQL 질의를 생성하기 위한 질의 함수는 매우 유사합니다. 다음 테이블에는 SQL 및 FedSQL 질의 사이의 차이점이 일부 나열됩니다.

질의 기능	SQL에서의 사용 가능성	FedSQL에서의 사용 가능성
출력 유형 - 보기	예	아니요
출력 데이터 - CAS 테이블*	예, 입력 데이터 소스가 하나 이상의 SAS 테이블 및 DBMS 데	예, 입력 데이터 소스가 하나 이상의 CAS 테이블인 경우입니다.

질의 기능	SQL에서의 사용 가능성	FedSQL에서의 사용 가능성
	이터셋인 경우입니다.	
입력 데이터 - 둘 이상의 디렉터리와 연결된 라이브러리 참조(예: SASHELP)	예	아니요
탭 선택: 길이, 레이블 및 입력형식 칼럼	예	아니요, 또한 출력형식 칼럼은 독립 실행형 질의에만 사용할 수 있습니다.
출력 정렬	예	예, 질의에서 CAS 데이터를 사용하지 않는 경우입니다. 예, 질의에서 CAS 데이터를 사용하는 경우입니다.
출력 옵션 - 세션 참조	아니요	예
출력 옵션: 처리할 일치 행 수 제한, 출력에 저장할 일치 행 수 제한	예	아니요
연산자: Contains, Does not contain, Sounds like, Does not sound like, Any, All	예	아니요
함수: CASE, CAT, CATS, CATX, COMPBL, COMPRESS, FIND, INPUT, INT, INTCK, INTNX, LEFT, LENGTH, MDY, N, NLDATE, NLDATM, PROPCASE, SCAN, SUBSTR, TRANWRD, WEEK, YRDIF	예	아니요
집계: CV, FREQ, MEAN, N, PRT, STD, SUMWGT, T, VAR	예	아니요 FedSQL에서만 다음 집계를 사용할 수 있습니다. <ul style="list-style-type: none"> ■ PROBT(PRT 대체) ■ STDDEV(STD 대체) ■ STUDENT_T(T 대체) ■ VARIANCE(VAR 대체)

* FedSQL 질의의 출력이 CAS 테이블이면 입력 데이터 소스도 CAS 테이블이어야 합니다. FedSQL 질의의 출력이 SAS 테이블 또는 DBMS 테이블이면 입력도 SAS 테이블 및 DBMS 테이블이어야 합니다.

데이터 작업

테이블 뷰어 정보	105
데이터 열기 및 보기	107
테이블을 생성하는 데 사용된 코드 보기	111
테이블 뷰어의 작업 또는 질의에서 데이터 열기	111
데이터 새로 고침	113
데이터 정렬 및 고정	113
데이터 필터링	114
데이터 필터링 정보	114
빠른 필터 생성	115
필터 표현식 생성	116
데이터 가져오기	118
SAS Studio에 데이터 가져오기 정보	118
Excel 워크시트 가져오기	118
구분된 파일 가져오기(CSV, TSV, DLM)	120
DBMS 파일 가져오기	122
데이터 가져오기 작업 저장	123
데이터 내보내기	124

테이블 뷰어 정보

테이블 뷰어는 SAS Studio에서 데이터를 표시하는 데 사용됩니다. ↓ 및 ↑을 클릭하여 데이터를 탐색합니다.

주: ↓ 및 ↑을 클릭하여 데이터의 마지막 또는 처음 페이지로 이동할 수 있습니다. CAS 테이블, 알 수 없는 수의 칼럼이 있는 테이블 또는 데이터 보기의 마지막 페이지로 직접 탐색할 수 없습니다.

	Make	Model	Type	Origin	Drive
1	Acura	MDX	SUV	Asia	All
2	Acura	RSX Type S 2dr	Sedan	Asia	Front
3	Acura	TSX 4dr	Sedan	Asia	Front
4	Acura	TL 4dr	Sedan	Asia	Front
5	Acura	3.5 RL 4dr	Sedan	Asia	Front
6	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front
7	Acura	NSX coupe 2dr manual S	Sports	Asia	Rear
8	Audi	A4 1.8T 4dr	Sedan	Europe	Front
9	Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front
10	Audi	A4 3.0 4dr	Sedan	Europe	Front
11	Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All
12	Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All
13	Audi	A6 3.0 4dr	Sedan	Europe	Front
14	Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All
15	Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front

주: 기본적으로 테이블 뷰어에는 테이블의 처음 200개 행과 100개 칼럼이 표시됩니다. 테이블이 열려 있는 동안 테이블이 구조 또는 데이터 값이 변경되면 테이블 뷰어를 새로 고쳐야 변경 사항을 볼 수 있습니다. 테이블 구조를 변경하고 새로 고침하지 않으면, 탐색 영역의 라이브러리 섹션에서 나열된 칼럼은 테이블 뷰어에서 표시되는 칼럼과 다를 수 있습니다. 자세한 내용은 “데이터 새로 고침” (113페이지)에서 확인하십시오.

탐색 영역의 라이브러리 섹션 도구 모음에서 을 클릭하여 테이블과 해당 칼럼의 속성을 볼 수 있습니다.

테이블 속성

일반 **칼럼 속성**

이름: CARS

레이블: 2004 Car Data

유형: 테이블

위치: SASHELP.CARS

행: 428

칼럼: 15

생성일: 2019. 11. 5. 오전 2:09:55

수정일: 2019. 11. 5. 오전 2:09:55

닫기

테이블 속성

일반 **칼럼 속성**

칼럼 이름	레이블	유형	길이	출력형식	입력형식
Make		문자	13		
Model		문자	40		
Type		문자	8		
Origin		문자	6		
DriveTrain		문자	5		
MSRP		숫자	8	DOLLAR8.	
Invoice		숫자	8	DOLLAR8.	
EngineSize	Engine Size (L)	숫자	8		
Cylinders		숫자	8		

닫기

데이터 열기 및 보기

SAS Studio에서 파일을 여러 방법으로 열 수 있습니다.

- 탐색 영역의 탐색기 및 라이브러리 섹션에서 파일을 더블 클릭할 수 있습니다.
- 탐색 영역의 탐색기 및 라이브러리 섹션에서 작업 영역으로 파일을 끌어올 수 있습니다.
- 파일을 검색하고 검색 결과에서 해당 파일을 더블 클릭하여 열 수 있습니다.

- 탐색 영역의 파일 참조 섹션에서 파일 참조를 사용하여 파일을 열 수 있습니다. 파일을 더블 클릭하거나 작업 영역으로 끌어와서 파일을 엽니다.

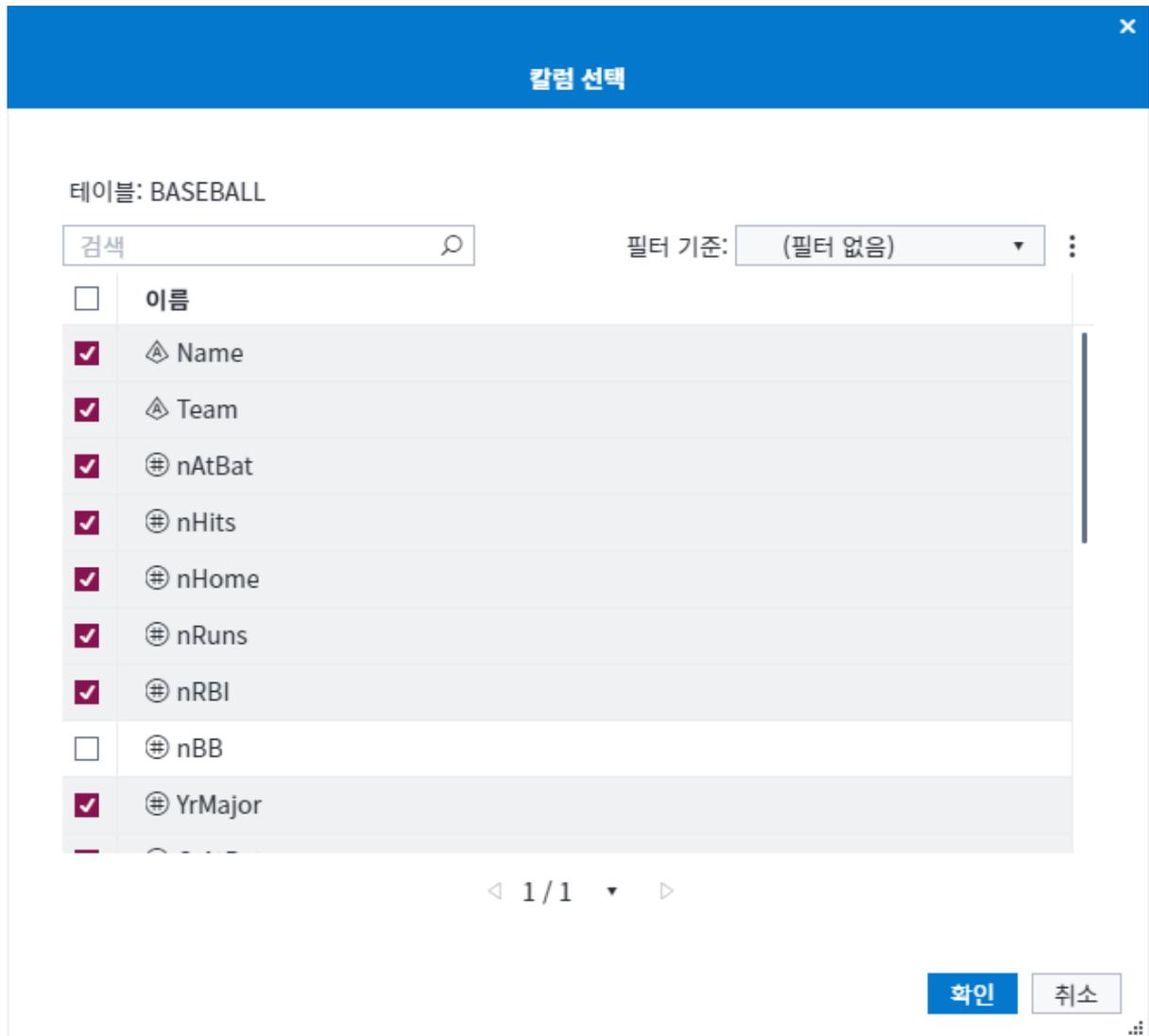
주: 폴더 이름에 "&"가 포함된 SAS 콘텐츠 폴더의 데이터 또는 암호로 보호된 데이터에 액세스할 수 없습니다.

주: 바이트 크기가 SAS Compute Server의 인코딩과는 다른 인코딩을 사용하여 생성된 테이블을 열 때는 LIBNAME 문에서 CVP(문자 변수 패딩) 엔진을 사용해야 합니다. 자세한 내용은 [Avoiding Character Data Truncation by using the CVP Engine\(SAS National Language Support \(NLS\): Reference Guide\)](#)을 참조하십시오.

테이블을 열면 기본적으로 테이블의 처음 100개 칼럼이 표시됩니다. 표시할 수 있는 칼럼의 개수는 **우선 설정** 창의 **테이블** 페이지에 있는 **표시된 칼럼** 옵션에 따라 결정됩니다. 자세한 내용은 ["테이블 우선 설정 지정" \(182페이지\)](#)에서 확인하십시오.

테이블 뷰어에 포함할 칼럼을 지정하려면 도구 모음에서  을 클릭하고 **칼럼 선택**을 선택합니다. **칼럼 선택** 창을 사용하여 칼럼을 검색하고 데이터의 특정 칼럼을 표시할지 숨길지를 지정할 수 있습니다. 테이블 뷰어에서 칼럼을 마우스 오른쪽 버튼으로 클릭하고 **숨기기**를 선택하여 칼럼을 숨길 수도 있습니다.

주: **칼럼 선택**에서  을 클릭하여 칼럼 이름 또는 레이블, 또는 둘 모두를 표시할지와 페이지당 표시할 칼럼 수, 칼럼 정렬 방법을 지정할 수 있습니다. 해당 설정은 **칼럼 선택** 창에만 영향을 미칩니다.



기본적으로 칼럼 이름이 표시되지만 도구 모음에서 **레이블 표시** 아이콘을 클릭하고 **레이블 표시**를 선택하여 칼럼 레이블을 표시하도록 선택할 수도 있습니다.

	Name	Team	nAtBat		
1	Allanson, Andy	Cleveland	293		
2	Ashby, Alan	Houston	315		
3	Davis, Alan	Seattle	475		
4	Dawson, Andre	Montreal	496		
5	Galarraga, Andres	Montreal	321		
6	Griffin, Alfredo	Oakland	594		
7	Newman, Al	Montreal	185		
8	Salazar, Argenis	Kansas City	298		
9	Thomas, Andres	Atlanta	323		
10	Thornton, Andre	Cleveland	401		
11	Trammell, Alan	Detroit	574		
12	Trevino, Alex	Los Angeles	201		
13	Van Slyke, Andy	St Louis	418	113	
14	Wiggins, Alan	Baltimore	239	60	
15	Almon, Bill	Pittsburgh	196	43	
16	Beane, Billy	Minneapolis	183	39	
17	Bell, Buddy	Cincinnati	568	158	
18	Biancalana, Buddy	Kansas City	190	46	

주: 기본적으로 테이블 뷰어는 테이블 행의 총 개수와 필터링된 행의 총 개수(데이터를 필터링했을 때)를 표시합니다. 그러나 SAS Studio에서 성능에 영향을 주지 않고 행 개수를 결정할 수 없으면 행 개수는 표시되지 않습니다. 관계형 테이블, 뷰 및 Hadoop 데이터에 대해 행 개수를 사용할 수 없습니다.

칼럼을 테이블 뷰어 또는 칼럼 영역의 새로운 위치에 끌어 놓아 테이블 뷰어의 칼럼 순서를 변경할 수 있습니다. 하나 이상의 칼럼을 기준으로 데이터를 정렬하고 고정할 수도 있습니다. 자세한 내용은 “데이터 정렬 및 고정” (113페이지)에서 확인하십시오.

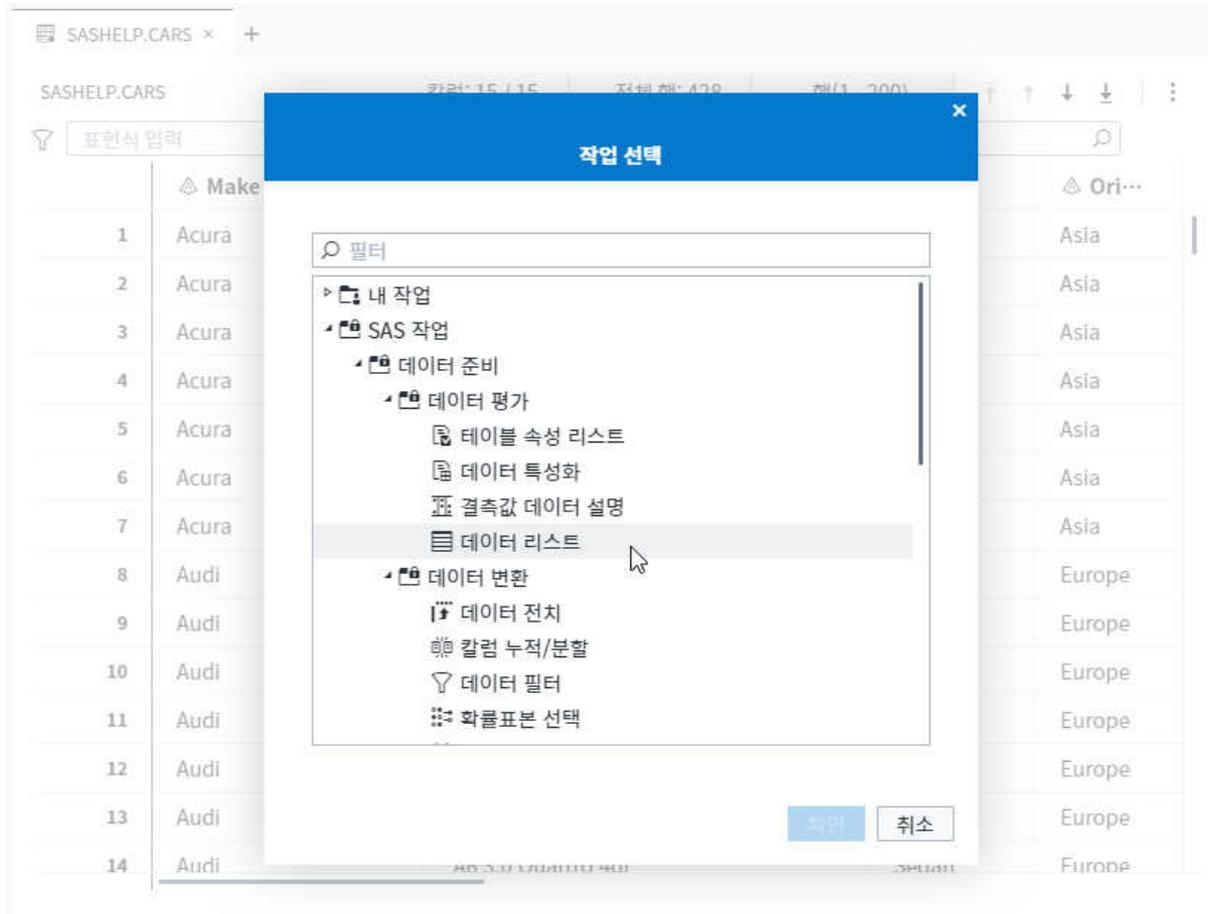
	Make	Model	Type	DriveTrain	Origin
1	Acura	MDX	SUV	Asia	
2	Acura	RSX Type S 2dr	Sedan	Asia	Front
3	Acura	TSX 4dr	Sedan	Asia	Front
4	Acura	TL 4dr	Sedan	Asia	Front
5	Acura	3.5 RL 4dr	Sedan	Asia	Front
6	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front
7	Acura	NSX coupe 2dr manual S	Sports	Asia	Rear
8	Audi	A4 1.8T 4dr	Sedan	Europe	Front
9	Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front
10	Audi	A4 3.0 4dr	Sedan	Europe	Front
11	Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All
12	Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All
13	Audi	A6 3.0 4dr	Sedan	Europe	Front
14	Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All

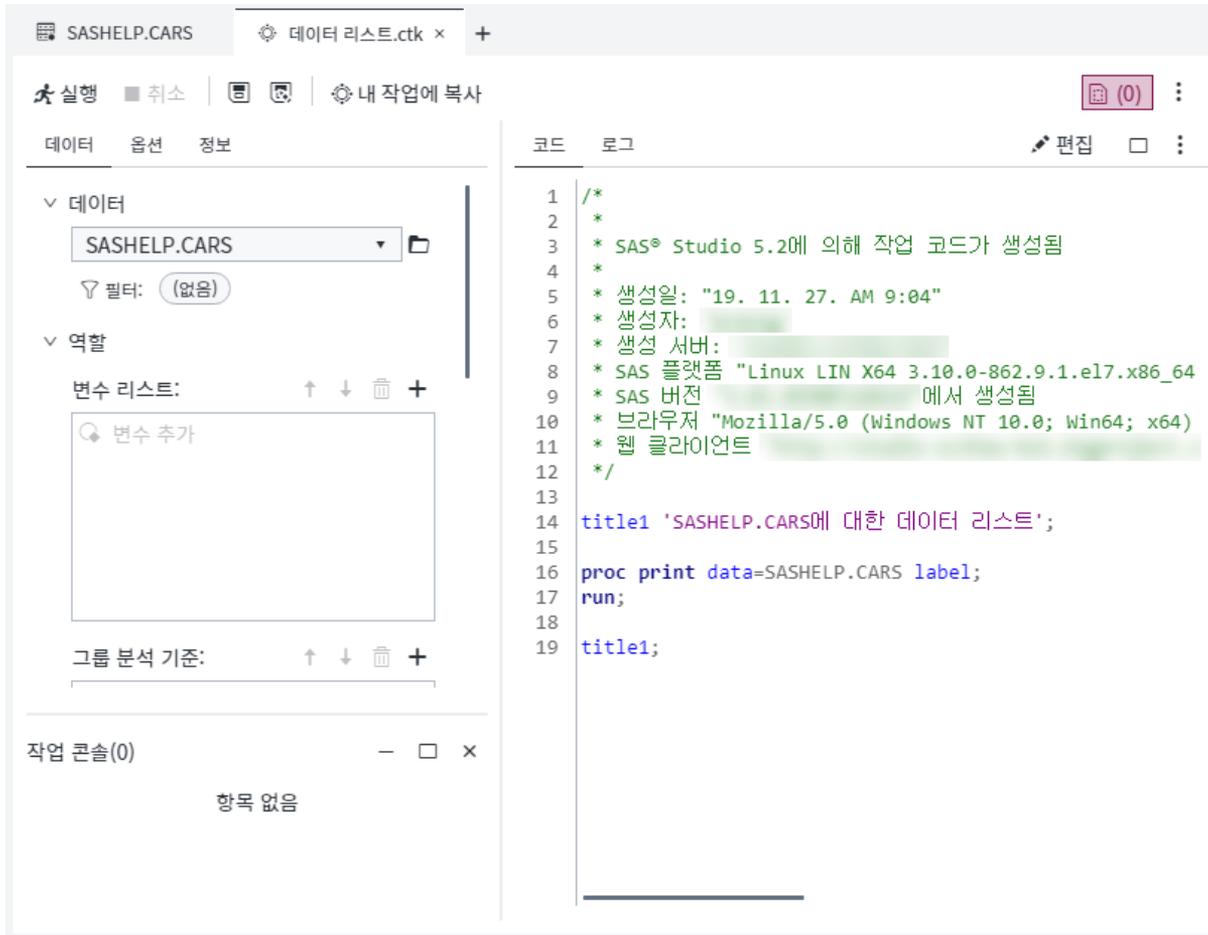
테이블을 생성하는 데 사용된 코드 보기

SAS Studio에서는 옵션을 선택하고 테이블을 원하는 모양으로 사용자 정의할 때 사용 가능한 SAS 코드가 자동으로 생성됩니다. 코드를 보려면 도구 모음에서 **⋮**를 클릭하고 **코드 생성**을 선택합니다. 새로운 프로그램 창에 테이블 뷰어의 테이블 뷰를 생성하는 데 사용된 코드가 나타납니다. 프로그램은 코드의 복사본으로, 더 이상 원래 코드와 연관되지 않습니다. 코드를 편집해도 테이블 뷰어에 표시된 데이터에는 영향이 없으며, 테이블 뷰어를 수정해도 코드 내용에는 영향이 없습니다.

테이블 뷰어의 작업 또는 질의에서 데이터 열기

테이블 뷰어를 사용하여 현재 보고 있는 데이터가 있는 작업 또는 질의를 열 수 있습니다. 입력 소스로 데이터가 자동으로 선택된 채 작업 또는 질의가 열립니다. 현재 선택된 데이터로 작업 또는 질의를 열려면 도구 모음에서 **⋮**를 클릭하고 **작업에서 열기** 또는 **질의에서 열기**를 선택합니다.





데이터 새로 고침

테이블 뷰어 새로 고침을 수행하여 테이블에 표시된 데이터를 업데이트할 수 있습니다. 데이터 새로 고침을 수행할 때 필터 또는 정렬 순서와 같이 지정한 모든 설정을 재설정할지 아니면 현재 설정을 유지할지 선택할 수 있습니다.

데이터를 새로 고침하여 설정을 재설정하려면 을 클릭하고 **테이블**을 선택합니다.

데이터를 새로 고침하여 설정을 유지하려면 을 클릭하고 **데이터**를 선택합니다.

데이터 정렬 및 고정

테이블 뷰어에서 칼럼 머리글을 마우스 오른쪽 버튼으로 클릭하여 해당 칼럼을 기준으로 데이터를 정렬하거나 데이터를 해당 칼럼에 고정할 수 있습니다. 데이터를 알파벳 오름차순 또는 내림차순으로 정렬하거나 데이터 테이블에 나타나는 순서대로 칼럼을 표시할 수 있습니다.

데이터를 정렬하려면 알맞은 칼럼을 오른쪽 마우스 버튼으로 클릭하고 **정렬** ⇒ **오름차순** 또는 **정렬** ⇒ **내림차순**을 선택합니다. 데이터가 하나 이상의 칼럼을 기준으로 이미 정렬되어 있으면, 칼럼을 마우스 오른쪽 버튼으로 클릭하고 **정렬** ⇒ **정렬에 추가(오름차순)** 또는 **정렬** ⇒ **정렬에 추가(내림차순)**을 선택하여 정렬 기준에 다른 칼럼을 추가할 수 있습니다.

칼럼을 고정하여 나머지 칼럼을 스크롤할 때 해당 칼럼과 그 왼쪽의 모든 칼럼이 계속 표시 되도록 하려면 적절한 칼럼을 마우스 오른쪽 버튼으로 클릭하고 **칼럼을 여기에 고정**을 선택합니다. 선택한 칼럼만 고정하고 테이블 뷰어의 가장 왼쪽 위치로 이동하려면 칼럼을 마우스 오른쪽 버튼으로 클릭하고 **칼럼 고정**을 선택합니다.

	Make	Model	Type	Origin	DriveTrain
1	Acura	MDX			All
2	Acura	RSX Type S 2dr			front
3	Acura	TSX 4dr			front
4	Acura	TL 4dr			front
5	Acura	3.5 RL 4dr		Asia	Front
6	Acura	3.5 RL w/Navigation		Asia	Front
7	Acura	NSX coupe 2dr manu		Asia	Rear
8	Audi	A4 1.8T 4dr	Sedan	Europe	Front
9	Audi	A41.8T convertible 2dr	Sedan	Europe	Front
10	Audi	A4 3.0 4dr	Sedan	Europe	Front
11	Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All
12	Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All
13	Audi	A6 3.0 4dr	Sedan	Europe	Front
14	Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All

데이터 필터링

데이터 필터링 정보

데이터 값에 따라 특정 기준을 충족하는 행만 표시하도록 데이터를 필터링할 수 있습니다. 단일 칼럼에 기반하여 빠른 필터를 생성하거나 하나 이상의 칼럼으로 필터 표현식을 생성하여 테이블에서 데이터를 필터링할 수 없습니다.

주: 소수점 이하 성분이 있는 값을 포함하는 숫자 칼럼에서 **Equal to** 연산자로 필터를 생성할 때에는 값 범위를 사용하여 필터를 생성해야 반올림 오차가 없는지 확인할 수 있습니다. 예를 들어 데이터에 **123.45678**의 값을 포함한다고 가정합니다. 해당 값을 포함한 행을 표시하는 필터를 생성하려면 다음 필터 표현식을 사용해야 합니다.

(column-name >= 123.456775 and column-name < 123.456785)

주: 필터를 생성할 때 값을 검색하려면 출력형식이 지정되지 않은 값을 사용하십시오.

빠른 필터 생성

칼럼 머리글을 마우스 오른쪽 버튼으로 클릭하고 **빠른 필터**를 선택하여 데이터에서 단일 칼럼에 대한 빠른 필터를 생성할 수 있습니다. 필터 추가 창에서 사용하려는 연산자와 값을 지정합니다. 값을 입력하거나 칼럼 값 리스트에서 선택할 수 있습니다.

필터에서 사용 중인 칼럼의 데이터 유형에 따라 다음 옵션 중에서 선택할 수 있습니다.

- **대소문자 구분** - 지정하는 값의 대소문자와 일치하는 행만 검색합니다. 이 옵션이 선택되지 않으면 UPPER 함수가 표현식에 적용됩니다. 이 옵션은 기본적으로 선택되어 있지 않으며 문자에 대해서만 사용할 수 있습니다.
- **문자열 따옴표로 묶기** - 값을 작은따옴표로 묶습니다. 이 옵션은 기본적으로 선택되어 있으며 문자 칼럼에 대해서만 사용할 수 있습니다. 필터 실행 중 매크로 변수 또는 계산된 다른 값을 사용하려면 이 옵션을 취소해야 합니다.
- **매크로 허용** - 숫자 칼럼에서 필터에 문자 값을 입력할 수 있습니다. 이 옵션은 기본적으로 선택되어 있지 않으며 숫자 칼럼에 대해서만 사용할 수 있습니다.
- **원시 값 사용** - 출력형식이 지정된 값 대신 출력형식이 지정되지 않은 숫자, 날짜, 시간 및 날짜시간 값을 사용합니다. 이 옵션은 기본적으로 선택되어 있으며 숫자, 날짜, 시간 및 날짜시간 칼럼에 대해서만 사용할 수 있습니다. 출력형식 지정된 값을 기반으로 필터를 만들려면 이 옵션을 선택 취소합니다.

팁 필터에 원시 값을 사용하는 것이 좋습니다. 출력형식이 지정된 원시 값을 사용하기 때문에 필터가 제대로 작동하지 않는 경우에는 출력형식이 지정되지 않은 원시 값을 필터링해보십시오.

The screenshot shows a SAS Data Studio interface with a 'PRDSALE' table. A '필터 추가' (Add Filter) dialog box is open, allowing the user to define a filter condition. The condition is set to 'Equal to' with a value of '\$925.00'. There are also checkboxes for '매크로 허용' (Allow macros) and '원시 값 사용' (Use raw values). The dialog box has buttons for '재설정' (Reset), '필터' (Filter), and '취소' (Cancel). The background table shows columns for 'DIVISION' and 'PF' with values like 'EDUCATION' and 'FURN'.

필터 표현식 생성

필터 표현식을 생성하여 하나 이상의 칼럼을 기준으로 데이터를 필터링할 수 있습니다. **테이블 행 필터링** 창에서 표현식 빌더를 사용하여 필터 표현식을 생성하려면 을 클릭합니다. 자세한 내용은 ["표현식 구성" \(187페이지\)](#)에서 참조하십시오.

테이블 행 필터링

데이터 함수
저장 취소

필터

- BASEBALL
 - Name
 - Team
 - nAtBat
 - nHits
 - nHome
 - nRuns
 - nRBI**
 - nBB
 - YrMajor
 - CrAtBat
 - CrHits
 - CrHome
 - CrRuns
 - CrRbi
 - CrBB
 - League
 - Division
 - Position
 - nOuts
 - nAssts
 - nError
 - Salary

+ - * ÷ ** || () ' " , and

AND OR NOT = < > <= >= | 고급 연산자

1 nRBI > 35

값: 33, 34, **35**, 36, 37, 38, 39, 41, 42, 43, 44

값 필터링: Equal Value | 대소문자 구분 문자열 따옴표로 묶기

SASHELP.BASEBALL x +					
BASEBALL 테이블 행: 322 칼럼: 24 / 24 행(1 - 200) (필터링됨) ↑ ↓ ↺ ⋮					
nRBI > 35					
	Name	Team	nAtBat	nRBI	
1	Ashby, Alan	Houston	315	38	
2	Davis, Alan	Seattle	479	72	
3	Dawson, Andre	Montreal	496	78	
4	Galarraga, Andres	Montreal	321	42	
5	Griffin, Alfredo	Oakland	594	51	
6	Thornton, Andre	Cleveland	401	66	
7	Trammell, Alan	Detroit	574	75	
8	Van Slyke, Andy	St Louis	418	61	
9	Bell, Buddy	Cincinnati	568	75	
10	Bochte, Bruce	Oakland	407	43	

데이터 가져오기

SAS Studio에 데이터 가져오기 정보

SAS Studio에는 두 가지 유형의 가져오기 기능이 있습니다. 빠른 가져오기를 사용하면 몇 가지 옵션을 선택하여 파일 시스템 또는 SAS 콘텐츠에서 데이터를 빠르게 가져올 수 있습니다.

데이터를 플로우로 가져올 수도 있습니다. 이 가져오기 기능은 보다 강력하며 추가 옵션을 제공합니다. 자세한 내용은 [“데이터 가져오기” \(118페이지\)](#)에서 참조하십시오.

SAS Studio에 다음과 같은 유형의 데이터 파일을 가져올 수 있습니다.

- 싼표 구분(파일*.csv)
- dBASE(dBase III, III+, IV, V)
- 구분된 파일(DLM)
- Stata 파일(DTA)
- JMP 파일(JMP)
- Paradox DB 파일(DB)
- SPSS 파일(SAV)
- 탭 구분(*.TSV)
- XLS(Microsoft Excel 5.0, 95, 97, 2000-2003)
- XLSX(Microsoft Excel 2007 이상 통합문서)

데이터 파일이 로컬 컴퓨터에 저장되는 경우 파일을 가져오려면 먼저 SAS Studio에 파일을 업로드해야 합니다. 자세한 내용은 [“탐색기 사용 ” \(8페이지\)](#)에서 확인하십시오.

다른 로케일에 있는 데이터를 정확히 가져오는지 여부는 SAS 서버가 가져오는 데이터 로케일을 지원하는지에 따라 달라집니다. 현재 로케일에 있는 문자와 다른 문자가 포함된 데이터를 가져오려면 유니코드(UTF-8) 서버를 사용하여 데이터를 가져옵니다. UTF-8 서버를 사용하지 않고 데이터 로케일이 지원되지 않으면 가져온 데이터에 지원되지 않는 값이 물음표(?)로 나타날 수 있습니다. [기본 텍스트 인코딩 옵션 설정에 대한 자세한 내용은 “일반 우선 설정 지정” \(175페이지\)](#)에서 참조하십시오.

주: FTP 파일 바로 가기에만 있는 파일인 원격 파일을 가져올 수 없습니다.

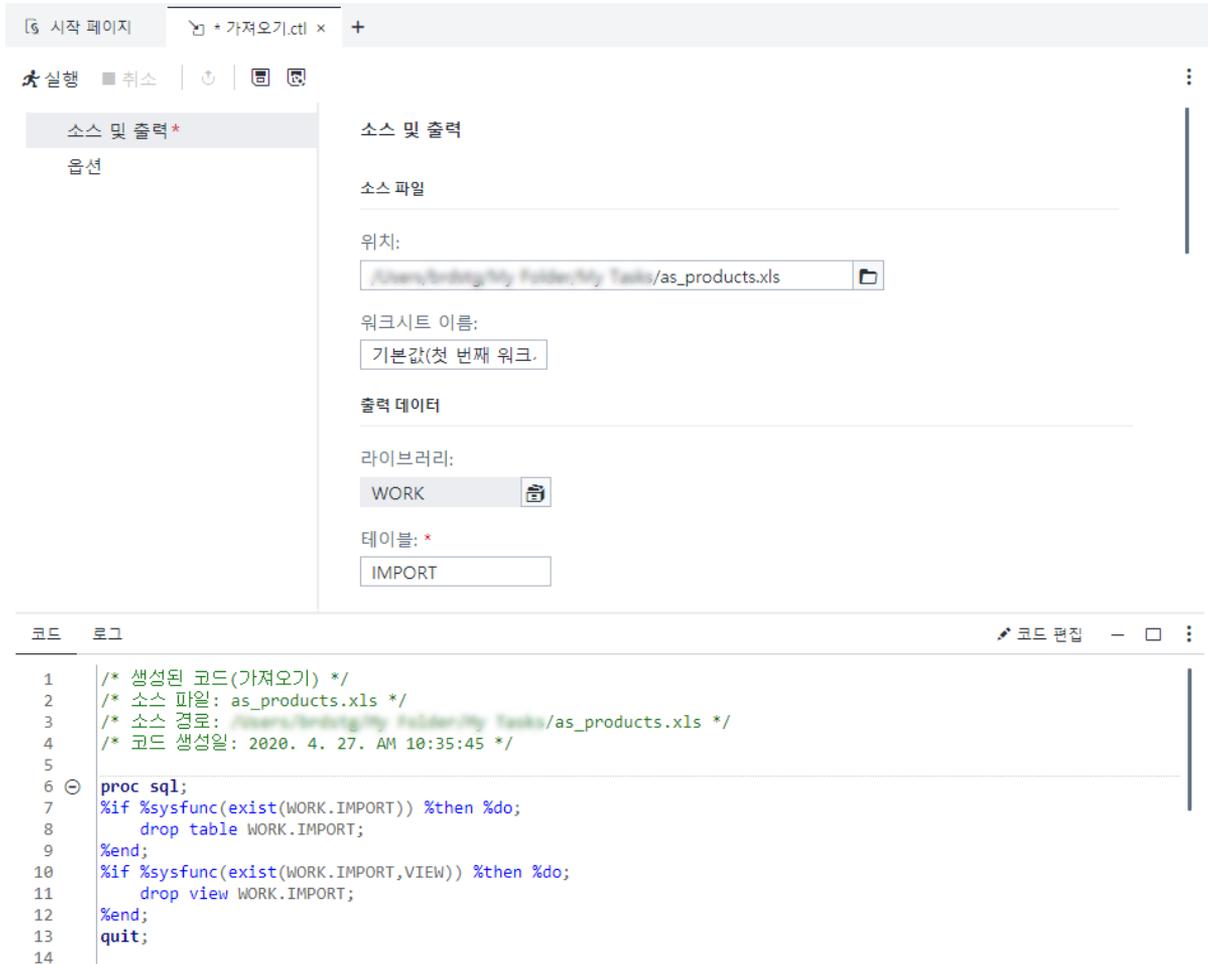
Excel 워크시트 가져오기

Excel 워크시트를 가져오려면 다음을 수행합니다.

- 1 **새로 만들기** ⇨ **빠른 가져오기**를 선택합니다.

- 2 작업 영역에서 **서버 파일 선택**을 클릭합니다.
- 3 가져올 파일을 선택하고 **열기**를 클릭합니다. Import.ctf 탭에 이 Excel 파일의 이름과 위치가 표시됩니다.

이 예에서는 as_products.xls 파일 가져오기를 보여줍니다.



- 4 특정 워크시트에서 데이터를 가져오려면 **워크시트 이름** 상자에 워크시트 이름을 입력합니다. 기본적으로 SAS Studio는 첫 번째 워크시트에서 데이터를 가져옵니다.
- 5 출력 데이터셋을 저장할 위치를 지정하려면  을 클릭합니다. 기본적으로 출력 데이터셋은 임시 위치인 Work 라이브러리에 저장됩니다. SAS Studio를 종료하면 이 라이브러리에 있는 내용은 삭제됩니다.
- 6 **옵션**을 클릭합니다.
 - 파일 유형을 선택합니다.
 - 워크시트 첫 번째 행의 데이터 값에서 SAS 변수 이름을 생성하려면 **SAS 변수 이름 생성**을 선택합니다. 입력 파일의 첫 번째 행에 있는 데이터 값을 읽고 SAS 이름에 유효하지 않은 특수 문자(예: 공백)를 포함하는 경우 SAS가 문자를 밑줄로 변환합니다.
 - 인코딩을 지정합니다.
- 7 Excel 워크시트를 가져오려면  을 클릭합니다.

구분된 파일 가져오기(CSV, TSV, DLM)

주: 탭으로 구분된 파일을 가져오려면, 파일 이름은 TSV 확장자를 가지고 있어야 합니다.

구분자로 구분된 파일을 가져오려면 다음을 수행합니다.

- 1 **새로 만들기** ⇨ **빠른 가져오기**를 선택합니다.
- 2 작업 영역에서 **서버 파일 선택**을 클릭합니다.
- 3 가져올 파일을 선택하고 **열기**를 클릭합니다. Import.ctl 탭에 이 텍스트 파일의 이름과 위치가 표시됩니다.

이 예에서는 CommaSep.txt라는 텍스트 파일 가져오기를 보여줍니다.

The screenshot shows the SAS Studio interface. At the top, there's a tab for '가져오기.ctl'. Below it, the '소스 및 출력' (Source and Output) section is active, showing options for the import process. The '소스 파일' (Source File) field is set to '/Users/brd@.../My Folder/My Tasks/CommaSep.txt'. The '행 구분자의 끝' (End of Line Separator) is set to '기본값' (Default). The '라이브러리' (Library) is set to 'WORK', and the '테이블' (Table) is set to 'IMPORT1'. Below this, the '소스 및 출력' section shows the generated SQL code:

```

1 /* 생성된 코드(가져오기) */
2 /* 소스 파일: CommaSep.txt */
3 /* 소스 경로: /Users/brd@.../My Folder/My Tasks/CommaSep.txt */
4 /* 코드 생성일: 2020. 4. 27. AM 10:45:01 */
5
6 proc sql;
7   %if %sysfunc(exist(WORK.IMPORT1)) %then %do;
8     drop table WORK.IMPORT1;
9   %end;
10  %if %sysfunc(exist(WORK.IMPORT1,VIEW)) %then %do;
11    drop view WORK.IMPORT1;
12  %end;
13  quit;
14

```

- 4 출력 데이터셋을 저장할 위치를 지정하려면  을 클릭합니다. 기본적으로 출력 데이터셋은 임시 위치인 Work 라이브러리에 저장됩니다. SAS Studio를 종료하면 이 라이브러리에 있는 내용은 삭제됩니다.
- 5 **옵션**을 클릭합니다.

- 파일 유형을 지정합니다.
- 텍스트 파일 첫 번째 행에 있는 데이터 값에서 SAS 변수 이름을 생성하려면 **SAS 변수 이름 생성**을 선택합니다. 입력 파일의 첫 번째 행에 있는 데이터 값을 읽고 SAS 이름에 유효하지 않은 특수 문자(예: 공백)를 포함하는 경우 SAS가 문자를 밑줄로 변환합니다.
- **구분자** 상자에서 파일에 있는 값에 대한 구분자를 입력합니다. 기본 구분자는 공백입니다.

주: 16진수 값을 사용하여 구분자를 지정하면 **다음표 구분자** 체크 박스를 선택할 필요가 없습니다.

- 구분자로 구분된 텍스트 파일에 있는 지정된 행에서 데이터를 읽으려면 **행에서 데이터 읽기 시작** 상자에서 시작 행을 입력합니다. 텍스트 파일 위쪽에 주석이 있거나 파일 첫 행이 칼럼 머리글일 때 이 옵션을 사용할 수 있습니다. 기본적으로 SAS Studio에서는 행 2에서 읽기를 시작합니다.
- SAS Studio에서 적합한 데이터 유형과 변수 길이를 결정하려면 **행 추측** 상자에 값을 입력합니다. 작업에서는 1번 행부터 지정한 행까지 입력 데이터 파일을 탐색합니다. 기본적으로 처음 20개 행이 탐색됩니다.
- 인코딩을 지정합니다.

6 데이터를 가져오려면  을 클릭합니다.

결과 탭을 클릭하여 가져온 데이터셋 속성을 봅니다.



CONTENTS 프로시저

데이터셋 이름	WORK.IMPORT	관측값	74
멤버 유형	DATA	변수	5
엔진	V9	인덱스	0
생성일	2020.04.27 11:04:42	관측값 길이	72
마지막 수정일	2020.04.27 11:04:42	삭제된 관측값	0
보호		압축여부	아니요
데이터셋 유형		정렬	아니요
레이블			
데이터 표현	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64, LINUX_POWER_64		
인코딩	utf-8 Unicode (UTF-8)		

엔진/호스트 관련 정보

데이터셋 페이지 크기	65536
데이터셋 페이지 번호	1
첫 번째 데이터 페이지	1
페이지 당 최대 관측값 수	908
첫 번째 데이터 페이지의 관측값 수	74
데이터셋 수리의 번호	0
파일 이름	..._sas-launcher-71b619c2-41
생성된 릴리즈	V.0400M0
생성된 호스트	Linux
Inode 번호	...
액세스 권한	rw-r--r--
소유자 이름	...
파일 크기	128KB
파일 크기 (바이트)	131072

#	변수	유형	길이	출력형식	입력형식
1	VAR1	숫자	8	BEST12.	BEST32.
2	VAR2	문자	31	\$31.	\$31.
3	VAR3	문자	20	\$20.	\$20.
4	VAR4	숫자	8	NLNUM12.	NLNUM32.
5	VAR5	문자	5	\$5.	\$5.

출력 데이터 탭을 클릭하여 새로운 SAS 데이터셋을 봅니다. 이 데이터셋이 예에 표시된 대로 Work 라이브러리에 있으면 영구 위치에 저장할 수 있습니다. Work 라이브러리에 있는 데이터는 임시이며 SAS Studio를 종료하면 삭제됩니다.

	1,Chai,10	boxes	x	20
1	2,Chang,24	-	12	oz
2	3,Aniseed	Syrup,12	-	550
3	4,Chef	Anton's	Cajun	Seasoning,48
4	5,Chef	Anton's	Gumbo	Mix,36
5	6,Grandma's	Boysenberr	Spread,12	-
6	7,Uncle	Bob's	Organic	Dried
7	8,Falserthwoods	Cranberry	Sauce,12	-
8	9,Mishi	Kobe	Niku,18	-
9	10,Ikura,12	-	200	ml
10	11,Queso	Cabrales,1	kg	pkg.,\$21.00,False
11	12,Queso	Manchego	La	Pastora,10
12	13,Konbu,2	kg	box,\$6.00,False	
13	14,Tofu,40	-	100	g
14	15,Genen	Shouyu,24	-	250
15	16,Pavlova,32	-	500	g
16	17,Alice	Mutton,20	-	1

DBMS 파일 가져오기

데이터베이스 관리 시스템(DBMS)에서 파일을 가져올 때 사용 가능한 옵션은 파일 유형에 따라 다릅니다. 지원되는 파일 유형 리스트는 [“SAS Studio에 데이터 가져오기 정보” \(118페이지\)](#)에서 참조하십시오.

- 1 **새로 만들기** ⇨ **빠른 가져오기**를 선택합니다.
- 2 작업 영역에서 **서버 파일 선택**을 클릭합니다.
- 3 가져올 파일을 선택하고 **열기**를 클릭합니다. Import.ctl 탭에 이 텍스트 파일의 이름과 위치가 표시됩니다.
- 4 출력 데이터셋을 저장할 위치를 지정하려면  을 클릭합니다. 기본적으로 출력 데이터셋은 임시 위치인 Work 라이브러리에 저장됩니다. SAS Studio를 종료하면 이 라이브러리에 있는 내용은 삭제됩니다.
- 5 **옵션**을 클릭합니다.
 - 파일 유형을 지정합니다.

- 텍스트 파일 첫 번째 행에 있는 데이터 값에서 SAS 변수 이름을 생성하려면 **SAS 변수 이름 생성**을 선택합니다. 입력 파일의 첫 번째 행에 있는 데이터 값을 읽고 SAS 이름에 유효하지 않은 특수 문자(예: 공백)를 포함하는 경우 SAS가 문자를 밑줄로 변환합니다.
 - 인코딩을 지정합니다.
- 6 데이터를 가져오려면 을 클릭합니다.

데이터 가져오기 작업 저장

사이트의 다른 사용자와 특정 파일을 가져오기 위한 설정을 공유할 수 있도록 데이터 가져오기 작업의 인스턴스를 저장할 수 있습니다. SAS Studio에서 인스턴스를 CTL 파일로 저장합니다. CTL 파일은 생성된 운영 환경과 동일한 환경에서 실행되어야 합니다. 예를 들어 Windows를 사용하여 CTL 파일을 생성하면 해당 CTL 파일은 Windows에서 실행되어야 합니다.

가져오기 작업을 저장하려면 다음을 수행합니다.

- 1 을 클릭합니다.

The screenshot shows the SAS Studio interface with the '소스 및 출력' (Source and Output) tab selected. The '소스 파일' (Source File) field is populated with the path '/Users/brndg/My Folder/My Tasks/CommaSep.txt'. The '라이브러리' (Library) is set to 'WORK' and the '테이블' (Table) is set to 'IMPORT1'. The code editor displays the following SAS code:

```

1 /* 생성된 코드(가져오기) */
2 /* 소스 파일: CommaSep.txt */
3 /* 소스 경로: /Users/brndg/My Folder/My Tasks/CommaSep.txt */
4 /* 코드 생성일: 2020. 4. 27. AM 10:45:01 */
5
6 proc sql;
7   %if %sysfunc(exist(WORK.IMPORT1)) %then %do;
8     drop table WORK.IMPORT1;
9   %end;
10  %if %sysfunc(exist(WORK.IMPORT1,VIEW)) %then %do;
11    drop view WORK.IMPORT1;
12  %end;
13  quit;
14
15
16
17  FILENAME REFFILE DISK = '/Users/brndg/My Folder/My Tasks / CommaSep.txt';
18
19  PROC IMPORT DATAFILE=REFFILE

```

- 2 다른 이름으로 저장 창에서 이름과 위치를 지정한 다음 **저장**을 클릭합니다. 파일이 CTL 확장자로 저장됩니다.

데이터 내보내기

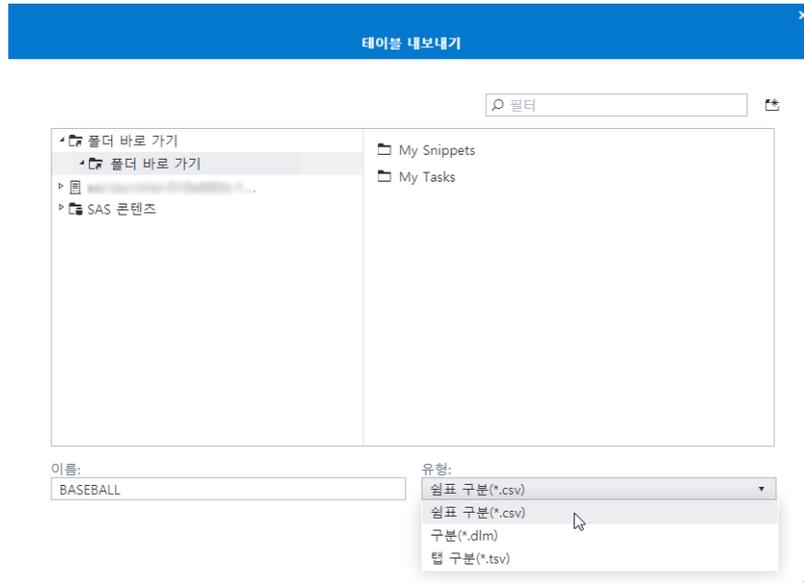
표준 영역에서 SAS Studio를 사용하여 데이터를 지정한 폴더에 다른 파일 유형으로 내보낼 수 있습니다.

주: 암호로 보호된 데이터는 SAS Studio에서 내보낼 수 없으며, 대화식 영역에서 데이터를 내보낼 수도 없습니다. 자세한 내용은 ["영역 이해" \(31페이지\)](#)에서 확인하십시오.

데이터를 내보내려면 다음을 수행합니다.

- 1 탐색 영역에서 **라이브러리**를 클릭하고 내보낼 파일을 검색합니다.

- 2 내보낼 파일을 마우스 오른쪽 버튼으로 클릭하고 **내보내기**를 선택합니다. **테이블 내보내기** 창을 엽니다.
- 3 내보내는 파일을 저장할 폴더를 선택합니다.
- 4 **이름** 상자에 내보내는 파일의 이름을 입력합니다.
- 5 **유형** 드롭다운 리스트에서 내보내는 파일의 출력형식을 선택합니다.



- 6 **저장**을 클릭하여 파일을 내보냅니다.

주: 데이터를 내보내면 전체 파일이 내보내집니다. 테이블 뷰어에서 데이터에 적용된 필터는 내보낸 데이터에는 적용되지 않습니다.

결과 및 출력 데이터 작업

결과 보기	127
기본 SAS Studio 출력	128
기본 결과 보기	128
생성된 데이터 다운로드	128
다른 사용자에게 결과 보내기	129
출력 데이터 뷰	131
SAS Output Delivery System 및 SAS ODS Statistical Graphics 정보	132

결과 보기

SAS Studio에서 작업 또는 프로그램을 실행하면 작업 영역에 결과가 표시됩니다. 기본적으로 결과는 HTML5 출력으로 생성되어 **결과** 탭에 표시됩니다. 다음은 결과에 대해 수행할 수 있는 작업입니다.

- **우선 설정** 창을 사용하여 PDF, Word, RTF, Excel, PowerPoint 등의 다른 출력형식을 추가합니다.
- 생성된 출력을 다운로드합니다. 생성된 결과를 사용 가능한 출력형식 중 하나로 다운로드할 수 있습니다.
- **우선 설정** 창을 사용하여 각 출력형식에 대한 기본 출력 스타일을 변경합니다.

기본 SAS Studio 출력

기본 결과 보기

SAS Studio에서 출력은 기본적으로 HTML5로 생성됩니다.

출력형식을 추가하려면 **우선 설정** 창을 사용하여 PDF 및 Word와 같은 다른 출력형식을 선택할 수 있습니다. 또한 사용 가능한 모든 ODS 스타일로 출력 기본 스타일을 변경할 수 있습니다. 자세한 내용은 **“결과 우선 설정 지정” (178페이지)**에서 확인하십시오.

기본적으로 HTML5 결과는 **결과** 탭에 표시되는 유일한 결과입니다. 결과를 볼 때 목차를 사용하여 다른 섹션을 탐색할 수 있습니다. 기본적으로 목차는 펼쳐져 있으므로 이동할 섹션을 바로 클릭할 수 있습니다.

The screenshot shows the SAS Studio interface with the 'Results' tab selected. The left pane shows the '데이터' (Data) section with 'SASHELP.CLASS' selected. The right pane shows a tree view of folders for different ages (Age=11 to Age=16) and a list of data tables for each age. The 'Age=11' section is expanded, showing a table with columns 'OBS', 'Name', 'Height', and 'Weight'.

Age=11			
OBS	Name	Height	Weight
1	Joyce	51.3	50.5
2	Thomas	57.5	85.0

Age=12			
OBS	Name	Height	Weight
3	James	57.3	83.0
4	Jane	59.8	84.5
5	John	59.0	99.5
6	Louise	56.3	77.0
7	Robert	64.8	128.0

Age=13			
OBS	Name	Height	Weight
8	Alice	56.5	84
9	Barbara	65.3	98
10	Jeffrey	62.5	84

생성된 데이터 다운로드

PDF 또는 RTF 등의 다른 출력형식으로 생성된 결과를 .csv 또는 .XML 등 다른 데이터 유형으로 다운로드하려면 다음을 수행합니다.

- 1 **결과** 탭에서 **⋮**을 클릭합니다.

2020. 4. 30. AM 11:17:50 (0)

코드 로그 결과 코드 편집

Print 프로시저

- Age=11
 - 데이터셋 WOR...
- Age=12
 - 데이터셋 WOR...
- Age=13
 - 데이터셋 WOR...
- Age=14
 - 데이터셋 WOR...
- Age=15
 - 데이터셋 WOR...
- Age=16
 - 데이터셋 WOR...

SASHELP.CLASS에 대한 데이터 리스트

Age=11

OBS	Name	Height	Weight
1	Joyce	51.3	50.5
2	Thomas	57.5	85.0

Age=12

OBS	Name	Height	Weight
3	James	57.3	83.0
4	Jane	59.8	84.5
5	John	59.0	99.5
6	Louise	56.3	77.0
7	Robert	64.8	128.0

2 다운로드와 결과 유형을 선택합니다.

다른 사용자에게 결과 보내기

이메일을 통해 결과의 복사본과 관련 코드 및 로그 파일을 다른 사용자에게 보낼 수 있습니다. HTML5, RTF 및 PDF 출력형식의 결과 파일뿐 아니라 결과와 연결된 코드 및 로그 파일도 보낼 수 있습니다. 코드는 SAS 프로그램 파일로 전송되며 로그 및 프로그램 요약 파일은 HTML5 파일로 전송됩니다. 이메일을 통해 파일을 보내려면 SMTP 서버에 대한 액세스가 필요합니다. 자세한 내용은 사이트 관리자에게 문의하십시오.

주: SAS Studio 이메일 메시지가 정크로 표시되면 [SAS Studio: Administrator's Guide](#)의 "Configuring the Mail Server"을 참조하십시오.

결과를 이메일로 보내려면 다음을 수행합니다.

1 프로그램, 질의 또는 작업 도구 모음에서 을 클릭합니다.

코드 로그 결과

코드 편집 □ ⋮

- Print 프로시저
 - Age=11
 - 데이터셋 WOR...
 - Age=12
 - 데이터셋 WOR...
 - Age=13
 - 데이터셋 WOR...
 - Age=14
 - 데이터셋 WOR...
 - Age=15
 - 데이터셋 WOR...
 - Age=16
 - 데이터셋 WOR...

SASHELP.CLASS에 대한 데이터 리스트

Age=11

OBS	Name	Height	Weight
1	Joyce	51.3	50.5
2	Thomas	57.5	85.0

Age=12

OBS	Name	Height	Weight
3	James	57.3	83.0
4	Jane	59.8	84.5
5	John	59.0	99.5
6	Louise	56.3	77.0
7	Robert	64.8	128.0

2 이메일을 선택합니다. 파일을 이메일로 전송 창이 나타납니다.

파일을 이메일로 전송 ×

받는 사람:

참조:

제목:

코드 로그 결과

노트를 추가합니다.

- 3 **받는 사람** 상자에 파일을 보낼 이메일 주소를 입력합니다. 주소는 세미콜론으로 구분합니다.
- 4 다른 주소로 이메일의 사본을 보내려면 **참조** 상자에 주소를 입력합니다.
- 5 **제목** 상자에 이메일의 제목을 입력합니다. 이메일 본문에 포함할 메시지를 추가할 수도 있습니다.
- 6 이메일에 첨부할 항목을 선택합니다. 기본적으로 코드가 선택됩니다. **결과**를 선택하면 생성된 모든 결과가 이메일에 포함됩니다.
- 7 **보내기**를 클릭하여 메시지와 첨부 파일을 전송합니다.

출력 데이터 뷰

테이블 또는 뷰를 생성하는 프로그램, 작업, 질의 또는 플로우를 실행하면 출력 데이터가 출력 데이터 탭에 표시됩니다. 탐색 영역의 **라이브러리** 섹션에서 여는 테이블 또는 뷰를 사용하는 동일한 방법으로 출력 데이터 탭의 데이터를 보고 사용할 수 있습니다. 자세한 내용은 **“테이블 뷰어 정보” (105페이지)**에서 확인하십시오.

기본적으로 프로그램, 작업 또는 플로우에서 여러 출력 테이블을 생성하는 경우 테이블 리스트가 축소 가능한 창에 표시됩니다.

주: 영역에서 출력 데이터 보기 옵션을 사용하여 영역 또는 간단한 메뉴에서 출력 테이블 리스트 표시 간에 전환할 수 있습니다. 자세한 내용은 **“일반 우선 설정 지정” (175페이지)**에서 참조하십시오.

다음과 같은 방법으로 영역에 출력 테이블이 표시되는 방식을 사용자 정의할 수 있습니다.

- 리스트 맨 위에 있는 필터 상자를 사용하여 테이블 이름에 대한 필터 기준을 지정하십시오.
-  을 클릭하고 올바른 순서를 선택하여 테이블의 정렬 순서를 지정합니다. 기본적으로 테이블은 생성된 순서대로 표시됩니다.

주: 생성 순서순으로 출력 데이터셋 정렬 옵션을 선택 취소하면 테이블을 오름차순 또는 내림차순으로만 정렬할 수 있습니다. 자세한 내용은 **“일반 우선 설정 지정” (175페이지)**에서 참조하십시오.

-  을 클릭하고 **라이브러리 이름 포함**을 선택 또는 선택 취소하여 각 테이블의 라이브러리 이름을 포함할지를 지정하십시오.
- 출력 테이블이 플로우에 있는 경우 **임시 포트 테이블 표시** 옵션을 사용하여 출력 포트와 연결되어 있지만 테이블 노드에 연결되어 있지 않은 테이블을 포함할지 여부를 지정합니다.

The screenshot shows the SAS Studio interface. At the top, there's a navigation bar with '시작 페이지' and '플로우' tabs. Below that, a toolbar contains various icons for execution and navigation. The main area is divided into two panes. The left pane shows a project tree with folders for 'CARS' and 'HORSEPOWER'. The right pane displays a data table with columns: 'make', 'Model', 'Type', 'Origin', and 'Drive'. A context menu is open over the table, showing options like '오름차순 정렬', '내림차순 정렬', '데이터 순서로 정렬', and '라이브러리 이름 포함'. Below the table, a 'SAS 프로그램' window shows the following code:

```

1 proc sort data=sashelp.cars out=CARS;
2   by make;
3   run;
4
5 data horsepower;
6   set sashelp.cars(where=(cylinders eq 8)) nobs=numobs;
7   format dollarPerHorse dollar12.2 runningAverageHorsepower 6.1;

```

SAS Output Delivery System 및 SAS ODS Statistical Graphics 정보

SAS ODS(Output Delivery System)는 다양한 출력형식 옵션을 사용하여 SAS 프로시저 및 DATA 스텝 출력을 생성, 저장 및 다시 생성할 수 있는 매우 유연한 기능을 제공합니다. ODS는 고유한 출력형식 지정 기능을 제공하며, ODS 없이 개별 프로시저나 DATA 스텝을 사용할 때는 이러한 기능을 사용할 수 없습니다.

SAS Studio에서는 출력이 웹 환경에서 올바르게 표시되도록 매우 특별한 ODS 옵션 및 GOPTIONS 문이 사용됩니다. 코드의 모든 ODS 옵션을 보려면 **옵션 > 우선 설정**을 클릭합니다. **우선 설정** 창에서 **SAS 프로그램 > 코드 및 로그**를 선택하고 **SAS 로그에 생성된 코드 표시** 옵션을 선택합니다.

주: 출력이 올바르게 표시되도록 하려면 생성된 코드에서 ODS 옵션 또는 GOPTIONS 문의 설정을 변경하지 마십시오.

SAS ODS Statistical Graphics는 SAS Output Delivery System(ODS)의 확장형으로, 일반적으로 SAS ODS Graphics라고 합니다. ODS는 프로시저에서 생성된 모든 출력을 관리하며, HTML과 PDF를 비롯한 다양한 형태로 출력을 표시할 수 있게 해 줍니다.

대부분의 SAS 분석 프로시저는 ODS Graphics 기능을 사용하여 그래프를 생성합니다. ODS Graphics는 GTL(그래프 템플릿 언어) 구문을 사용하므로 많은 복잡한 그래프를 생성할 수 있는 강력하고 유연한 기능을 제공합니다. GTL은 통계 그래프를 정의하기 위한 포괄적인 언어입니다.

SAS Studio의 Git 통합 이해

<i>SAS Studio의 Git 통합 정보</i>	134
<i>Git 프로파일 작업</i>	135
Git 프로파일 생성	135
Git 프로파일 삭제	136
<i>Git 레파지토리 복제 및 열기</i>	137
<i>로컬 레파지토리 생성</i>	139
<i>SAS Studio의 Git 레파지토리 탭 이해</i>	140
<i>커밋 히스토리 보기</i>	141
<i>변경 사항을 로컬 레파지토리에 커밋</i>	142
<i>파일 풀(pull) 및 페치(fetch)</i>	146
<i>파일 푸시</i>	146
<i>로컬 레파지토리 재설정</i>	147
<i>Git에서 브랜치 작업</i>	147
<i>브랜치 생성</i>	148
<i>브랜치 체크아웃</i>	149
<i>병합 및 리베이스 이해</i>	150
<i>브랜치 병합</i>	150
<i>현재 브랜치 리베이스</i>	151
<i>변경 사항 스테시</i>	151
<i>병합 충돌 해결</i>	152
<i>레파지토리 삭제</i>	154
<i>샘플 Git 워크플로우 시나리오</i>	155

SAS Studio의 Git 통합 정보

SAS Studio에는 Git 통합, 변경 사항 추적 시스템 및 여러 사용자 간의 버전 제어 관리가 포함되어 있습니다. Git은 GitHub 및 Bitbucket과 같은 여러 레파지토리 호스팅 서비스에 사용할 수 있습니다.

주: 이 문서에서는 사용자가 Git 기술과 레파지토리 호스팅 서비스에 기본적으로 익숙하다고 가정합니다.

다음은 SAS Studio에서 Git 기능을 사용하기 위한 요구 사항입니다.

- 표준 영역. 자세한 내용은 [“영역 이해” \(31페이지\)](#)에서 확인하십시오.
- RHEL(Red Hat Enterprise Linux), Windows WX6 또는 Windows W32 운영 체제. SUSE Linux Enterprise Server를 사용하고 있으면 추가 라이브러리를 설치해야 합니다. support@sas.com에서 SAS 기술 지원에 문의하십시오.

주: RHEL8 배포를 사용하는 경우 compat-openssl10 rpm을 설치해야 합니다. 자세한 내용은 [SAS Studio: Administrator's Guide의 “Using Git Functions on a RHEL 8 Deployment”](#)에서 확인하십시오.

- SAS 관리자가 Git 구성 속성을 지정해야 합니다. 자세한 내용은 [SAS Studio: Administrator's Guide의 “Configuration Properties for Git Integration”](#)에서 참조하십시오.

탐색 영역의 **Git 레파지토리** 섹션을 사용하여 SAS Studio 내에서 사용 가능한 Git 기능에 액세스할 수 있습니다. 여기에는 SAS Studio 내에서 레파지토리 복제(clone), 파일 변경 커밋(commit) 및 스택시(stash), 파일 풀(pull) 및 푸시(push), 레파지토리 히스토리 보기, 브랜치 생성 및 병합(create & merge), 브랜치 리베이스(rebase), 로컬 레파지토리에서 파일 간 기본 diff 수행 등이 포함됩니다. 기본적으로 SAS Studio에서는 SSH 키를 사용하여 레파지토리 호스팅 서비스를 인증합니다.

주: 다음 인증 수단을 사용하려면 시스템 관리자에게 문의하십시오.

- 사용자 이름과 비밀번호가 있는 HTTPS
- Kerberos. Kerberos 인증을 사용하려면 SAS 서버에 Kerberos가 설치 및 구성되어 있어야 합니다.

SAS 프로그램에서 SAS Git 기능을 사용하여 SAS에서 Git 레파지토리와 상호 작용할 수도 있습니다. 자세한 내용은 [SAS Functions and CALL Routines: Reference의 “Using Git Functions in SAS”](#)에서 참조하십시오.

Git 프로파일 작업

Git 프로파일 생성

SAS Studio의 Git 기능을 사용하려면 먼저 Git 프로파일을 생성해야 합니다. 기본적으로 SAS Studio에서는 SSH 키를 사용하여 인증합니다. SSH 인증을 사용하려면 공개 및 개인 SSH 키를 서버 파일 시스템에 업로드해야 합니다. SSH 키 생성에 대한 자세한 내용은 레파지토리 호스팅 서비스 문서를 참조하십시오.

팁 탐색 영역의 **탐색기** 섹션에서 적절한 디렉터리를 선택하고 **T**를 클릭하여 SSH 키 파일을 서버 파일 시스템에 업로드할 수 있습니다. **+**을 클릭하고 SSH 키 파일을 찾습니다. 파일을 선택하고 **열기**를 클릭합니다. **업로드**를 클릭하여 서버 파일 시스템의 디렉터리로 파일을 업로드합니다.

Git 프로파일을 생성하려면 **옵션** ⇨ **Git 연결 관리**를 선택한 다음 **+**을 클릭합니다. **프로파일 추가** 창이 나타납니다.

SSH 키를 사용하여 Git 프로파일을 생성하려면 알맞은 상자에 프로파일 이름, 사용자 이름 및 올바른 이메일을 입력하십시오. **탐색**을 클릭하여 공개 및 개인 SSH 키 파일을 찾거나 알맞은 상자에 경로 이름을 입력합니다. **확인**을 클릭하여 프로파일을 생성합니다.

✕
프로파일 추가

프로파일 이름: *

사용자 이름: *

이메일: *

공용 SSH 파일 경로:

📁

개인 SSH 파일 경로:

📁

HTTPS 또는 Kerberos 인증을 사용하여 Git 프로필을 생성하려면 시스템 관리자에게 사용하려는 인증에 알맞은 구성 속성을 활성화하는 방법에 대한 내용을 문의하십시오. 자세한 내용은 [SAS Studio: Administrator's Guide](#)의 "Configuration Properties for Git Integration"에서 참조하십시오.

팁 기본 프로파일을 지정하려면 **옵션** ⇨ **Git 연결 관리**를 선택하고 **프로파일**을 클릭합니다. 기본 프로파일로 사용할 프로파일을 선택하고 **기본값으로 설정**을 클릭합니다.

Git 프로파일 삭제

프로파일을 삭제하려면 **옵션** ⇨ **Git 연결 관리**를 선택하고 **프로파일**을 클릭합니다. 삭제할 프로파일을 선택하고 **🗑**을 클릭합니다.

프로파일이 활성 레파지토리에서 사용되고 있다면 활성 레파지토리를 닫아야 프로파일을 삭제할 수 있습니다. 다른 레파지토리와 연결된 프로파일을 삭제하면 **Git 연결 관리** 창을 사용하여 다른 프로파일과 연결하기 전까지는 해당 레파지토리를 열 수 없습니다.

Git 레파지토리 복제 및 열기

SAS Studio의 Git 레파지토리에 있는 파일에 액세스하려면 레파지토리를 서버에 파일 시스템의 로컬 레파지토리로 복제해야 합니다. 레파지토리를 복제하면 원격 레파지토리에 있는 모든 파일의 사본이 서버 파일 시스템에서 지정한 현재 작업 중인 디렉터리에 복사됩니다. SSH 또는 HTTPS 인증을 사용하여 원격 서버에서 레파지토리를 복제할 수 있습니다. 서버 파일 시스템의 로컬 레파지토리로도 복제할 수 있습니다.

주: 기존 Git 레파지토리가 서버 파일 시스템에 복제되었으면 해당 Git 레파지토리는 이미 로컬 레파지토리이며 액세스할 수 있습니다. 그러나 Git 레파지토리가 서버 파일 시스템에 없으면 복제해야 액세스할 수 있습니다.

탐색 영역의 **탐색기** 섹션에서 디렉터리로 이동하여 복제된 레파지토리의 파일에 액세스할 수 있습니다. 현재 작업 중인 디렉터리에서 파일을 변경한 후 해당 변경 사항을 로컬 레파지토리에 커밋할 수 있습니다.

주: 로컬 레파지토리의 파일에 적용한 변경 사항은 원격 레파지토리에 해당 변경 사항을 푸시(push)하지 않으면 다른 사용자에게 보이지 않습니다.

Git 레파지토리를 복제하려면 다음을 수행합니다.

- 1 탐색 영역에서  을 클릭한 다음 **레파지토리 복제** 를 클릭하거나 탐색 영역의 **Git 레파지토리** 섹션의 도구 모음에서 **복제** 를 클릭합니다. 프로파일 복제 창이 나타납니다.

팁 다음 세 가지 방법으로 레파지토리를 복제하는 옵션에 액세스할 수 있습니다.

- 탐색 영역의 **Git 레파지토리** 에서 **레파지토리 복제** 를 클릭합니다. 이 옵션은 탐색 영역의 **Git 레파지토리** 섹션에 나열된 레파지토리가 없을 때에만 사용 가능합니다.
- 탐색 영역의 **Git 레파지토리** 섹션의 도구 모음에서 **클론** 을 클릭합니다. 이 옵션은 탐색 영역의 **Git 레파지토리** 섹션에 레파지토리가 이미 나열되어 있을 때에만 사용 가능합니다.
- **옵션** ⇨ **Git 연결 관리** 를 선택합니다. **Git 연결 관리** 창에서 **레파지토리** 를 클릭합니다. **+** 을 클릭하고 **레파지토리 복제** 를 선택합니다.



주: 기존 로컬 레파지토리를 열려면 **레파지토리 추가**를 클릭하거나 탐색 영역의 **Git 레파지토리** 섹션에서 **추가** 도구 모음 버튼을 사용합니다. **폴더 선택** 창에서, Git 프로파일을 열고 지정하려는 레파지토리를 찾습니다.

2 **레파지토리** 상자에서 복제할 레파지토리의 위치를 입력합니다. 세 가지 방법으로 레파지토리를 복제할 수 있습니다.

- SSH 키를 사용하여 레파지토리 호스팅 서비스를 인증합니다. 다음 세 가지 중 하나와 유사한 형식으로 레파지토리 이름을 입력합니다.

git@github.com:folder/sasgit.git

또는

git@github.com:jquery/jquery-ui.git

주: 기본적으로 SAS Studio에서는 SSH 키를 사용하여 레파지토리 호스팅 서비스를 인증합니다.

- HTTPS 인증 사용 - 다음 예와 유사한 형식으로 레파지토리 이름을 입력합니다.

https://github.com/example/repo.git

- 로컬 레파지토리 사용 - 서버 파일 시스템의 로컬 레파지토리에 대한 전체 경로 이름을 입력합니다.

주: 로컬 레파지토리를 복제할 때 레파지토리에서 파일을 풀(pull)할 수 있지만 변경 사항을 푸시(push)할 수는 없습니다.

3 **서버 위치** 상자에서 복제된 레파지토리에 대한 서버 파일 시스템에 있는 폴더의 경로를 입력합니다.

주: 복제한 레파지토리에 사용할 폴더는 빈 폴더를 선택해야 합니다. 폴더에 파일이 있으면 복제 작업을 완료할 수 없습니다.

- 4 **프로파일** 드롭다운 리스트에서 Git 레파지토리와 연결할 때 사용할 프로파일을 선택합니다.

- 5 **복제**를 클릭하여 레파지토리를 복제합니다. Git 레파지토리 탭이 작업 영역에서 열립니다. 탐색 영역의 **탐색기** 섹션을 사용하여 복제된 레파지토리의 파일을 볼 수도 있습니다.

주: 탐색 영역의 **탐색기** 섹션에서 레파지토리의 이름을 바꿀 수 있습니다. 레파지토리 폴더를 마우스 오른쪽 버튼으로 클릭하고 **속성**을 선택합니다. **이름** 상자에서 새로운 이름을 입력합니다.

로컬 레파지토리 생성

탐색 영역의 **탐색기** 섹션에 있는 서버 시스템 폴더에서 로컬 레파지토리를 생성할 수 있습니다. 탐색 영역의 **Git 레파지토리** 섹션에서 로컬 레파지토리에 액세스하고 변경 사항을 추적하고 커밋할 수 있습니다. 또한 로컬 레파지토리를 원격 레파지토리와 연결하고 변경 사항을 원격 레파지토리에 푸시할 수도 있습니다.

주: SAS 콘텐츠 폴더에서는 로컬 레파지토리를 생성할 수 없습니다.

로컬 레파지토리를 생성하려면 다음을 수행합니다.

- 1 탐색 영역의 **탐색기** 섹션에서 로컬 레파지토리로 사용할 폴더를 마우스 오른쪽 버튼으로 클릭하고 **Git 초기화**를 선택합니다. 폴더 트리가 새로 고침되고 폴더는 트리에서 Git 레파지토리로 식별됩니다.
- 2 새로운 로컬 레파지토리를 보려면 탐색 영역에서 **+**을 클릭합니다. 자동으로 열리지 않으면 새로운 로컬 레파지토리를 더블 클릭하여 작업 영역에서 엽니다. 폴더의 모든 파일

이 레파지토리에 스테이징 해제된 변경 사항으로 추가된 것을 볼 수 있습니다. 파일을 레파지토리에 추가하려면 먼저 파일을 스테이징한 다음 커밋해야 합니다. 자세한 내용은 [“변경 사항을 로컬 레파지토리에 커밋” \(142페이지\)](#)에서 참조하십시오.

- 3 커밋된 파일을 원격 레파지토리에 푸시하려면 Git 레파지토리 탭에서 **푸시**를 클릭합니다. **원격 레파지토리 생성** 창에서 원격 레파지토리의 위치를 입력하고 **생성**을 클릭합니다.

SAS Studio의 Git 레파지토리 탭 이해

Git 레파지토리를 복제하거나 서버에서 기존 Git 레파지토리를 열면 Git 레파지토리 탭이 작업 영역에서 열립니다.

Git 레파지토리 탭에는 2개의 하위 탭 즉, 히스토리 및 커밋이 있습니다. 히스토리 하위 탭에는 레파지토리의 모든 변경 사항 또는 커밋의 히스토리가 표시되며, 선택된 커밋과 상위 커밋 간의 차이를 볼 수 있습니다. 커밋 하위 탭에서는 레파지토리의 파일에 대한 변경 사항을 스테이징, 스테이징 해제, 재설정 및 커밋할 수 있습니다. 또한 파일 버전 간의 차이도 볼 수 있습니다.

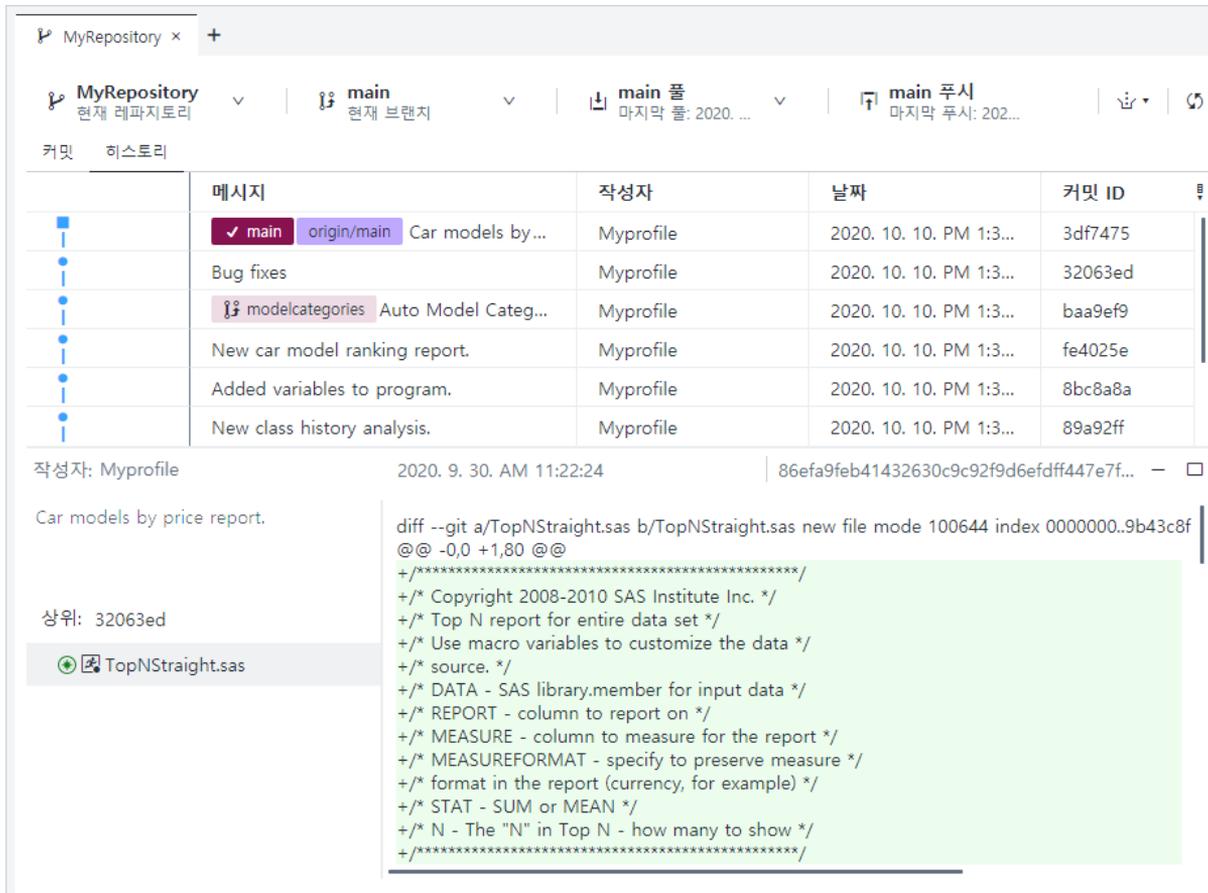
로컬 Git 레파지토리 탭을 열려면 탐색 영역에서 을 클릭하고 열고자 하는 레파지토리를 더블 클릭합니다. Git 레파지토리 탭이 작업 영역에서 열립니다.

주: SAS Studio에서는 한 번에 하나의 Git 레파지토리만 볼 수 있습니다.

기본 Git 레파지토리 도구 모음에서 다음 작업을 수행할 수 있습니다.

- 열고자 하는 로컬 레파지토리를 지정합니다.
- 로컬 브랜치를 체크아웃합니다.
- 원격 레파지토리에서 데이터를 풀(pull)하거나 페치합니다. 자세한 내용은 [“파일 풀 \(pull\) 및 페치\(fetch\)” \(146페이지\)](#)에서 참조하십시오.
- 커밋된 변경 사항을 원격 레파지토리에 푸시합니다. 자세한 내용은 [“파일 푸시” \(146페이지\)](#)에서 참조하십시오.
- 변경 사항을 스테이징하고 다시 적용합니다. 자세한 내용은 [“변경 사항 스테이징” \(151페이지\)](#)에서 참조하십시오.
- 로컬 레파지토리를 새로 고침합니다.

주: Git 레파지토리 탭에서 을 클릭하고 **새로 고침**을 선택하여 탐색 영역의 **Git 레파지토리** 섹션을 새로 고침할 수 있습니다.



커밋 히스토리 보기

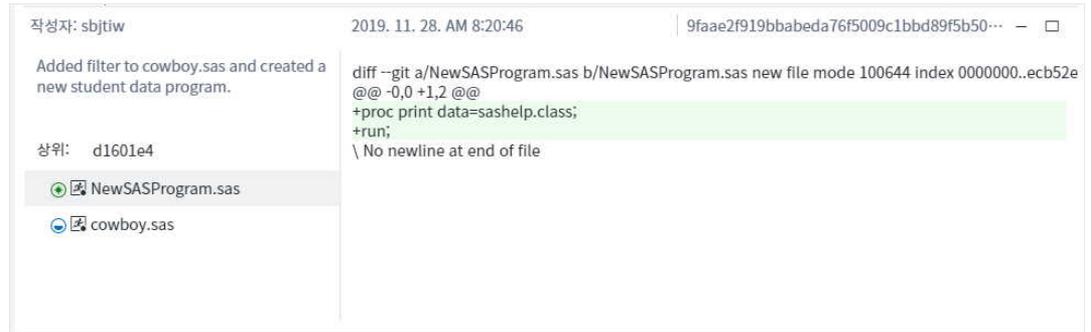
커밋 히스토리를 사용하여 레파지토리에 적용한 모든 변경 사항의 히스토리를 볼 수 있습니다. 특정 커밋의 상세 정보를 비롯해 두 커밋 간의 차이도 확인할 수 있습니다.

변경 사항을 작업 디렉터리의 파일에 저장한 후에는 로컬 Git 레파지토리 탭에 해당하는 탭이 작업 영역에서 열려 있는지 확인합니다. **히스토리** 하위 탭을 클릭합니다. 레파지토리에 적용한 모든 커밋의 리스트가 역 연대순으로 작업 영역에 표시됩니다. 각 커밋에 대해 커밋 메시지, 작성자, 커밋 날짜 및 시간, 커밋 ID가 기본적으로 표시됩니다. 표시되는 칼럼을 수정하려면 을 클릭하고 표시할 칼럼을 선택합니다.

주: 기본적으로 최대 1,000개의 행이 커밋 히스토리에 로드됩니다. **옵션** ⇨ **Git 연결 관리**를 선택하여 다른 제한을 지정할 수 있습니다. **옵션**을 클릭하고 **히스토리 로그에 로드된 최대 행** 옵션에 새로운 값을 입력합니다. 로드되는 행의 수를 늘리면 성능에 영향을 줄 수 있습니다.

특정 커밋의 상세 정보를 보려면 커밋 히스토리에서 커밋을 선택합니다. 커밋의 상세 정보가 작업 영역의 아래쪽 섹션에 표시됩니다. 상세 정보에서 상위 커밋 ID를 클릭하여 선택된 커밋 상위의 상세 정보를 볼 수 있습니다.

선택된 커밋과 상위 커밋의 변경 사항 간 차이를 보려면 작업 영역의 아래쪽 섹션에서 파일을 클릭합니다. 또한, 비교할 커밋을 선택하여 커밋 히스토리 리스트의 인접한 두 커밋을 비교할 수 있습니다.



주: 별도의 창에서 파일 차이를 보려면 작업 영역의 차이 섹션을 마우스 오른쪽 버튼으로 클릭하고 **새 창에서 열기**를 선택합니다.

변경 사항을 로컬 레파지토리에 커밋

로컬 Git 레파지토리에서 원격 레파지토리로 하나 이상의 변경 사항을 푸시(push)하기 전에 먼저 작업 디렉터리에서 로컬 레파지토리로 변경 사항을 커밋해야 합니다. 파일을 커밋하면 파일이 **스테이징 해제된 변경 사항** 영역에서 **스테이징된 변경 사항** 영역으로 이동되어 커밋에 포함될 파일이 스테이징됩니다. 그런 다음, 커밋을 설명하는 주석을 입력한 후 파일을 커밋합니다. 커밋할 파일을 선택할 때에는 서로 관련되어 있고 캡처할 프로젝트 개발 상태를 나타내는 변경 사항을 그룹화하는 것이 좋습니다.

주: 로컬 레파지토리에 변경 사항을 커밋할 때 해당 변경 사항은 다른 사용자에게 여전히 보이지 않지만 원격 레파지토리로 푸시(push)할 수 있습니다.

로컬 레파지토리에 변경 사항을 커밋하려면 다음을 수행합니다.

- 1 변경 사항을 작업 디렉터리의 파일에 저장한 후에는 로컬 Git 레파지토리 탭이 작업 영역에서 열려 있는지 확인합니다. **커밋** 하위 탭을 클릭합니다.
- 2 변경된 파일이 **스테이징 해제된 변경 사항** 영역에 나열되지 않으면  을 클릭하여 리스트를 새로 고칩니다.

스테이징 영역에 있는 파일은 변경 유형을 나타내는 다음과 같은 상태 아이콘에 따라 식별됩니다.

-  - 로컬 레파지토리에 추가될 새로운 파일
-  - 이름이 바뀐 파일
-  - 변경 사항이 있는 기존 파일
-  - 삭제된 파일
-  - 로컬 레파지토리의 다른 파일과 매우 유사하거나 동일한 콘텐츠가 있는 파일
-  - 병합 충돌이 있는 파일

커밋
히스토리

스태이징 해제된 변경 사항(2)
⬇ ⬆

- NewSASProgram.sas
- cowboy.sas

스태이징된 변경 사항(0)
⬆ ⬇

항목 없음

커밋 주석 입력

커밋

.....
 주: 파일 이름에 ~을 포함하고 있는 자동 저장된 파일은 스타이징 해제된 파일 리스트에 포함되지 않습니다.

- 3 커밋할 변경 사항을 하나 이상 선택하고  을 클릭합니다. 커밋할 파일을 선택할 때에는 서로 관련되어 있고 단일 커밋에 함께 포함된 변경 사항을 그룹화하는 것이 좋습니다.

팁 또한, 파일을 더블 클릭하여 스테이징 및 스테이징 해제할 수 있습니다.

.....
주: 스테이징하지 않은 변경 사항은 커밋에 포함되지 않습니다. 변경 사항은 커밋할 때 까지 작업 디렉터리 및 **스테이징 해제된 변경 사항** 영역에 남아 있습니다.
.....

커밋
히스토리

스테이징 해제된 변경 사항(0)
⬇ ⬇

항목 없음

스테이징된 변경 사항(2)
⬆ ⬆

NewSASProgram.sas	
cowboy.sas	

커밋 주석 입력

커밋

.....
 주: 변경된 모든 파일을 한 번에 스테이징하려면 을 클릭합니다.

팁 커밋 탭에서 파일을 클릭하여 스테이징 영역과 로컬 레파지토리에 있는 파일 버전 간의 차이를 볼 수 있습니다. 작업, 질의 및 플로우 파일에 대한 파일의 차이 점은 강조 표시되지 않으므로 커밋 주석에 변경 사항을 문서화하는 것을 고려해야 합니다.

스테이징 영역의 파일에 대한 변경 사항을 무시하고 파일을 로컬 레파지토리의 버전으로 재설정하려면 파일을 마우스 오른쪽 버튼으로 클릭하고 **재설정**을 선택합니다. 재설정된 파일이 작업 영역에서 열려 있으면, 로컬 레파지토리에 버전을 다시 로드하거나 편집된 버전을 유지하라는 메시지가 표시됩니다.

- 4 주석 상자에 커밋하는 파일의 변경 사항에 대한 설명을 입력하고 **스테이징된 항목 커밋**을 클릭합니다. 변경된 파일을 원격 레파지토리에 푸시(push)할 수 있습니다.

파일 풀(pull) 및 페치(fetch)

원격 레파지토리의 데이터는 두 가지 방법, 즉 풀(pull)과 페치(fetch)를 통해 로컬 레파지토리로 다운로드할 수 있습니다. 풀(pull) 기능을 사용하면 원격 레파지토리에 있는 파일을 다운로드할 수 있고 로컬 레파지토리를 업데이트하여 콘텐츠를 최신으로 유지할 수 있습니다. 로컬 레파지토리의 일부 파일에 변경 사항을 적용했다면 Git에서 원격 레파지토리의 콘텐츠를 로컬 파일과 병합합니다. 원격 레파지토리의 파일을 풀(pull)하려면 로컬 Git 레파지토리 탭이 작업 영역에서 열려 있는지 확인하고 **브랜치 이름 풀** ⇨ **풀**을 선택합니다.

주: SAS Studio에서는 원격 파일을 풀(pull)할 때 원격 및 로컬 레파지토리에 있는 파일 간 충돌이 발생하면 메시지가 표시됩니다. 충돌이 발생하면 충돌하는 파일은 **스테이징 해제된 변경 사항** 영역으로 이동됩니다. 충돌을 해결해야 다시 파일을 커밋할 수 있습니다. 원격 레파지토리에 풀(pull)했고 충돌이 없는 파일은 커밋을 위해 스테이징됩니다.

페치 기능은 원격 레파지토리에 있는 파일을 다운로드하지만 변경 사항을 현재 작업 브랜치에 병합하지는 않습니다. 원격 레파지토리의 파일을 페치하려면 로컬 Git 레파지토리에 대한 탭이 작업 영역에서 열려 있는지 확인합니다. 다음으로 **브랜치 이름 풀** ⇨ **페치**를 클릭하고 변경 사항을 특정 브랜치에서 페치할지 또는 모든 브랜치에서 페치할지를 지정합니다.

파일 푸시

현재 작업 중인 디렉터리에서 로컬 레파지토리로 변경한 파일을 스테이징 및 커밋한 후에는 원격 레파지토리로 해당 파일을 푸시(push)할 수 있습니다. 커밋한 변경 사항을 원격 레파지토리로 푸시(push)하면 로컬 레파지토리의 파일은 원격 레파지토리와 동기화됩니다.

파일을 원격 레파지토리로 푸시하려면 로컬 Git 레파지토리 탭이 작업 영역에서 열려 있는지 확인합니다. 도구 모음에서 **브랜치 이름 푸시**를 클릭합니다. 커밋 히스토리는 원격 레파지토리로 병합된 커밋한 변경 사항을 보여줍니다.

로컬 레파지토리 재설정

로컬 레파지토리를 이전 커밋으로 재설정하여 작업한 커밋을 실행 취소할 수 있습니다.

로컬 레파지토리를 재설정하려면 다음을 수행합니다.

- 1 로컬 레파지토리에 대해 커밋 히스토리가 열려 있는지 확인합니다. 재설정할 커밋을 마우스 오른쪽 버튼으로 클릭하고 **재설정**을 선택합니다. **로컬 레파지토리 재설정** 창이 나타납니다.
- 2 수행할 재설정 유형을 선택합니다.
 - **Soft** - 현재 브랜치를 이전 커밋으로 재설정합니다. 스테이징된 변경 사항(인덱스) 또는 현재 작업 중인 디렉터리는 변경되지 않습니다.
 - **Mixed** - 현재 브랜치를 이전 커밋으로 재설정하고 스테이징된 변경 사항(인덱스)을 재설정합니다. 현재 작업 중인 디렉터리는 변경되지 않습니다.
 - **Hard** - 현재 브랜치를 이전 커밋으로 재설정하고 스테이징된 변경 사항(인덱스)을 재설정하고 현재 작업 중인 디렉터리의 모든 변경 사항을 무시합니다. Hard 재설정은 실행 취소할 수 없습니다.
- 3 **확인**을 클릭하여 레파지토리를 재설정합니다.

Git에서 브랜치 작업

Git 작업은 브랜치에서 관리합니다. 기본적으로 원격 레파지토리의 기본 브랜치는 origin/main 브랜치로 식별됩니다. 로컬 레파지토리의 기본 브랜치는 기본적으로 main 브랜치로 식별됩니다. 추가 브랜치를 생성하고 브랜치를 병합할 수 있습니다. 커밋한 변경 사항을 원격 레파지토리로 푸시(push)하면 로컬 레파지토리의 브랜치 내 파일은 원격 레파지토리의 브랜치와 동기화됩니다. 히스토리 하위 탭을 사용하여 브랜치와 연결된 커밋 및 브랜치가 동기화되는지를 볼 수 있습니다. 자세한 내용은 ["커밋 히스토리 보기" \(141페이지\)](#)에서 참조하십시오.

이 예에서 main 브랜치는 origin/main 커밋보다 한 개 커밋만큼 앞에 있습니다. 이렇게 동기화가 되어 있지 않으면 "New graph program" 커밋이 아직 원격 레파지토리로 푸시되지 않았기 때문에 원격 origin/main 브랜치에 포함되지 않은 로컬 main 브랜치를 변경했음을 의미합니다. 자세한 내용은 ["파일 푸시" \(146페이지\)](#)에서 참조하십시오.

The screenshot shows the Git interface for 'MyRepository'. The current branch is 'main'. The commit history table is as follows:

커밋	메시지	작성자	날짜	커밋 ID
✓ main	New graph program	Myprofile	2020. 10. 10. PM 1:5...	000874f
origin/main	Car models by price report.	Myprofile	2020. 10. 10. PM 1:3...	3df7475
	Bug fixes	Myprofile	2020. 10. 10. PM 1:3...	32063ed
modelcategories	Auto Model Categ...	Myprofile	2020. 10. 10. PM 1:3...	baa9ef9
	New car model ranking report.	Myprofile	2020. 10. 10. PM 1:3...	fe4025e
	Added variables to program.	Myprofile	2020. 10. 10. PM 1:3...	8bc8a8a

"New graph program" 커밋이 원격 레파지토리에 푸시된 후에는 origin/main 및 main 브랜치가 동기화되고 커밋 히스토리에서 같은 행에 나열됩니다.

The screenshot shows the Git interface after pushing the 'New graph program' commit. The commit history table is updated as follows:

커밋	메시지	작성자	날짜	커밋 ID
✓ main origin/main	New graph program	Myprofile	2020. 10. 10. PM 1:5...	000874f
	Car models by price report.	Myprofile	2020. 10. 10. PM 1:3...	3df7475
	Bug fixes	Myprofile	2020. 10. 10. PM 1:3...	32063ed
modelcategories	Auto Model Categ...	Myprofile	2020. 10. 10. PM 1:3...	baa9ef9
	New car model ranking report.	Myprofile	2020. 10. 10. PM 1:3...	fe4025e
	Added variables to program.	Myprofile	2020. 10. 10. PM 1:3...	8bc8a8a

Git에서 브랜치 관리에 대한 자세한 내용은 레파지토리 호스팅 서비스 문서에서 참조하십시오.

브랜치 생성

진행 중인 다른 작업에서 현재 진행 중인 작업을 분리하기 위해 Git에서 브랜치를 생성할 수 있습니다. 브랜치를 생성할 때 해당 브랜치의 커밋은 브랜치를 다른 브랜치와 병합할 때까지 다른 커밋으로부터 분리됩니다.

주: 브랜치가 여러 개이면 Git 레파지토리 탭의 상단에 있는 **현재 브랜치** 드롭다운 리스트를 사용하여 작업할 브랜치를 체크아웃할 수 있습니다. 또한 커밋 히스토리에서 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **브랜치 체크아웃** ⇨ **브랜치 이름**을 선택할 수도 있습니다. 자세한 내용은 "**브랜치 체크아웃**" (149페이지)에서 참조하십시오.

브랜치를 생성하려면 다음을 수행합니다.

- 1 로컬 레파지토리에 대해 히스토리 하위 탭이 열려 있는지 확인합니다. 새로운 브랜치를 시작할 커밋을 마우스 오른쪽 버튼으로 클릭하고 **새로운 브랜치 생성**을 선택합니다. **브랜치 생성** 창이 나타납니다.
- 2 **브랜치 이름** 상자에 새로운 브랜치 이름을 입력합니다. 브랜치 이름에는 문자와 숫자, 하이픈 또는 밑줄과 같은 일부 특수 문자만 포함될 수 있습니다. 분기 이름도 다음 규칙을 따라야 합니다.
 - 이름에는 공백, 연속된 두 개의 마침표(..), 연속된 여러 개의 슬래시(/), at 기호 다음에 종괄호(@) 또는 " | ~ ^ : ? * [\ 문자가 포함될 수 없습니다.
 - 이름은 at 기호(@)로만 구성될 수 없습니다.
 - 이름은 슬래시(/)로 시작하거나 끝날 수 없습니다.
 - 이름은 마침표(.)로 끝날 수 없습니다.
- 3 브랜치를 생성하지만 현재 브랜치에 남아있으려면 **생성 후 체크아웃** 체크 박스를 선택 해제합니다. 이 옵션은 기본적으로 선택되어 있습니다.
- 4 브랜치를 생성할 때 커밋하지 않은 작업 디렉터리의 모든 변경 사항을 무시하려면 **중복 브랜치 이름이 존재하면 강제 적용** 체크 박스를 선택합니다. 이 옵션은 기본적으로 선택되어 있지 않습니다.
- 5 **생성**을 클릭하여 브랜치를 생성합니다. 로컬 레파지토리에 브랜치 생성을 위해 선택한 커밋의 파일이 포함됩니다.

브랜치 생성 ✕

커밋되지 않은 모든 변경 사항은 새로운 브랜치로 이동됩니다.

브랜치 생성 위치:

Auto Model Categories Count report.

baa9ef9a7981d13dc6d9127d7c44a67ffa94:

브랜치 이름:

modelcategories

생성 후 체크아웃

중복 브랜치 이름이 존재하면 강제 적용

생성

취소

브랜치 체크아웃

작업 디렉터리에서 브랜치를 열려면 브랜치를 체크아웃해야 합니다.

Git 탭 상단의 **현재 브랜치** 드롭다운 리스트를 사용하거나, 커밋 히스토리에서 로컬 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **브랜치 체크아웃** ⇨ **브랜치 이름**을 선택하여 로컬 브랜치를 체크아웃할 수 있습니다.

해당 로컬 브랜치가 없는 원격 브랜치로 작업하려면 다음 방법 중 하나를 이용하여 작업 디렉터리에 해당 브랜치의 로컬 사본을 생성할 수 있습니다.

- 원격 브랜치와 동일한 이름의 새로운 로컬 브랜치를 생성합니다. 자세한 내용은 [“브랜치 생성” \(148페이지\)](#)에서 참조하십시오.
- 커밋 히스토리에서 원격 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **브랜치 체크아웃** ⇨ **브랜치 이름**을 선택합니다. 브랜치의 로컬 버전이 생성되고 체크아웃됩니다.

.....

주: 이미 해당 로컬 브랜치가 있는 원격 브랜치를 체크아웃하면 원격 브랜치가 로컬 브랜치에 병합됩니다.

.....

병합 및 리베이스 이해

병합과 리베이스는 모두 한 브랜치의 변경 사항을 다른 브랜치에 통합하는 데 사용됩니다.

한 브랜치를 다른 브랜치로 병합하면 Git에서는 각 브랜치의 커밋 시퀀스 변경 사항을 결합합니다. Git에서 두 브랜치 모두에서 변경된 파일 내용을 발견하면 병합 충돌이 발생합니다. 모든 충돌을 수동으로 해결해야 병합을 완료할 수 있습니다. 병합이 완료된 후에도 두 브랜치의 히스토리를 계속 볼 수 있습니다. 기존 브랜치는 어떤 식으로도 변경되지 않습니다.

브랜치를 리베이스하면 전체 브랜치를 다른 브랜치의 맨 위에서 시작되도록 이동합니다. 리베이스를 사용하면 한 브랜치에 있는 모든 커밋을 다른 브랜치에 통합할 수 있습니다. 모든 충돌을 수동으로 해결해야 리베이스를 완료할 수 있습니다. 병합과 달리 리베이스는 프로젝트 히스토리를 다시 작성하며 선형 프로젝트 히스토리를 생성합니다. 더 이상 원래의 브랜치 히스토리를 개별적으로 볼 수 없습니다.

팁 이 액션은 다른 공개 브랜치 사용자에게 문제를 초래할 수 있으므로 main와 같은 공개 브랜치를 새로운 브랜치로 리베이스하면 안 됩니다.

브랜치 병합

브랜치에서 작업을 완료하면 해당 브랜치의 작업을 다른 브랜치에 병합할 수 있습니다.

브랜치를 병합하려면 다음을 수행합니다.

- 1 로컬 레파지토리에 대해 커밋 히스토리가 열려 있는지 확인합니다. **현재 브랜치** 드롭다운 리스트에서, 새로운 브랜치를 병합할 브랜치를 선택합니다. 예를 들어, *newbranch*를 main 브랜치와 병합하려면 main 브랜치를 체크아웃해야 합니다.

- 병합할 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **현재 브랜치 이름(으)로 병합** ⇨ **새로운 브랜치 이름**을 선택합니다. 예를 들어 *newbranch*를 main 브랜치와 병합하려면 *newbranch*를 마우스 오른쪽 버튼으로 클릭하고 **main(으)로 병합** ⇨ **새로운 브랜치**를 선택합니다. 커밋 히스토리 리스트의 현재 브랜치는 새로운 브랜치의 최신 커밋으로 업데이트됩니다.

주: 브랜치를 체크아웃되지 않은 브랜치로 병합하려면 **브랜치로 병합** ⇨ **브랜치 이름**을 선택합니다.

주: SAS Studio에서는 브랜치를 병합할 때 파일 간 충돌이 발생하면 메시지를 표시합니다. 충돌이 발생하면 충돌하는 파일은 커밋 하위 탭의 스테이징 해제된 변경 사항 영역으로 이동됩니다. 충돌을 해결한 후 파일을 스테이징 및 커밋하여 병합을 완료할 수 있습니다.

현재 브랜치 리베이스

리베이스 기능을 사용하여 브랜치를 새로운 베이스 커밋으로 이동할 수 있습니다. 브랜치를 리베이스하면 브랜치가 다른 커밋에서 생성된 것으로 보입니다. 리베이스는 레파지토리의 히스토리를 변경할 수 있는 강력한 기능입니다.

브랜치를 리베이스하려면 다음을 수행합니다.

- 로컬 레파지토리에 대해 커밋 히스토리가 열려 있는지 확인합니다. **현재 브랜치** 드롭다운 리스트에서 리베이스하려는 브랜치를 선택합니다. 예를 들어 *newbranch*를 main 브랜치로 리베이스하려면 *newbranch* 브랜치를 체크아웃해야 합니다.
- 새로운 베이스 커밋으로 사용할 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **브랜치 이름 리베이스** ⇨ **선택한 커밋**을 선택합니다.

주: 브랜치를 리베이스할 때 파일 간 충돌이 있으면 SAS Studio에 메시지가 표시됩니다. 충돌이 발생하면 충돌하는 파일은 커밋 하위 탭의 스테이징 해제된 변경 사항 영역으로 이동됩니다. 충돌을 해결한 후 파일을 스테이징 및 커밋하고 리베이스를 계속합니다.

변경 사항 스테시

스테시 기능을 사용하면 아직 원격 레파지토리에 푸시할 준비가 되지 않은 모든 커밋되지 않은 변경 사항(스테이징 및 스테이징 해제 상태)을 저장할 수 있습니다. 하나 이상의 파일에 대한 변경 사항을 스테시하면 변경 사항이 파일의 작업 사본에서 제거되고 다시 적용할 준비가 될 때까지 저장됩니다.

변경 사항을 파일에 다시 적용할 때 스테시의 변경 사항을 삭제할지 아니면 저장할지 선택할 수 있습니다.

주: 변경 사항을 스테시하면 레파지토리에 있는 모든 업데이트된 파일과 커밋되지 않은 파일에 영향을 미칩니다.

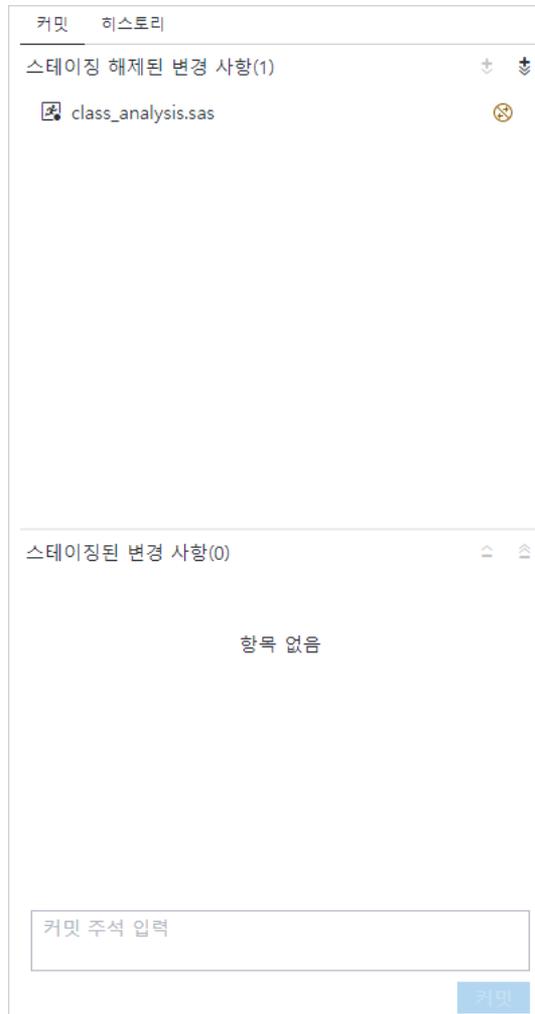
스테시 기능을 사용하려면 로컬 Git 레파지토리 탭이 작업 영역에서 열려 있는지 확인합니다. 도구 모음에서 을 클릭하고 다음 옵션 중에서 선택합니다.

- **스테시** — 커밋되지 않은 모든 변경 사항을 로컬 레파지토리의 파일에 저장하고 로컬 파일에서 변경 사항을 제거합니다.
- **스테시 팝** — 저장된 모든 변경 사항을 로컬 레파지토리의 파일에 다시 적용하고 스테시에서 변경 사항을 삭제합니다.
- **스테시 삭제** — 스테시에서 저장된 변경 사항을 삭제합니다. 스테시에서 변경 사항을 삭제하면 로컬 파일에 다시 적용할 수 없습니다.
- **스테시 적용** — 저장된 모든 변경 사항을 로컬 레파지토리의 파일에 다시 적용하고 스테시에서 변경 사항을 저장합니다.

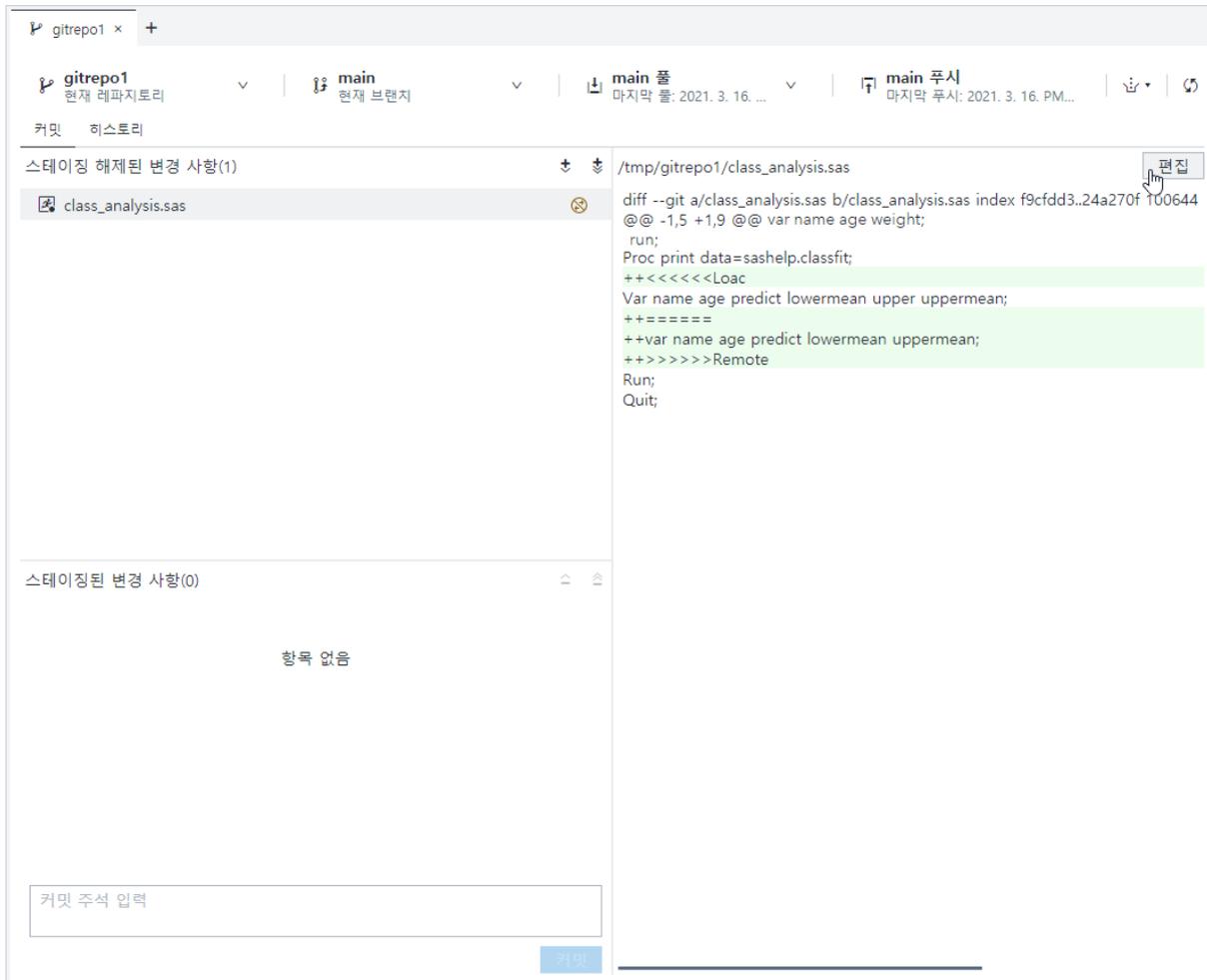
병합 충돌 해결

때때로 공유 레파지토리에서 작업할 때 사용자와 공동 작업자가 파일의 동일한 행을 변경했기 때문에 오류가 발생할 수 있습니다. SAS Studio는 로컬 브랜치에서 변경 사항을 병합하려고 할 때 파일에서 충돌을 식별합니다. SAS Studio에서 병합은 원격 레파지토리에서 파일을 풀하고 한 브랜치에서 다른 브랜치로 작업을 병합하고 Pop stash 및 Apply stash 옵션을 사용할 때 발생합니다.

예를 들어, 동료는 변경 사항을 파일에 푸시할 때 공동 작업자의 업데이트를 먼저 파일로 풀하지 않고 동일한 파일에 변경 사항을 푸시하려고 하면 오류가 발생할 수 있습니다. 파일에 충돌이 있을 때 원격 레파지토리에서 가져오거나 다른 병합 작업을 수행하려고 하면 SAS Studio는 오류 메시지를 표시하고 충돌을 해결하는 데 도움이 되는 정보를 제공합니다. SAS Studio는 파일의 로컬 버전과 원격 버전 간에 존재하는 충돌을 식별하기 위해 파일에 정보를 추가하여 충돌을 자동으로 문서화합니다. 충돌을 해결할 수 있도록 파일이 Commit 탭의 Unstaged Changes 영역으로 다시 이동됩니다.



파일의 충돌을 보려면 Unstaged Changes 영역에서 파일을 클릭합니다. Commit 탭의 오른쪽 영역에 파일이 표시됩니다. SAS Studio가 파일에 삽입한 충돌에 대한 문서를 봅니다. 로컬 섹션에는 로컬 커바의 변경 사항이 포함되고 원격 섹션에는 원격 레파지토리에서 가져온 커바의 충돌하는 행이 포함됩니다.



파일의 충돌을 해결하려면 **편집**을 클릭합니다. 그러면 해당 파일이 작업 영역의 새로운 탭에 열립니다. 필요에 따라 파일을 편집한 다음(두 변경 사항을 모두 고려) 파일을 저장합니다. Git 레파지토리 탭을 클릭한 다음 업데이트된 파일을 스테이지하고 커밋합니다.

레파지토리 삭제

서버 파일 시스템에서 로컬 레파지토리를 삭제할 수 있습니다. 레파지토리를 삭제할 때 현재 작업 중인 디렉터리의 모든 파일을 삭제할지를 선택할 수 있습니다.

로컬 레파지토리를 삭제하려면 다음을 수행합니다.

- 1 **옵션** ⇨ **Git 연결 관리**를 선택합니다. **레파지토리**를 클릭한 후 삭제할 레파지토리를 선택합니다. **삭제**를 클릭합니다.
- 2 수행할 삭제 유형을 선택합니다.
 - **Soft** - Git 레파지토리 정의만 삭제합니다.

주: 소프트 삭제 후에는 탐색 영역의 **탐색기** 섹션을 사용하여 폴더를 계속 볼 수 있습니다. 폴더를 다시 로컬 Git 레파지토리로 사용하려면 폴더를 마우스 오른쪽 버튼으로 클릭하고 **Git 초기화**를 선택해야 합니다.

- **Mixed** - Git 레파지토리 정의를 삭제하고 로컬 레파지토리 디렉터리에서 모든 Git 구성 정보를 제거합니다. 현재 작업 중인 디렉터리의 파일은 삭제되지 않습니다.

주: 복합 삭제 후에는 탐색 영역의 **탐색기** 섹션을 사용하여 폴더를 계속 볼 수 있습니다. 폴더를 다시 로컬 Git 레파지토리로 사용하려면 폴더를 마우스 오른쪽 버튼으로 클릭하고 **Git 초기화**를 선택해야 합니다.

- **Hard** - Git 레파지토리 정의 및 로컬 레파지토리 디렉터리와 현재 작업 중인 로컬 디렉터리의 모든 파일을 삭제합니다.

3 삭제

3 **삭제**를 클릭하여 레파지토리를 삭제합니다.

주: 탐색 영역의 **탐색기** 섹션에서 레파지토리를 삭제할 수도 있습니다. 현재 작업 중인 디렉터리를 마우스 오른쪽 버튼으로 클릭하고 **삭제**를 선택합니다. 확인 창에서 **삭제**를 클릭한 다음 수행할 삭제 유형을 선택합니다.

샘플 Git 워크플로우 시나리오

Git 레파지토리에 SAS 프로그램을 저장하는 프로젝트를 도와줄 일이 생겼다고 가정해 봅시다. 한 프로그램을 약간 수정하고 프로젝트에 새로운 프로그램을 추가해야 합니다.

이 샘플 Git 시나리오에 레파지토리를 서버 파일 시스템으로 복제하고 일부 변경 사항을 커밋 및 푸시(push)하고 브랜치를 생성합니다. 그런 다음, 새로운 브랜치를 main 브랜치와 병합하고 변경 사항을 푸시(push)합니다.

- 1 SAS Studio에서 Git 레파지토리에 액세스할 수 있도록 프로파일을 생성합니다.
 - a **옵션** ⇒ **Git 연결 관리**를 선택합니다. **프로파일**이 선택되어 있는지 확인하고 **+**을 클릭하여 **프로파일 추가** 창을 엽니다.
 - b Git 레파지토리 액세스에 필요한 자격 증명을 입력하고 **확인**을 클릭합니다.

프로파일 추가

프로파일 이름: *

사용자 이름: *

이메일: *

공용 SSH 파일 경로:

개인 SSH 파일 경로:

- 2 프로젝트의 파일에 액세스할 수 있도록 팀의 레파지토리를 서버 파일 시스템으로 복제합니다.
 - a 탐색 영역의 **Git 레파지토리**에서 **레파지토리 복제**를 클릭합니다.
 - b **레파지토리 복제** 창에서 사용할 서버 파일 시스템에서의 위치 및 원격 레파지토리를 지정합니다.

레파지토리 복제

레파지토리:

https://github.com/example/repo.git
git@bitbucket.org/exmple/repo.git

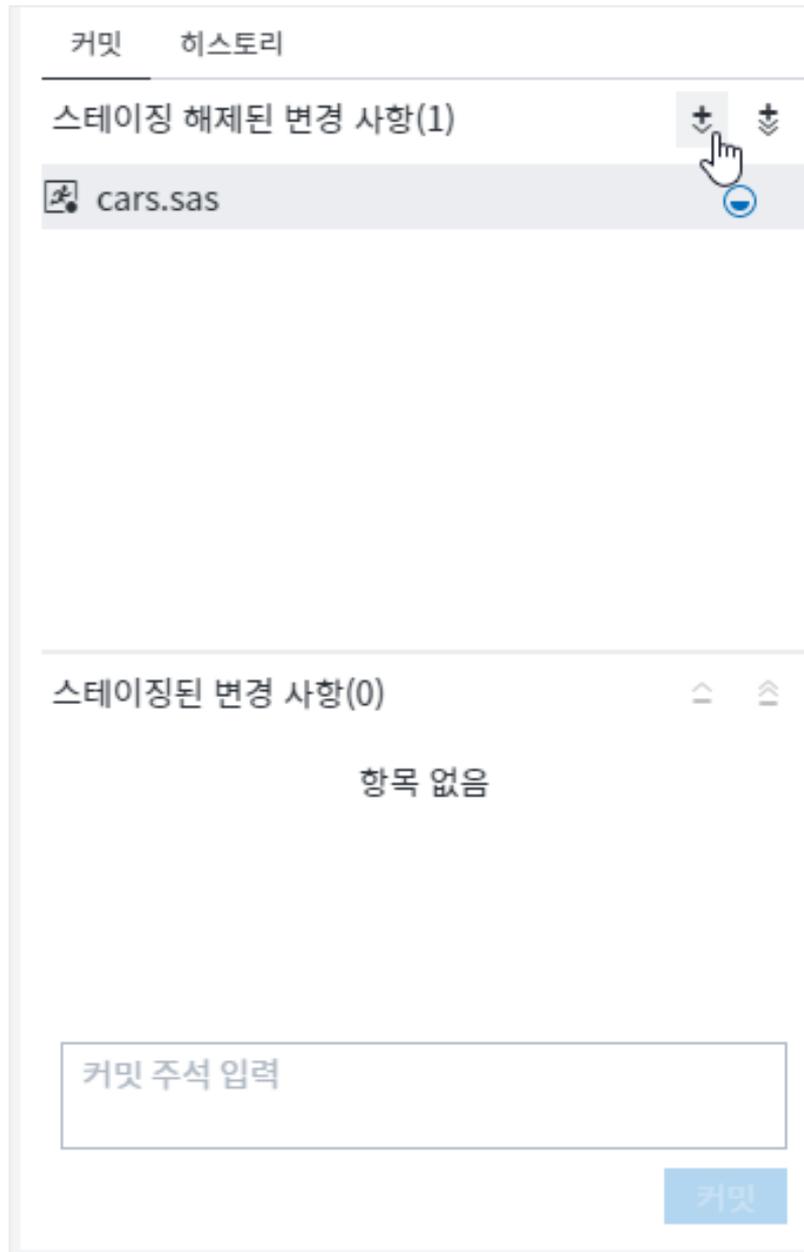
서버 위치:

프로파일:

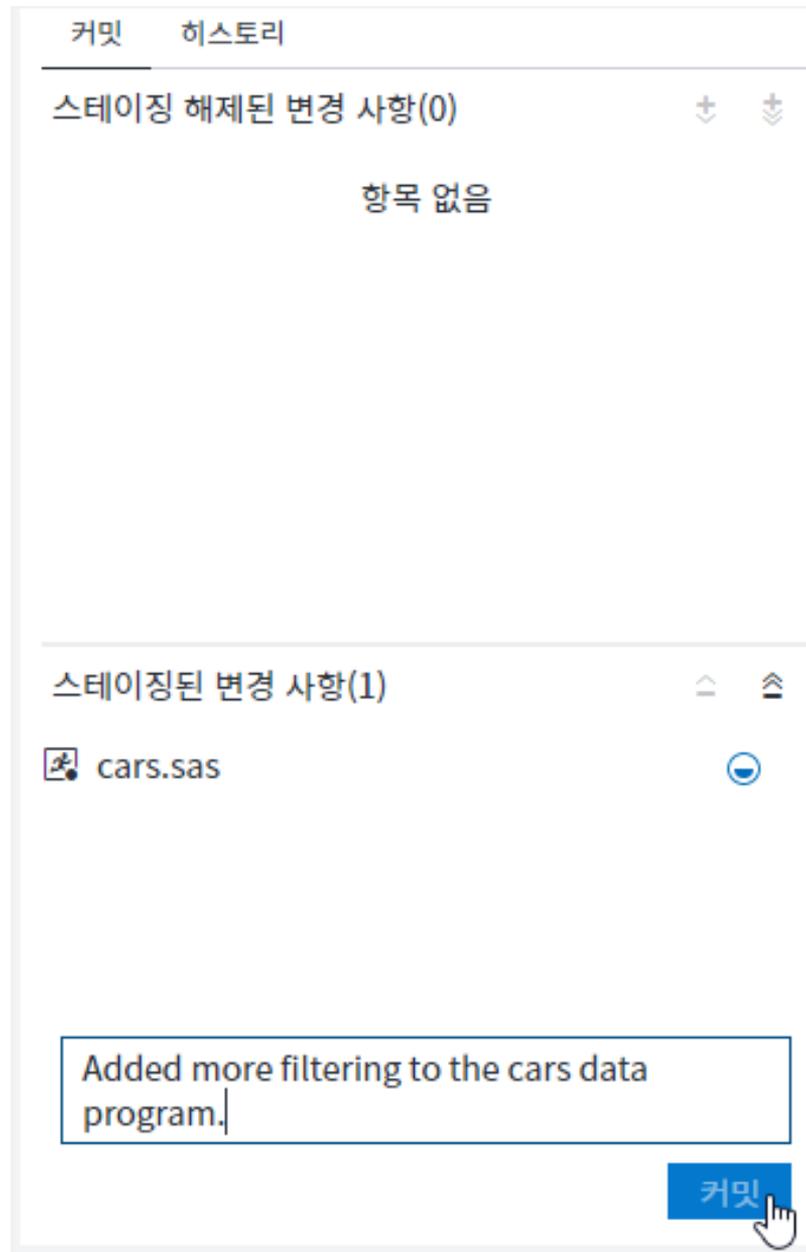
- c **복제**를 클릭하여 레파지토리를 생성합니다. 탐색 영역의 **Git 레파지토리** 섹션에 레파지토리가 추가됩니다.



- 3 로컬 레파지토리에서 업데이트가 필요한 프로그램을 열고 변경한 후 프로그램을 저장합니다.
 - a 탐색 영역의 을 클릭하여 **탐색기**를 열고 작업 디렉터리로 이동합니다.
 - b 편집 및 변경할 SAS 프로그램을 더블 클릭합니다. 을 클릭하여 변경 내용을 저장합니다.
- 4 변경한 파일을 스테이징하고, 커밋 메시지를 입력하고 로컬 레파지토리로 파일을 커밋합니다.
 - a 탐색 영역에서 을 클릭한 다음 레파지토리를 더블 클릭하여 작업 영역에서 엽니다. **커밋** 탭이 선택되어 있는지 확인합니다. 변경된 파일이 **스테이징 해제된 변경 사항** 영역에 나열되지 않으면 을 클릭하여 리스트를 새로 고침합니다.
 - b 업데이트된 파일을 선택하고 을 클릭하여 파일을 **스테이징된 변경 사항** 영역으로 이동합니다.



- c 커밋 상자에 커밋 주석을 입력하고 스타이징된 항목 커밋을 클릭합니다.



변경된 파일이 로컬 레파지토리로 커밋됩니다.

- d Git 레파지토리 탭에서 **히스토리**를 클릭하여 커밋 히스토리를 봅니다. 커밋은 커밋 히스토리 리스트 상단에 나열됩니다. 커밋은 현재 로컬 레파지토리의 main 브랜치에만 존재합니다. 로컬 레파지토리의 main 브랜치는 **✓ main**으로 식별됩니다.

리스트의 두 번째 커밋은 원격 레파지토리의 main 브랜치의 상태를 나타냅니다. 원격 레파지토리의 main 브랜치는 **origin/main**으로 식별됩니다.

변경 사항을 원격 레파지토리로 푸시(push)하면 커밋은 원격 레파지토리의 origin/main 브랜치의 일부가 됩니다.

The screenshot shows the SAS Studio Git interface. At the top, there are tabs for 'myfolder', 'main', 'main 풀', and 'main 푸시'. Below this is a commit history table with columns for '메시지', '작성자', '날짜', and '커밋 ID'. The current commit is 'Updated filter criteria' by 'Myprofile' with ID '048c629'. Below the table, the 'diff' view for 'car_models.sas' is shown, highlighting changes in a SQL query.

메시지	작성자	날짜	커밋 ID
✓ main Updated filter criteria	Myprofile	2020. 10. 10. PM 2:...	048c629
origin/main Added sorting	Myprofile	2020. 10. 10. PM 2:...	48b0152
Bug fixes and added more filtering	Myprofile	2020. 10. 10. PM 1:...	7a5bae0
Code revisions	Myprofile	2020. 10. 10. PM 1:...	1d481f3
New car model ranking report	Myprofile	2020. 10. 10. PM 1:...	8d6a4f8
Performance enhancements	Myprofile	2020. 10. 10. PM 1:...	949ba42

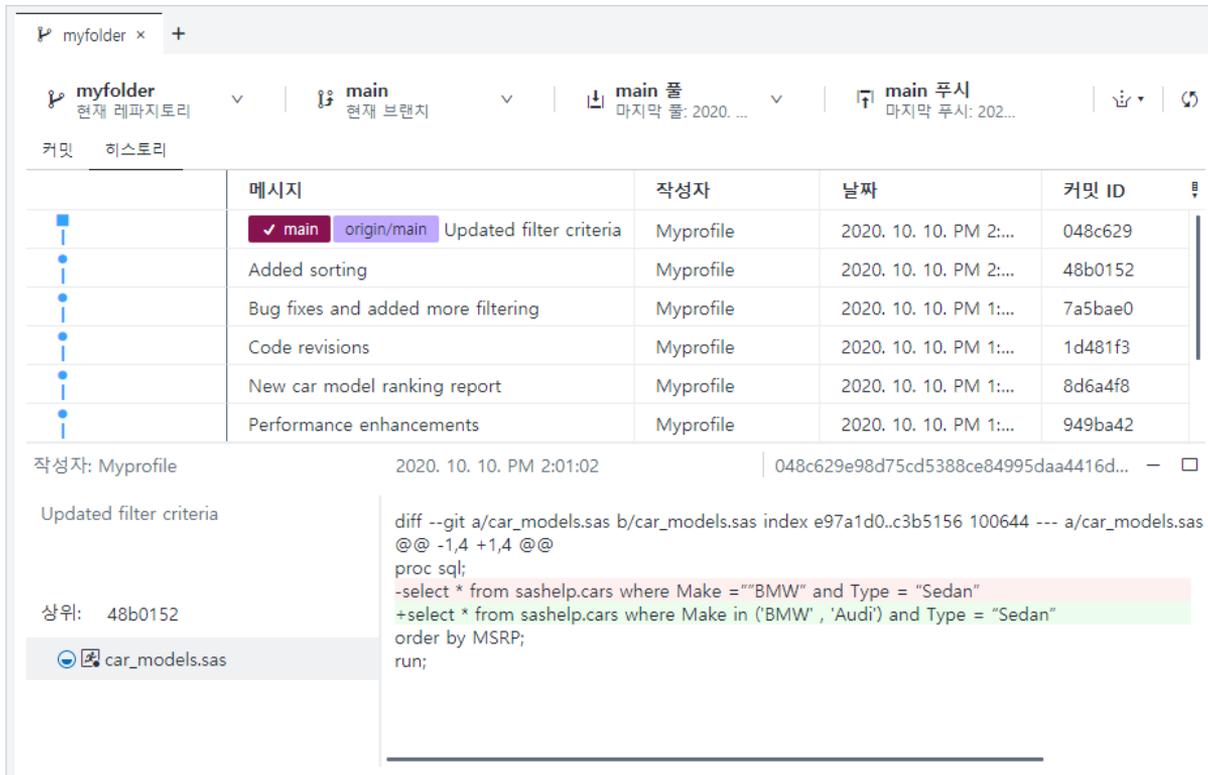
```

diff --git a/car_models.sas b/car_models.sas index e97a1d0..c3b5156 100644 --- a/car_models.sas
@@ -1,4 +1,4 @@
proc sql;
- select * from sashelp.cars where Make = "BMW" and Type = "Sedan"
+ select * from sashelp.cars where Make in ('BMW', 'Audi') and Type = "Sedan"
order by MSRP;
run;
  
```

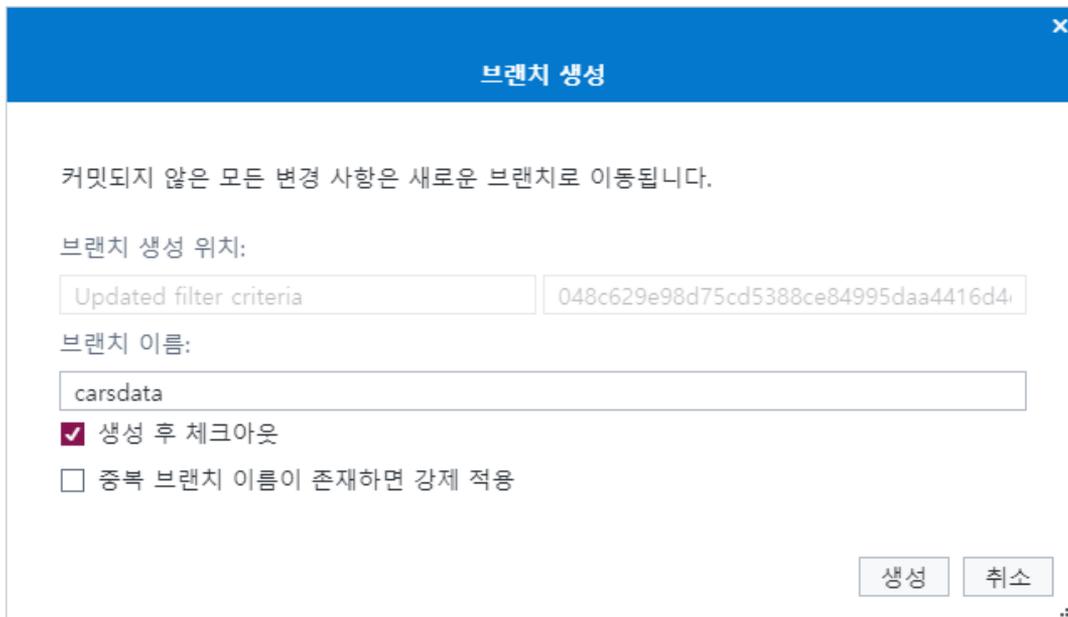
5 원격 레파지토리에서 파일을 풀(pull)하고 커밋한 파일을 푸시(push)합니다.

팁 충돌이 일어나지 않도록 푸시하기 전에 원격 레파지토리에서 파일을 풀(pull)하는 것이 좋습니다.

- Git 레파지토리 탭에서 **브랜치 이름 풀** ⇒ **풀**을 클릭합니다. 로컬 레파지토리가 원격 레파지토리의 최신 콘텐츠로 업데이트됩니다. 원격 레파지토리의 모든 변경 사항이 로컬 파일과 병합됩니다. 원격 및 로컬 레파지토리에서 파일 간 충돌이 있으면 진행하기 전에 충돌을 해결할 수 있도록 SAS Studio에서 메시지를 표시합니다.
- 브랜치 이름 푸시**를 클릭하여 변경된 파일을 원격 레파지토리로 푸시합니다. 커밋 히스토리가 업데이트됩니다. 커밋이 원격 origin/main 브랜치의 일부가 됩니다.



- 6 작업을 main 브랜치와 분리하기 위한 브랜치를 생성합니다. 새로운 프로그램을 추가하고 시험한 후 변경 사항을 해당 브랜치에 커밋합니다.
 - a 커밋 히스토리 리스트에서 커밋을 마우스 오른쪽 버튼으로 클릭하고 **새로운 브랜치 생성**을 선택합니다. **브랜치 생성** 창이 나타납니다.
 - b 브랜치의 이름을 입력하고 **생성**을 클릭합니다. 새로운 브랜치가 체크아웃되고 작업 중인 활성 브랜치가 될 수 있도록 **생성 후 체크아웃** 옵션이 선택됩니다.



커밋 히스토리 리스트에서 새로운 carsdata 브랜치, 원격 origin/main 브랜치 및 로컬 레파지토리 main 브랜치가 모두 동기화되어 같은 행에 표시된 것을 볼 수 있습니다.

The screenshot shows the SAS Studio interface with the 'carsdata' branch selected. The commit history table is as follows:

커밋	메시지	작성자	날짜	커밋 ID
048c629	Updated filter criteria	Myprofile	2020. 10. 10. PM 2:01:02	048c629
48b0152	Added sorting	Myprofile	2020. 10. 10. PM 2:01:02	48b0152
7a5bae0	Bug fixes and added more filtering	Myprofile	2020. 10. 10. PM 1:01:02	7a5bae0
1d481f3	Code revisions	Myprofile	2020. 10. 10. PM 1:01:02	1d481f3
8d6a4f8	New car model ranking report	Myprofile	2020. 10. 10. PM 1:01:02	8d6a4f8
949ba42	Performance enhancements	Myprofile	2020. 10. 10. PM 1:01:02	949ba42

The diff view for commit 48b0152 shows the following changes to 'car_models.sas':

```
diff --git a/car_models.sas b/car_models.sas index e97a1d0..c3b5156 100644 --- a/car_models.sas
@@ -1,4 +1,4 @@
proc sql;
-select * from sashelp.cars where Make = "BMW" and Type = "Sedan"
+select * from sashelp.cars where Make in ('BMW', 'Audi') and Type = "Sedan"
order by MSRP;
run;
```

- c. 탐색 영역의 **탐색기** 섹션을 사용하여 프로그램을 생성하고 작업 디렉터리에 저장합니다.
- d. 새로운 프로그램을 스테이징하고 로컬 레파지토리의 새로운 carsdata 브랜치에 커밋합니다. 커밋 히스토리 리스트에서 carsdata 브랜치의 커밋이 원격 origin/main 브랜치 또는 로컬 main 브랜치와 동기화되지 않은 것을 볼 수 있습니다.

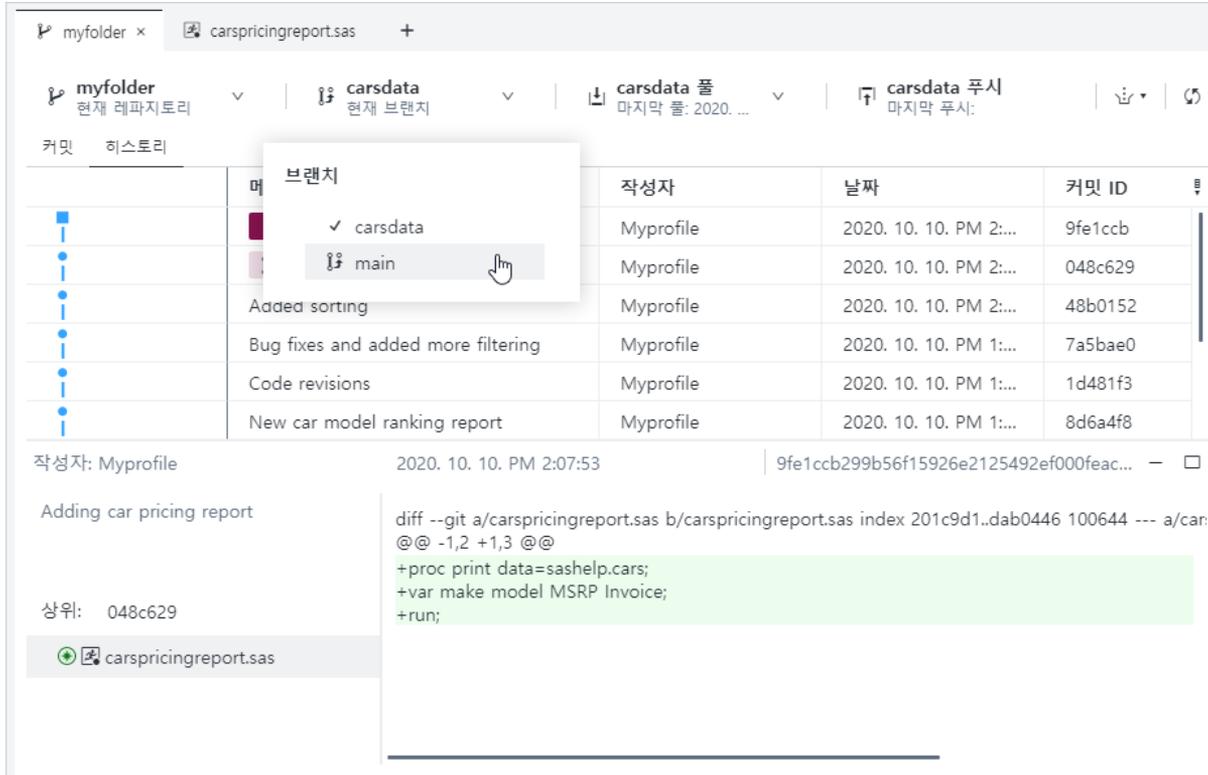
The screenshot shows the SAS Studio interface with the 'carsdata' branch selected. The commit history table is as follows:

커밋	메시지	작성자	날짜	커밋 ID
9fe1ccb	Adding car pricing report	Myprofile	2020. 10. 10. PM 2:07:53	9fe1ccb
048c629	Updated filter criteria	Myprofile	2020. 10. 10. PM 2:01:02	048c629
48b0152	Added sorting	Myprofile	2020. 10. 10. PM 2:01:02	48b0152
7a5bae0	Bug fixes and added more filtering	Myprofile	2020. 10. 10. PM 1:01:02	7a5bae0
1d481f3	Code revisions	Myprofile	2020. 10. 10. PM 1:01:02	1d481f3
8d6a4f8	New car model ranking report	Myprofile	2020. 10. 10. PM 1:01:02	8d6a4f8

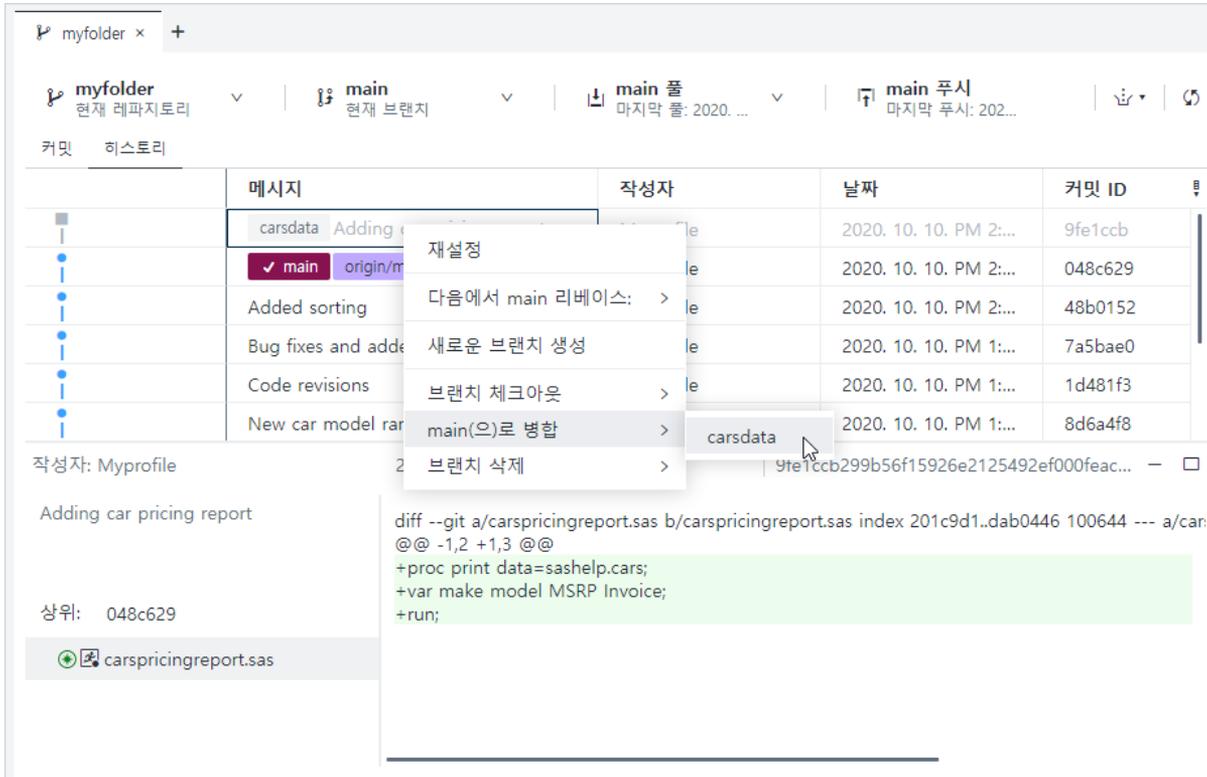
The diff view for commit 9fe1ccb shows the following changes to 'carspricingreport.sas':

```
diff --git a/carspricingreport.sas b/carspricingreport.sas index 201c9d1..dab0446 100644 --- a/carspricingreport.sas
@@ -1,2 +1,3 @@
+proc print data=sashelp.cars;
+var make model MSRP Invoice;
+run;
```

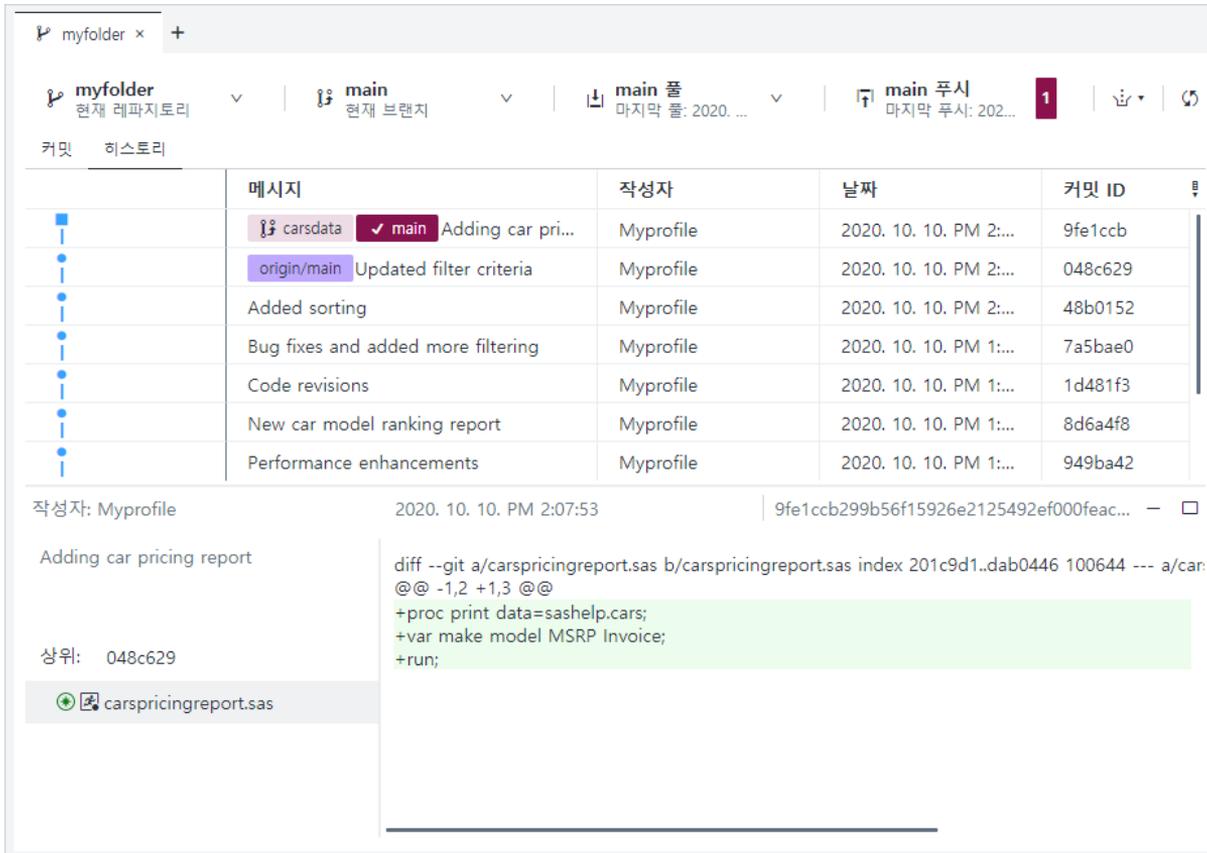
- 7 새로운 프로그램이 완료되면 새로운 브랜치를 다시 main 브랜치와 병합하고 원격 레파지토리 최종 풀(pull) 및 푸시(push)를 수행합니다.
 - a carsdata 브랜치를 main 브랜치로 병합할 준비가 되면 먼저 main 브랜치를 체크아웃해야 합니다. 현재 브랜치 드롭다운 리스트에서 main 브랜치를 선택합니다. 확인 창에서 **다시 로드**를 클릭하여 모든 변경 사항이 저장되었는지 확인합니다.



커밋 히스토리 리스트에서 carsdata 브랜치를 마우스 오른쪽 버튼으로 클릭하고 **main으로 병합** ⇒ **carsdata**를 선택합니다.



커밋 히스토리 리스트에서 main 브랜치가 carsdata 브랜치의 새로운 파일로 업데이트된 것을 볼 수 있습니다.



b Git 탭에서 main 풀 ⇨ 풀을 선택합니다.

- c 푸시를 클릭하여 변경 사항을 원격 레파지토리로 푸시합니다. 커밋 히스토리가 업데이트됩니다. carsdata 브랜치와 병합된 로컬 main 브랜치 및 원격 origin/main 브랜치가 동기화되었음을 알 수 있습니다.

The screenshot shows a Git client interface for a repository named 'myfolder'. The current branch is 'main', and the commit being viewed is 'main 푸시' (main push) with ID '9fe1ccb'. The commit history table is as follows:

커밋	메시지	작성자	날짜	커밋 ID
9fe1ccb	carsdata main origin/main Ad...	Myprofile	2020. 10. 10. PM 2:...	9fe1ccb
048c629	Updated filter criteria	Myprofile	2020. 10. 10. PM 2:...	048c629
48b0152	Added sorting	Myprofile	2020. 10. 10. PM 2:...	48b0152
7a5bae0	Bug fixes and added more filtering	Myprofile	2020. 10. 10. PM 1:...	7a5bae0
1d481f3	Code revisions	Myprofile	2020. 10. 10. PM 1:...	1d481f3
8d6a4f8	New car model ranking report	Myprofile	2020. 10. 10. PM 1:...	8d6a4f8
949ba42	Performance enhancements	Myprofile	2020. 10. 10. PM 1:...	949ba42

The diff view for the selected commit shows the following changes:

```

diff --git a/carspricingreport.sas b/carspricingreport.sas index 201c9d1..dab0446 100644 --- a/car:
@@ -1,2 +1,3 @@
+proc print data=sashelp.cars;
+var make model MSRP Invoice;
+run;
    
```

The file 'carspricingreport.sas' is listed in the file explorer on the left.

SAS Studio의 작업 이해

작업이란?	167
작업을 실행하는 방법	167
사전 정의된 작업 편집	168
사용자 정의 작업 생성	169
작업 코드에 대한 옵션 지정	170

작업이란?

작업은 자동으로 SAS 코드를 생성하고 결과의 출력형식을 지정하는 XML 및 Apache Velocity 코드 파일입니다. 작업에는 단순한 데이터 나열에서부터 복잡한 분석 프로시저에 이르는 다양한 SAS 프로시저가 포함됩니다. SAS Studio에서는 표준 영역에서 사용할 수 있는 사전 정의된 몇 가지 작업이 제공됩니다. 작업은 범주로 구성됩니다. 필요한 SAS 제품이 없기 때문에 일부 범주 및 작업은 사이트에서 실행되지 않을 수 있습니다.

이러한 각 작업에 대한 자세한 내용은 *SAS Studio: Task Reference Guide*에서 확인하십시오.

작업을 실행하는 방법

사전 정의된 작업을 실행하려면 다음을 수행합니다.

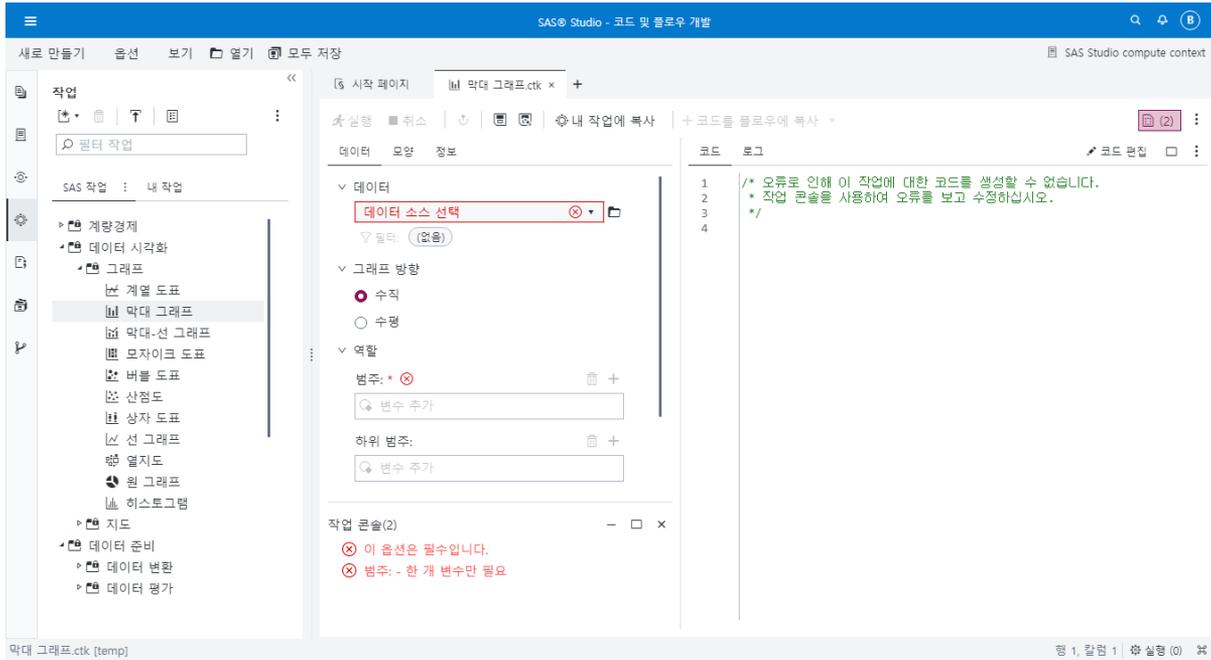
- 1 탐색 영역에서 을 클릭하여 **작업** 영역을 엽니다.
- 2 작업이 포함된 폴더를 펼칩니다.

주: 탐색 영역에서 선택한 작업 또는 작업 폴더를 마우스 오른쪽 버튼으로 클릭하고 **숨기기**를 선택하여 하나 이상의 작업 및 작업 폴더를 숨길 수 있습니다. 숨겨진 작업 및 작

업 폴더는 탐색 영역에서 업데이트된 아이콘과 함께 표시되며 실행할 수 없습니다. 숨겨진 작업 및 작업 폴더를 표시하려면 작업 또는 작업 폴더를 마우스 오른쪽 버튼으로 클릭하고 **표시**를 선택합니다.

3. 작업 이름을 마우스 오른쪽 버튼으로 클릭하고 **작업 열기**를 선택합니다. 또는 작업을 더블 클릭하여 열 수도 있습니다.

작업 영역 오른쪽에 작업이 열립니다.



4. **데이터** 탭을 사용할 수 있으면 입력 데이터 소스를 지정하고 데이터 소스의 역할에 대한 칼럼을 선택합니다. 역할은 작업 내의 변수 용도에 대한 설명입니다. 역할에 칼럼을 추가하려면 **+**을 클릭합니다. 그러면 해당 역할에 사용 가능한 칼럼의 리스트가 나타납니다. 여러 칼럼을 할당할 수 있으면 Ctrl 또는 Shift 키를 누른 상태로 리스트에서 여러 칼럼을 선택하고 **확인**을 클릭합니다.
5. 나머지 탭에서 빨간색 별표로 표시된 다른 필수 옵션을 지정합니다. 작업에 값을 할당하면 관련 SAS 코드가 생성됩니다. 각 작업에 대해 사용 가능한 옵션에 대한 자세한 내용은 [SAS Studio: Task Reference Guide](#)에서 확인하십시오.

팁 작업 콘솔에는 작업을 실행하는 데 필요한 모든 정보가 표시됩니다.

6. 작업을 실행하려면 **실행**을 클릭합니다.

사전 정의된 작업 편집

사전 정의된 작업을 사용자의 사이트에 맞게 사용자 정의하려면 작업을 생성하는 데 사용된 XML 코드를 편집합니다.

사전 정의된 작업을 편집하려면 다음을 수행합니다.

- 1 탐색 영역에서 을 클릭하여 **작업** 영역을 엽니다.
- 2 작업이 포함된 폴더를 펼칩니다.
- 3 편집할 작업의 이름을 마우스 오른쪽 버튼으로 클릭하고 **작업 템플릿으로 복사**를 선택합니다. 내 작업 또는 탐색기에서 작업 저장을 선택할 수 있습니다.
- 4 작업을 위한 이름 및 설명을 지정합니다. 기본적으로 사전 정의된 작업의 이름과 설명을 사용합니다.

확인을 클릭합니다.

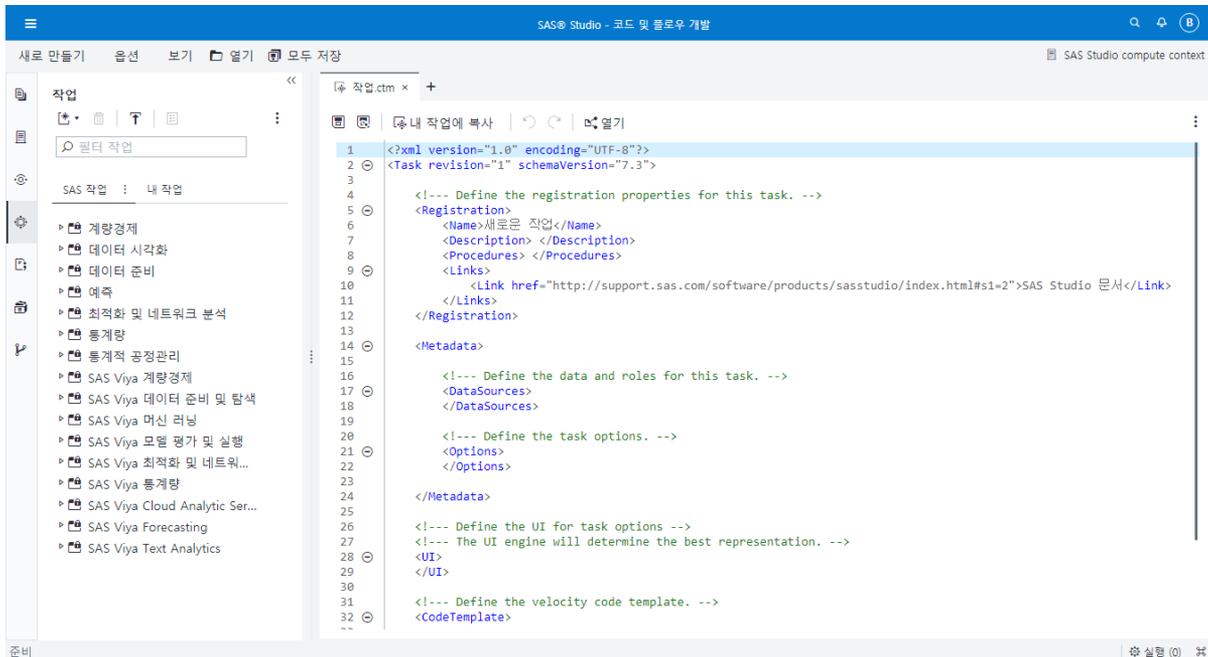
- 5 복사된 작업을 마우스 오른쪽 버튼으로 클릭하고 **작업 템플릿 편집**를 선택합니다. 작업의 CTM 파일이 새 탭에 나타납니다.
- 6 XML 파일을 편집하고 변경 내용을 저장합니다. 변경 내용을 미리 보려면 **열기**를 클릭합니다.

사용자 정의 작업 생성

SAS Studio에서는 사용자의 사이트에 맞는 사용자 정의 작업을 생성하는 데 사용할 수 있는 템플릿이 제공됩니다.

사용자 정의 작업을 생성하려면 다음을 수행합니다.

- 1 탐색 영역에서 을 클릭하여 **작업** 영역을 엽니다.
- 2 를 클릭하고 **작업**을 선택합니다. 빈 작업 템플릿이 열립니다.



- 3 작업 템플릿의 코드를 편집하여 작업을 생성합니다. 작업 템플릿의 사용자 인터페이스를 보려면 **열기**를 클릭합니다. 이 파일에 대한 자세한 내용은 *SAS Studio: 사용자 정의 작업 작성을 위한 개발자 가이드*에서 확인하십시오.
- 4 을 클릭합니다.

작업 코드에 대한 옵션 지정

우선 설정 창에서 작업 코드가 표시되는 방식에 영향을 주는 여러 옵션을 변경할 수 있습니다.

이러한 옵션에 액세스하려면 **옵션** ⇨ **우선 설정**을 선택합니다. **우선 설정** 창에서 **작업**을 클릭합니다.

자세한 내용은 “**작업 우선 설정 지정**” (183페이지)에서 확인하십시오.

부록 1

SAS Studio 사용자 정의

SAS Studio 사용자 정의 정보	171
글로벌 설정 생성	172
설정 창 정보	172
일반 설정 지정	172
지역 및 언어 설정 지정	173
접근성 설정 지정	173
우선 설정 지정	174
우선 설정 창 정보	174
시작 기본 설정 지정	174
일반 우선 설정 지정	175
코드 및 로그 우선 설정 지정	176
결과 우선 설정 지정	178
편집기 우선 설정 지정	179
일반 코드 편집기 우선 설정 지정	180
코드 편집기에서 모양 옵션 설정	181
테이블 우선 설정 지정	182
작업 우선 설정 지정	183
플로우 우선 설정 지정	184
백그라운드 실행 우선 설정 지정	184
질의 우선 설정 지정	184
가져오기 우선 설정 지정	185

SAS Studio 사용자 정의 정보

테마 및 로케일과 같은 글로벌 설정을 변경하고, 코드 편집기 및 테이블 뷰어 옵션과 같은 응용 프로그램 우선 설정을 지정하여 SAS Studio를 사용자 정의할 수 있습니다.

글로벌 설정 생성

설정 창 정보

설정 창에서는 모양 테마, 로케일 및 접근성에 대한 설정을 지정할 수 있습니다. 글로벌 설정을 변경하려면 응용 프로그램 표시줄에서 사용자 버튼을 클릭하고 **설정**을 선택합니다.

일반 설정 지정

일반 페이지에서 SAS Studio에 대한 옵션을 지정할 수 있습니다.

옵션	설명
테마	<p>SAS Studio 웹 응용 프로그램의 모양을 변경할 수 있습니다. 테마는 응용 프로그램에 표시되는 색상, 그래픽 및 글꼴의 컬렉션을 지정합니다. SAS 테마나 사용자 정의 테마(사용할 수 있는 테마가 있을 때) 중에서 선택할 수 있습니다.</p> <ul style="list-style-type: none"> ■ 기본 테마(테마 이름) 사용 — 시스템 관리자가 지정한 기본 테마를 사용합니다. ■ 테마 선택 — 테마를 선택할 수 있습니다. 다음과 같은 SAS 테마 중에서 선택할 수 있습니다. <ul style="list-style-type: none"> High Contrast 시력이 약한 사용자의 요구를 충족하기 위해 어두운 배경과 고대비 전경 요소를 표시합니다. Dark 그래프, 시각화 및 기타 요소가 눈에 띄도록 어두운 사용자 인터페이스를 표시합니다. Light 깨끗하고 복잡하지 않으며 사용하기 쉬운 색상 팔레트를 포함합니다. 기본 테마입니다. <p>테마 변경은 설정 창을 닫은 후에 적용됩니다.</p>
모든 경고 및 정보 메시지를 표시하도록 재설정	이전에 숨기도록 선택한 경고 및 메시지를 표시합니다.
프로파일 사진	프로파일 사진을 표시합니다.

지역 및 언어 설정 지정

지역 및 언어 페이지에서 다음과 같은 옵션을 지정할 수 있습니다.

옵션	설명
지역 형식 및 정렬을 위한 로케일	데이터를 정렬하고 날짜, 시간, 숫자 및 통화 같은 값의 출력형식을 지정하는 데 사용되는 로케일을 지정합니다. 기본 설정은 브라우저 로케일입니다. 변경 사항은 로그아웃했다가 다시 로그인한 후에 적용됩니다.
오프라인 프로세스를 위한 로케일	리포트 배포 또는 알림과 같은 백그라운드 프로세스나 오프라인 작업에 사용되는 로케일을 지정합니다. 기본 설정은 Java 런타임 환경 로케일입니다.

접근성 설정 지정

접근성 페이지에서 다음과 같은 옵션을 지정할 수 있습니다.

옵션	설명
소리 사용	사용자 인터페이스 내에서 발생하는 이벤트에 대해 오디오 표시기를 사용하도록 설정합니다.
시각 효과 사용	상태 변경 사항을 나타내는 시각 효과를 추가합니다. 예를 들어 이 설정이 사용되면 항목을 삭제할 때 사용자 인터페이스에서 약간의 움직임을 볼 수 있습니다.
팝업 알림 표시 기간 조정	팝업 알림이 표시되는 기간(초)을 지정할 수 있습니다. 기본값은 5초입니다.
응용 프로그램 색상 반전	SAS Studio의 표시 색상을 반전합니다.
키보드를 사용하여 탐색할 때 툴팁 표시	키보드를 사용하여 탐색할 때 응용 프로그램의 각 항목에 대한 툴팁을 팝업 창으로 표시합니다. 툴팁을 표시할 브라우저 창 내의 위치를 지정할 수 있습니다.
포커스 표시기 설정 사용자 정의	활성 상태인 사용자 인터페이스 구성 요소를 나타내는 윤곽선의 속성을 사용자 정의할 수 있습니다.

우선 설정 지정

우선 설정 창 정보

우선 설정 창에서는 SAS Studio의 몇 가지 옵션을 사용자 정의할 수 있습니다.

우선 설정을 변경하려면 **옵션** ⇒ **우선 설정**을 선택합니다.

주: 우선 설정은 플로우의 노드에 적용되지 않습니다.

시작 기본 설정 지정

시작 페이지에서 SAS Studio 시작에 대한 옵션을 지정할 수 있습니다.

옵션	설명
작업 공간 구성 파일 로드	SAS Studio 세션이 시작될 때 구성을 지정하는 작업 공간 구성 파일 (.workspace)을 선택할 수 있습니다. 자세한 내용은 “ 작업 공간 구성 파일을 사용하여 시작을 사용자 정의 ” (6페이지)에서 참조하십시오.
탭 열기	SAS Studio 응용 프로그램을 시작할 때 표시할 탭을 지정합니다. 다음 옵션 중에서 선택할 수 있습니다. <ul style="list-style-type: none"> ■ 없음 - 작업 영역에 아무 탭도 열지 않은 상태로 SAS Studio를 시작합니다. 이 옵션은 기본적으로 선택되어 있습니다. ■ 파일 열기 영역에 나열된 탭 복원 - SAS Studio의 이전 세션에서 열었던 탭을 복원합니다. 이 옵션을 선택하면 SAS Studio 세션에서 표준 및 대화식 영역 간에 전환할 때 탭도 복원됩니다. 열었던 임시 파일은 복원되지 않습니다. <p>주: 탭이 항상 SAS Studio를 닫을 때의 나열 순서와 같은 순서로 복원되는 것은 아닙니다. 많은 데이터를 표시하는 탭은 여는 데 더 오랜 시간이 걸리므로 마지막에 열릴 수도 있습니다.</p> ■ 새로운 파일 열기 - 새로운 프로그램 또는 플로우 탭을 열지를 선택할 수 있습니다. 대화식 영역을 사용하고 있으면 이 옵션은 SAS 프로그램 탭을 엽니다.
시작 페이지 표시	SAS Studio 응용 프로그램을 시작할 때 시작 페이지 탭이 열립니다. 작업 영역에서 다른 탭을 열 때 시작 페이지 탭을 자동으로 닫으려면 자동 해제 를 선택합니다.
탐색기 영역에서 열려	SAS Studio의 이전 세션에서 탐색 영역의 탐색기 섹션에서 펼쳐진 폴더를 자동으로 다시 엽니다.

옵션	설명
있는 폴더 상태 복원	

일반 우선 설정 지정

일반 페이지에서 해당 옵션을 설정할 수 있습니다.

옵션	설명
실행 후 탭 표시	프로그램 또는 작업 실행 후에 표시할 탭을 지정합니다. 코드, 로그 또는 결과 탭을 표시하도록 선택할 수 있습니다. 표준 영역을 실행 중이면 출력 데이터 탭을 표시하도록 선택할 수도 있습니다. 결과 탭이 기본으로 표시됩니다.
출력 데이터 탭을 활성화할 때 자동으로 열기	데이터를 생성하는 프로그램 또는 작업을 실행할 때 표시할 출력 데이터 탭을 지정합니다.
영역에서 출력 데이터 보기	접기 가능한 영역에 프로그램, 작업 및 플로우에 대한 출력 데이터 리스트를 표시합니다. 이 옵션이 선택되지 않으면, 간단한 메뉴에 출력 데이터 리스트가 표시됩니다.
생성 순서순으로 출력 데이터셋 정렬	생성된 순서대로 출력 데이터셋을 정렬할 수 있습니다. 이 옵션은 기본적으로 선택되며 많은 수의 출력 데이터셋을 생성하는 경우 성능에 영향을 줄 수 있습니다. 이 옵션을 선택 취소하면 출력 데이터셋을 오름차순 또는 내림차순으로만 정렬할 수 있습니다. 기본적으로 출력 데이터셋은 오름차순으로 정렬됩니다. 자세한 내용은 “출력 데이터 뷰” (131페이지) 에서 확인하십시오.
텍스트 파일이 지정된 크기보다 크면 경고 표시	지정된 크기(MB)보다 큰 텍스트 파일을 열면 경고를 표시합니다. 텍스트 파일에는 *.sas, *.txt 및 *.csv와 같은 파일 유형이 포함됩니다. 지정된 크기보다 큰 파일을 열면 해당 파일이 잘린 후 읽기 전용 모드로 열립니다. 기본 크기는 10MB입니다.
기본 텍스트 인코딩	*.sas 프로그램과 같은 텍스트 파일을 읽거나 쓸 때 사용되는 문자 집합 인코딩을 지정합니다. 기본값은 UTF-8입니다. 일부 인코딩 옵션과 관련 언어 리스트는 부록 3, “텍스트 인코딩 옵션 및 언어 매핑” (189페이지) 에서 참조하십시오.
파일 시스템에서 숨겨진 폴더 및 항목 표시	숨겨진 파일 및 폴더를 탐색 영역의 탐색기 섹션에 표시합니다.
응용 프로그램 탐색 영역에 레이블 표시	탐색 영역의 아이콘과 함께 레이블을 표시합니다.
문서 복구 사용	부재 중 SAS Studio가 시간 초과되거나 정전과 같은 상황으로 인해 예기치 않게 종료되는 경우 SAS 프로그램, 플로우, 질의 및 작업 등 열려 있는 파일의 가장 최근에 저장된 버전을 복구할 수

옵션	설명
	<p>있습니다. 이 옵션을 선택하면 문서 복구 창을 사용하여 복구할 파일을 선택할 수 있습니다. 자세한 내용은 “문서 복구 창 사용” (27페이지)에서 확인하십시오.</p> <p>이 옵션은 기본적으로 선택되어 있습니다. 문서 저장 간격 옵션을 사용하여 파일을 저장할 빈도를 지정할 수 있습니다. 기본 구간은 30초입니다.</p>
각 실행 이후에 라이브러리를 자동으로 새로 고침	<p>프로그램, 작업 또는 질의를 실행한 후 라이브러리 변경 사항이 바로 표시되도록 탐색 영역의 라이브러리 섹션을 새로 고칩니다. 그러나 이 옵션을 선택하면 라이브러리 리스트의 변경 사항이 손실될 수 있으며 성능에도 영향을 미칩니다. 이 옵션은 기본적으로 선택되어 있습니다.</p> <p>주: 이 옵션을 선택하지 않고 새로운 파일이나 라이브러리를 생성하는 코드를 실행하면 ☰ > 새로 고침을 클릭해서 변경 사항을 표시해야 합니다. 개별 테이블 또는 라이브러리를 새로 고침하려면 테이블 또는 라이브러리를 마우스 오른쪽 버튼으로 클릭하고 새로 고침을 선택합니다. SAS Studio 사용자 인터페이스를 사용하여 새로운 파일 또는 라이브러리를 생성할 때는 이 옵션이 선택되어 있지 않더라도 변경 사항이 자동으로 표시됩니다.</p>
각 실행 이후에 파일 및 폴더를 자동으로 새로 고침	<p>프로그램, 작업 또는 질의를 실행한 후 파일 및 폴더 변경 사항이 바로 표시되도록 탐색 영역의 탐색기 섹션을 새로 고칩니다. 그러나 이 옵션을 선택하면 파일 및 폴더 리스트의 변경 사항이 손실될 수 있으며 성능에도 영향을 미칩니다. 이 옵션은 기본적으로 선택되어 있습니다.</p> <p>주: 이 옵션을 선택하지 않고 새로운 파일이나 폴더를 생성하는 코드를 실행하면 ☰ > 새로 고침을 클릭해서 변경 사항을 표시해야 합니다. 개별 폴더를 새로 고침하려면 폴더를 마우스 오른쪽 버튼으로 클릭하고 새로 고침을 선택합니다. SAS Studio 사용자 인터페이스를 사용하여 새로운 파일 또는 폴더를 생성할 때는 이 옵션이 선택되어 있지 않더라도 변경 사항이 자동으로 표시됩니다.</p>
제한된 명령 표시	명령행에 입력할 때 가능한 명령 옵션을 표시합니다.

코드 및 로그 우선 설정 지정

SAS 프로그램 ⇨ **코드 및 로그** 페이지에서 코드 및 로그 탭에 대한 옵션을 지정할 수 있습니다.

옵션	설명
SAS 로그에 사용자 정의 코드 표시	로그 파일에 사용자 정의 코드를 포함합니다.

옵션	설명
고급 로그 뷰어 사용	<p>프로그램이 실행될 때마다 로그를 자동으로 생성합니다. 이 옵션은 기본적으로 선택되어 있습니다. 이 옵션을 선택하지 않으면 로그 탭이 활성화될 때에만 로그가 로드됩니다. 고급 로그 뷰어를 사용하면 성능에 영향을 줄 수 있습니다. 고급 로그 뷰어를 사용할 때 다음 옵션 중에서 선택할 수 있습니다.</p> <ul style="list-style-type: none"> ■ 오류, 경고 및 노트 요약 생성은 로그의 모든 오류, 경고 및 노트에 쉽게 액세스할 수 있는 섹션을 로그 맨 위에 생성합니다. 메시지를 클릭하면 메시지가 로그에서 발생하는 정확한 위치를 볼 수 있도록 SAS Studio가 해당 메시지를 로그에서 강조 표시합니다. ■ 코드 탭 여백에 오류 및 경고 아이콘 표시 - 코드 탭의 여백에 오류 및 경고 아이콘을 표시합니다. 아이콘을 클릭할 때 SAS Studio가 로그에서 메시지를 강조 표시합니다. 전체 프로그램을 실행하고 로그 자동 지우기가 선택될 때에만 여백에 아이콘이 표시됩니다. 로그 추가가 선택되면 아이콘을 표시할 수 없습니다. ■ SAS 로그에 오류 및 경고만 표시 - 로그에 오류 및 경고 메시지만 표시합니다. 정보 노트는 표시되지 않습니다.
프로시저가 실행되는 동안 로그 업데이트 스트리밍	<p>프로시저가 처리되는 동안 로그 업데이트를 표시합니다. 이 옵션을 선택하지 않으면 프로시저 실행이 완료될 때 로그가 표시됩니다. 이 옵션은 기본적으로 선택되어 있으며 성능에 약간의 영향을 줄 수 있습니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.</p>
로그가 특정 크기보다 클 때 경고 표시	<p>로그의 행 수가 로그 최대 행 수 옵션에 지정된 수를 초과할 때 경고를 표시합니다. 값은 100 - 10000 범위입니다.</p>
로그 자동 지우기	<p>코드를 실행할 때마다 로그를 지웁니다. 이 옵션은 기본적으로 선택되어 있습니다. 이 옵션을 선택하지 않았을 때는 프로그램 편집기에서 로그 지우기 버튼을 사용할 수 있습니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.</p>
로그 추가	<p>기존 로그에 새로운 로그 정보를 추가합니다. 프로그램, 작업, 또는 프로그램과 작업 모두에 대한 로그를 추가하도록 선택할 수 있습니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.</p>
프로그램 탭 레이아웃	<p>작업 영역에서 로그, 결과, 출력 데이터 및 리스트 탭의 배열을 지정합니다.</p>
미리 보기 섹션에 탭 표시	<p>코드 실행 후 프로그램 탭의 미리 보기 섹션에 포함할 탭을 지정합니다.</p> <p>주: 이 옵션은 옵션을 선택한 이후에 연 프로그램에 대해서만 적용됩니다.</p>

결과 우선 설정 지정

결과 페이지에서 해당 옵션을 설정할 수 있습니다.

옵션	설명
결과가 특정 크기보다 클 때 경고 표시	최대 크기 옵션에 지정된 크기보다 큰 결과 파일을 열려고 하면 경고 메시지를 표시합니다. 0.5MB와 10MB 사이의 값을 입력할 수 있습니다. 기본값은 3MB입니다.
목차 생성	HTML 출력 목차를 생성합니다.
HTML 출력 생성	결과를 HTML 형식으로 생성합니다. 기본 결과 출력형식입니다. 이 옵션을 선택 취소하면 프로그램을 실행할 때 결과 탭이 표시되지 않습니다.
HTML 출력 스타일 지정	HTML 형식의 결과에 적용되는 스타일을 지정할 수 있습니다. 이 옵션을 선택하지 않으면 SAS Studio에서는 응용 프로그램 테마와 연결된 스타일을 사용합니다. 이 옵션을 선택하면 출력 스타일 드롭다운 리스트를 사용하여 결과에 적용될 스타일을 선택할 수 있습니다.
HTML 그래프를 SVG로 생성	HTML 출력에서 PNG 그래프 대신 SVG 그래프를 생성합니다. SVG 그래프는 확대 및 축소할 때 선명도가 유지됩니다.
액세스 가능한 그래프 옵션 사용 설정	ODS 그래픽에서 생성된 접근성 메타데이터를 그래프에 추가합니다. 장애가 있는 사용자는 SAS Graphics Accelerator를 사용하여 접근성 메타데이터에 액세스합니다. 이 옵션은 SAS 9.4M4 이상을 실행하고 있을 때만 사용할 수 있습니다. 자세한 내용은 <i>SAS Output Delivery System: User's Guide</i> 의 ODS HTML5 문 을 참조하십시오.
PDF 출력 생성	결과를 PDF 형식으로 생성합니다. 이 옵션은 기본적으로 선택되어 있지 않으며 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
출력 스타일	PDF 형식의 결과에 적용되는 스타일을 표시합니다. 결과에 적용되는 스타일을 변경하려면 드롭다운 리스트에서 다른 스타일을 선택합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
목차 생성	PDF 파일에 목차를 생성합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
액세스 가능한 PDF 옵션 사용 설정	화면 판독기와 같은 보조 기술로 파일에 액세스할 수 있도록 PDF 파일에 접근성 메타데이터를 추가합니다. 메타데이터가 추가되면 이 파일은 "태그가 지정된 PDF"라고 하며 PDF/UA(PDF/Universal Accessibility) 출력형식을 따릅니다.

옵션	설명
WORD 출력 생성	Microsoft Word 형식으로 결과가 생성됩니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
출력 스타일	Word 형식의 결과에 적용할 스타일을 표시합니다. 결과에 적용되는 스타일을 변경하려면 드롭다운 리스트에서 다른 스타일을 선택합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
RTF 출력 생성	결과를 RTF 형식으로 생성합니다. 이 옵션은 기본적으로 선택되어 있지 않으며 표준 영역을 실행하고 있을 때만 사용할 수 있습니다. 주: Microsoft는 RTF 규격에 대한 성능 향상을 중단했습니다. WORD 출력 생성 옵션을 사용하여 출력을 생성하는 것이 좋습니다. WORD 출력 생성 옵션을 사용하면 출력에 액세스 가능한지도 확인할 수 있습니다.
출력 스타일	RTF 형식의 결과에 적용되는 스타일을 표시합니다. 결과에 적용되는 스타일을 변경하려면 드롭다운 리스트에서 다른 스타일을 선택합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
EXCEL 출력 생성	결과를 Microsoft Excel 형식으로 생성합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
출력 스타일	Excel 형식의 결과에 적용되는 스타일을 표시합니다. 결과에 적용되는 스타일을 변경하려면 드롭다운 리스트에서 다른 스타일을 선택합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
PPT 출력 생성	결과를 Microsoft PowerPoint 형식으로 생성합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
출력 스타일	PowerPoint 형식의 결과에 적용되는 스타일을 표시합니다. 결과에 적용되는 스타일을 변경하려면 드롭다운 리스트에서 다른 스타일을 선택합니다. 이 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.
<p>.....</p> <p>주: 사용자 정의 스타일을 사용하려면 SAS Studio 출력 환경을 사용자 정의해야 합니다. 자세한 내용은 부록 4, "사용자 정의된 출력 환경" (191페이지)에서 참조하십시오.</p> <p>.....</p>	

편집기 우선 설정 지정

편집기 페이지에서 SAS 및 XML 언어 편집기에 대한 옵션을 설정할 수 있습니다.

옵션	설명
Snippet 축약어 사용	@ <i>snippet-name</i> 을 입력하거나, @를 입력한 후 팝업 창을 사용하여 내 Snippet 폴더에 있는 Snippet의 리스트에서 Snippet을 선택하는 방법으로 프로그램에 Snippet을 삽입할 수 있습니다. 이 옵션은 기본적으로 선택되어 있습니다. 자세한 내용은 “Snippet 생성” (72페이지) 에서 확인하십시오.
편집기 옵션	코드 편집기의 편집 및 표시 옵션과 모양 옵션을 지정할 수 있습니다. 자세한 내용은 “일반 코드 편집기 우선 설정 지정” (180페이지) 및 “코드 편집기에서 모양 옵션 설정” (181페이지) 에서 확인하십시오.
구문 강조 표시 사용	구문의 각 요소를 식별하기 쉽도록 코드 편집기의 텍스트를 서로 다른 색상으로 표시합니다. 이 옵션은 기본적으로 선택되어 있습니다.
키워드에 마우스를 올리면 힌트를 표시합니다.	프로그램의 올바른 SAS 키워드 위에 마우스 포인터를 놓을 때 구문 도움말을 표시합니다. 이 옵션이 선택되어 있지 않으면 키워드를 마우스 오른쪽 버튼으로 클릭하고 구문 도움말 을 선택하여 구문 도움말을 볼 수 있습니다. 이 옵션은 기본적으로 선택되어 있지 않습니다.

일반 코드 편집기 우선 설정 지정

편집기 페이지에서 **편집기 옵션**을 클릭한 다음 **동작 및 모양**을 클릭하여 코드 편집기에 대한 편집 및 표시 옵션을 설정합니다.

옵션	설명
자동 완성 리스트 표시	코드 편집기의 자동 완성 기능을 설정합니다. 이 기능은 사용자가 실제로 키워드를 모두 입력하기 전에 입력하려고 하는 다음 키워드를 예측하여 완성해 주는 기능입니다. 자세한 내용은 “자동 완성 기능 사용” (43페이지) 에서 확인하십시오.
자동 들여쓰기	새로운 행을 이전 행의 들여쓰기 공간만큼 자동으로 들여씁니다.
블록 주석 자동 삽입	/*를 입력하고 Enter 키를 누르면 여러 줄로 된 주석을 삽입할 수 있도록 닫는 주석 태그를 자동으로 삽입합니다.
겹쳐쓰기 모드 사용	코드 편집기에 입력할 때 기존 문자를 새로운 문자로 바꿉니다.
자동 쌍 따옴표	따옴표를 입력할 때 자동으로 닫는 따옴표를 삽입하고 커서를 두 따옴표 사이에 놓습니다.
일치 괄호 강조 표시	커서를 괄호에 놓으면 일치 괄호를 강조 표시합니다.

옵션	설명
자동 쌍 괄호	왼쪽 괄호를 입력할 때 자동으로 닫는 괄호를 삽입하고 커서를 두 괄호 사이에 놓습니다.
코드 접기 사용	코드 편집기에서 코드 영역을 펼치고 접을 수 있습니다.
라인 강조 표시 사용	행으로 이동 기능을 사용할 때 사용자가 지정한 코드 행을 자동으로 강조 표시합니다.
들여쓰기 안내자 표시	들여쓰는 코드 섹션의 시작 및 끝 위치를 나타내는 세로 선을 표시합니다.
행 길이 가이드 표시	프로그램 편집기에서 지정한 칼럼 위치에 세로 선을 표시합니다. 기본 칼럼 위치는 80입니다.
숨김 문자 표시	프로그램 편집기에서 공백 및 캐리지 리턴과 같은 문자를 표시합니다.
행 번호 표시	프로그램 탭의 맨 왼쪽 칼럼에 행 번호를 표시합니다.
행 번호 인쇄	프로그램을 인쇄할 때 맨 왼쪽 칼럼에 행 번호를 포함합니다.
텍스트 줄바꿈	페이지의 오른쪽 가장자리를 지나서 계속되는 텍스트를 다음 행에 표시합니다.
탭 너비	탭 문자를 삽입할 때 텍스트에 삽입되는 공백 수를 지정합니다. 기본값은 탭 문자당 네 개의 공백입니다.
탭을 공백으로 대체	단일 탭 문자 대신 탭 너비 옵션에 지정된 수만큼의 공백을 삽입합니다.

코드 편집기에서 모양 옵션 설정

코드 편집기에서 SAS, Python 및 XML 코드에 대한 모양 옵션을 설정할 수 있습니다. **편집기** 페이지에서 **편집기 옵션**을 클릭하고 **SAS 색상 및 글꼴**, **Python 색상 및 글꼴** 또는 **XML 색상 및 글꼴**을 클릭합니다.

옵션	설명
구문 테마	코드 편집기에서 밝은 배경 또는 어두운 배경을 사용할지 여부를 지정합니다.
글꼴	코드 편집기에서 사용되는 글꼴을 지정합니다.
편집기 배경	코드 편집기에서 사용되는 배경색을 지정합니다.

옵션	설명
구문 강조 표시 사용	코드 편집기에서 특정 파일 요소에 사용되는 텍스트 스타일 및 배경색을 지정할 수 있습니다.

테이블 우선 설정 지정

테이블 페이지에서 테이블 뷰어에 대한 옵션을 설정할 수 있습니다.

옵션	설명
라이브러리를 펼칠 때 표시되는 테이블 수 제한	탐색 영역의 라이브러리 섹션에서 라이브러리를 펼칠 때 표시되는 테이블 수를 제한합니다. 표시된 테이블 상자를 사용하여 표시할 최대 테이블 수를 지정합니다. 테이블 리스트 끝에 있는 자세히 를 클릭하여 표시되지 않은 추가 테이블을 볼 수 있습니다. 이 옵션에 대한 변경 사항은 라이브러리 트리를 새로 고침하거나 SAS Studio를 다시 시작할 때 적용됩니다.
칼럼 머리글	테이블 뷰어에 칼럼 이름 또는 레이블이 표시되는지 여부를 지정합니다. 칼럼 이름은 기본적으로 표시됩니다.
페이지당 행 수	데이터 페이지당 표시할 행 수를 지정합니다. 기본값은 200입니다. 주: 테이블 뷰어 도구 모음에서  을 클릭하고 행 페이징 을 선택하면 테이블 뷰어에 열려 있는 테이블에 대해 이 옵션을 오버라이드할 수 있습니다.
표시된 칼럼	테이블 뷰어에 표시할 최대 칼럼 수를 지정합니다. 기본값은 100입니다. 표시되는 칼럼 수를 늘리면 성능에 영향을 줄 수 있습니다.
SAS 변수 이름 정책	SAS 변수 이름에 적용할 규칙 세트 중 하나를 지정할 수 있습니다. <ul style="list-style-type: none"> ■ ANY 변수 이름은 빈 칸을 포함한 최소 하나의 문자를 포함하여 시작해야 하고 널(null) 바이트는 포함되지 않도록 지정합니다. 변수 이름은 대소문자 혼합 문자 뿐만 아니라 특수 및 다중 바이트 문자도 포함할 수 있습니다. 이름 최대 길이는 32바이트입니다. 이 옵션은 기본적으로 선택되어 있습니다. 앞의 빈 칸은 유지되지만 뒤의 빈 칸은 무시됩니다. ■ V7 변수 이름이 라틴 알파벳 문자(A-Z, a-z) 또는 밑줄 문자로 시작하도록 지정합니다. 빈 칸이나 밑줄을 제외한 특수 문자로 시작할 수 없으며, 특수 SAS 자동 변수 또는 변수 리스트의 이름으로 할당될 수 없습니다. 변수 이름은 대소문자 혼합 문자를 포함할 수 있으며 길이는 최대 32자가 될 수 있습니다.

옵션	설명
	<ul style="list-style-type: none"> ■ UPCASE는 SAS의 이전 버전처럼 변수 이름이 대문자라는 점을 제외하고 변수 이름이 V7과 동일한 규칙을 따르도록 지정합니다.
SAS 멤버 이름 정책	<p>SAS 데이터셋 이름, SAS 데이터 뷰 이름 및 항목 저장소 이름에 적용할 규칙 세트를 다음 중 하나로 지정할 수 있습니다. 이 옵션에 대한 변경 사항은 SAS Studio를 시작하거나 SAS 세션을 재설정할 때 적용됩니다.</p> <ul style="list-style-type: none"> ■ EXTEND(확장 멤버 이름) - 이름이 최소 하나의 문자(문자, 숫자, 유효한 특수 문자 및 국가별 문자)를 포함해야 하고, 공백이나 마침표로 시작할 수 없으며, 널(null) 바이트를 포함할 수 없도록 지정합니다. 이름은 대소문자 혼합 문자를 포함할 수 있으며 길이는 최대 32바이트가 될 수 있습니다. 앞의 빈 칸은 유지되지만 뒤의 빈 칸은 무시됩니다. 특수 문자에는 / \ * ? " < > : - 문자가 포함될 수 없습니다. 이 옵션은 기본적으로 선택되어 있습니다. ■ COMPATIBLE(기본 멤버 이름) - 라틴 알파벳 문자(A-Z, - z) 또는 밑줄 문자로 시작해야 하고 공백이나 밑줄을 제외한 특수 문자를 포함할 수 없도록 지정합니다. 이름은 대소문자 혼합 문자를 포함할 수 있으며 길이는 최대 32자가 될 수 있습니다.

작업 우선 설정 지정

작업 페이지에서는 생성된 SAS 코드 및 **SAS Studio** 작업 공간의 작업 레이아웃에 대한 옵션을 설정할 수 있습니다. 이러한 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.

옵션	설명
작업 코드에 대한 머리글 주석 생성	생성된 SAS 작업 코드 앞에 주석을 추가합니다.
생성 코드 출력형식 자동 지정	작업에 의해 생성되는 코드의 출력형식을 자동으로 지정합니다.
미리 보기 섹션 보기	작업 실행 후 작업 탭의 미리 보기 섹션을 표시할지 여부와 미리 보기에 포함할 탭을 지정합니다.

플로우 우선 설정 지정

플로우 페이지에서 플로우에 대한 옵션을 설정할 수 있습니다. 이러한 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.

미리 보기 탭 레이아웃	플로우 캔버스와 관련하여 미리 보기 탭이 표시되는 방식을 지정합니다. 기본적으로 미리 보기 탭은 가로 또는 흐름 캔버스 아래에 표시됩니다.
--------------	---

백그라운드 실행 우선 설정 지정

백그라운드 실행 페이지에서 배경 실행 옵션을 설정할 수 있습니다. 이러한 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.

로그 및 출력 파일의 위치	<p>출력 및 로그 파일을 저장할 위치를 지정합니다. 다음 옵션 중에서 선택할 수 있습니다.</p> <ul style="list-style-type: none"> ■ 파일과 동일한 이름 및 위치 사용 - 로그 및 출력 파일에 프로그램 파일의 이름 및 위치를 사용합니다. ■ 출력 및 로그 파일 이름을 위한 프롬프트 - 로그 및 출력 파일을 저장할 위치를 지정하라는 메시지를 표시합니다. ■ 이름 및 위치 지정 - 로그 및 출력 파일의 이름과 위치를 지정할 수 있습니다.
----------------	--

로그 또는 출력 파일이 존재함	<p>로그 및 출력 파일이 이미 존재하는 경우 백그라운드 실행을 처리하는 방법을 지정합니다. 다음 옵션 중에서 선택할 수 있습니다.</p> <ul style="list-style-type: none"> ■ 기존 파일 바꾸기 - 동일한 이름의 기존 로그 및 출력 파일을 자동으로 바꿉니다. ■ 실행 취소 - 동일한 이름의 기존 로그 및 출력 파일이 있을 때 백그라운드 실행을 취소합니다. ■ 타임스탬프를 자동으로 추가는 각 파일에 대해 고유한 파일 이름을 생성하여 모든 로그 및 출력 파일을 저장합니다. 로그 및 출력 파일은 program-name(YYYY-MM-DD HH:MM:SS)로 저장됩니다. ■ 프롬프트 - 백그라운드 실행을 수행하기 전에 기존 로그 및 출력 파일에 대해 바꾸거나 새로운 위치를 지정할지를 확인하는 메시지 창이 표시됩니다. 취소를 선택하면 백그라운드 실행이 취소됩니다.
------------------	--

질의 우선 설정 지정

질의 페이지에서 질의에 대한 옵션을 설정할 수 있습니다. 이러한 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.

다음을 사용하여 코드 생성	PROC SQL 또는 PROC FEDSQL을 사용하여 질의 코드를 생성할지를 지정합니다. 기본값은 PROC SQL입니다. 자세한 내용은 “ FedSQL 질의 생성 ” (102페이지)에서 확인하십시오.
PROC SQL의 선택 탭에 표시할 칼럼	테이블 및 칼럼 이름과 함께 선택 탭에 표시되는 칼럼을 지정합니다. 이 옵션은 PROC SQL을 사용하여 질의 코드를 생성하는 질의에만 적용됩니다.
미리 보기 섹션 보기	질의 실행 후 질의 탭의 미리 보기 섹션을 표시할지 여부와 미리 보기에 포함할 탭을 지정합니다.

가져오기 우선 설정 지정

가져오기 페이지에서 데이터 가져오기에 대한 옵션을 설정할 수 있습니다. 이러한 옵션은 표준 영역을 실행하고 있을 때만 사용할 수 있습니다.

미리 보기 섹션 보기	데이터를 가져온 후 가져오기 탭의 미리 보기 섹션을 표시할지 여부와 미리 보기에 포함할 탭을 지정합니다.
--------------------	--

부록 2

표현식 빌더 사용

표현식 구성 187

표현식 구성

데이터에 대한 필터를 생성할 때 또는 질의에서 필터, 조인 또는 계산된 칼럼을 생성할 때 표현식 빌더를 사용할 수 있습니다.

주: 플로우에서 노드에 대한 표현식을 생성 중이면 다음 기능은 표현식 빌더에서 사용할 수 없습니다.

- 표현식 빌더 창의 값 및 로그 영역
- **값 가져오기** 버튼으로 값 리스트에서 값을 선택하고 특정 값을 검색할 수 있습니다.
- **검증** 버튼을 클릭하여 표현식 구문을 검증합니다.

표현식 빌더 창에는 세 가지 주요 영역이 있습니다.

데이터 및 함수 영역	사용 가능한 데이터 및 함수 칼럼에 액세스할 수 있습니다.
표현식 영역	작성하는 표현식이 표시됩니다. 이 영역에서 표현식을 입력할 수도 있습니다.
값 및 로그 영역	값 탭에는 칼럼의 개별 값이 표시됩니다. 로그 탭에는 확인 로그가 표시됩니다. 표현식 빌더를 사용하여 계산된 칼럼을 생성하면 이 영역에 속성 탭도 포함됩니다.

주: 플로우에서 노드에 대한 표현식을 생성 중이면 값 및 로그 영역을 사용할 수 없습니다.

이름을 더블 클릭하여 표현식에 칼럼 및 함수를 추가할 수 있습니다. 칼럼을 마우스 오른쪽 버튼으로 클릭하고 **표현식에 추가**를 선택하거나 표현식 상자에 칼럼을 끌어와서 칼럼 이름을 추가할 수도 있습니다.

칼럼에 대한 고유 값 리스트를 채우려면 **데이터** 탭에서 칼럼을 선택한 다음 **값** 탭에서 **값 가져오기**를 클릭합니다. 값을 더블 클릭하거나 도구 모음의 **표현식에 추가**를 클릭하거나 또는 표현식 상자에 값을 끌어와서 표현식에 값을 추가할 수 있습니다.

주: 테이블 뷰어 위에 있는 표현식 상자에 직접 표현식을 입력할 수도 있습니다.

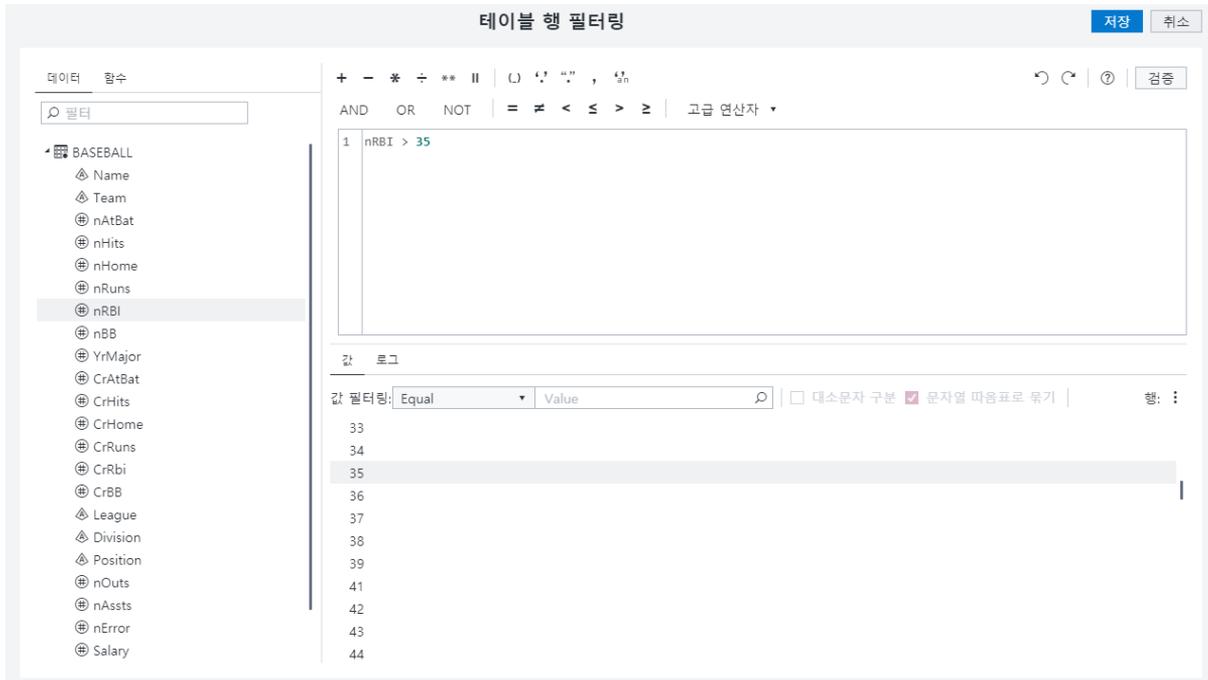
주: 플로우에서 노드에 대한 표현식을 생성 중이면 값 리스트에서 선택하고 특정 값을 검색하는 옵션은 사용할 수 없습니다.

표현식의 구문을 확인하려면 **확인**을 클릭합니다. 그러면 **로그** 탭에 표현식 구문의 오류가 표시됩니다.

주: 다음 상황에서는 구문 검증 옵션을 사용할 수 없습니다.

- 대화식 영역을 사용하는 중입니다. 자세한 내용은 **“영역 이해” (31페이지)**에서 확인하십시오.
- 플로우에서 노드에 대한 표현식을 생성하는 중입니다.

주: = 연산자를 십진수 값과 함께 사용하면 예상한 결과를 얻을 수 없습니다. 십진수 값을 사용한 필터를 생성하려면 **Between** 연산자를 사용하십시오.



부록 3

텍스트 인코딩 옵션 및 언어 매핑

텍스트 인코딩 및 언어 매핑 정보	189
텍스트 인코딩 옵션 및 언어 매핑	189

텍스트 인코딩 및 언어 매핑 정보

다음 표에서는 일부 텍스트 인코딩 옵션과 관련 언어를 나열합니다. 자세한 내용은 “일반 옵션 설정 지정” (175페이지)에서 확인하십시오.

텍스트 인코딩 옵션 및 언어 매핑

텍스트 인코딩 옵션	언어
Windows-1250	(중앙 유럽 언어): 폴란드어, 체코어, 슬로바키아어, 헝가리어, 슬로베니아어, 세르비아어(라틴), 크로아티아어, 보스니아어, 루마니아어, 알바니아어
Windows-1251	(키릴어): 러시아어, 벨로루시어, 불가리아어, 세르비아어(키릴 자모), 마케도니아어, 우크라이나어
Windows-1252	(서유럽어): 아프리카스어, 바스크어, 카탈로니아어, 발렌시아어, 웨일스어, 덴마크어, 독일어, 영어, 스페인어, 바스크어, 핀란드어, 페로어, 프랑스어, 서부 프리지아어, 아일랜드어, 갈리시아어, 인도네시아어, 아이슬란드어, 이탈리아어, 이누이트어, 룩셈부르크어, 말레이어, 노르웨이어(부크몰), 네덜란드어, 노르웨이어(뉘노르스크), 포르투갈어, 케추아어, 로만시어, 북부 사미어, 스웨덴어, 스와힐리어, 츠와나어, 코사어, 줄루어

텍스트 인코딩 옵션	언어
Windows-1253	그리스어
Windows-1254	터키어
Windows-1255	히브리어
Windows-1256	아랍어
Windows-1257	(발트어): 에스토니아어, 라트비아어, 리투아니아어
Windows-1258	베트남어

부록 4

사용자 정의된 출력 환경

개요	191
다른 출력 대상에 대한 출력 생성	192
다른 위치에 결과 전송	192
출력에 대한 사용자 정의 스타일 사용	193
기본값 대신 이미지 출력형식 사용	193
드릴다운 그래프 생성	193
애니메이션 GIF 또는 SVG 이미지 생성	194

개요

다음 작업을 수행하려면 SAS Studio 출력 환경을 사용자 정의해야 합니다.

- 다른 출력 대상에 대한 출력 생성
- 다른 위치에 결과 전송
- 출력에 대한 사용자 정의 스타일 사용
- 기본값 대신 이미지 출력형식 사용
- 드릴다운 그래프 생성
- 애니메이션 GIF 또는 SVG 이미지 생성

SAS Studio 출력 환경을 사용자 정의하려면 먼저 기본 출력 환경을 비활성화하여 시스템 리소스를 보존합니다. 그런 다음, 사용자 고유의 출력 환경을 구성하고 출력 생성에 필요한 SAS 문을 실행합니다. SAS 프로그램에 있는 ODS 문, ODS 프로시저, 또는 ODS 옵션을 사용하여 필요한 환경을 정의합니다.

대표적인 예로, SAS 프로그램에 사용자 정의된 SAS Studio 출력 환경이 필요하다면 다음 단계를 항상 수행해야 합니다.

- 1 ODS 출력에 대한 파일 참조를 생성합니다. SAS Studio에 정의되어 있는 `&_SASWS_` 매크로 변수를 사용하여 다음 문장에 표시된 대로 홈 디렉터리를 참조할 수 있습니다.

```
filename odsout "&_SASWS_/charts";
```

이미지 파일을 별도의 디렉터리에 저장하려면 다음 문장에 표시된 대로 이미지 파일에 대한 두 번째 파일 참조를 생성합니다.

```
filename ods1out "&_SASWS_/charts/images";
```

주: 이미 지정한 디렉터리가 존재해야 하며, 디렉터리에 대한 쓰기 권한이 있어야 합니다.

- 2 시스템 리소스를 보존하려면 다음 문장을 사용하여 기본 출력 환경을 비활성화합니다.

```
ods_all_close;
```

- 3 원하는 ODS 대상을 엽니다. PATH= 옵션을 사용하여 ODS 출력용으로 생성한 파일 참조를 지정합니다. 이미지 파일에 대한 별도의 파일 참조를 생성했으면 GRAPH= 옵션을 사용하여 이미지 출력 파일 참조를 지정합니다. 예:

```
ods html path=odsout gpath=ods1out file="saleschart.html";
```

- 4 출력을 생성하는데 필요한 SAS 문을 실행합니다.

- 5 ODS 대상을 닫습니다.

기본 SAS Studio 출력 환경을 비활성화하면 프로그램 기간 **결과** 탭에 결과가 표시되지 않습니다. 결과는 ODS 대상을 열 때만 결과가 생성됩니다.

다른 출력 대상에 대한 출력 생성

기본 HTML5, PDF 또는 RTF 출력 외에 출력을 생성하려면 사용자 고유의 ODS 대상을 열어야 합니다. 출력 대상의 예로는 HTML, PowerPoint 및 LISTING을 포함합니다. 기본 출력 환경을 비활성화한 후 ODS 문을 사용하여 사용자 고유의 출력 대상을 엽니다. 예:

```
filename odsout "&_SASWS_/charts";
ods_all_close;
ods powerpoint path=odsout file="filename";
```

ODS 문 사전에 액세스하려면 *SAS Output Delivery System: User's Guide*에서 참조하십시오.

다른 위치에 결과 전송

SAS Studio에서 프로그램을 실행할 때 **결과** 탭에서 로컬 컴퓨터로 출력을 다운로드할 수 있습니다. 출력을 다른 위치로 직접 전송하려면 사용자 고유의 ODS 대상을 열어야 합니다. 기본적으로 사용자가 연 ODS 대상에서 생성되는 출력 파일은 홈 디렉터리에 작성됩니다.

결과를 특정 위치에 전송하려면 FILENAME 문을 사용하여 원하는 위치에 파일 참조를 정의합니다. &_SASWS_ 매크로 변수를 사용하여 홈 디렉터리를 참조할 수 있습니다. 파일 참조를 생성한 후 ODS 문에 있는 PATH=*file-reference* 옵션을 사용합니다. 예:

```
filename odsout "&_SASWS_/charts";
```

```
ods _all_ close;
ods html path=odsout file="sales.htm";
```

이때 file sales.htm 및 생성된 모든 이미지 파일은 홈 디렉터리에 있는 하위 디렉터리인 **그 래프**에 작성됩니다.

출력에 대한 사용자 정의 스타일 사용

기업 스타일과 같은 SAS Studio 결과에 대한 사용자 정의 ODS 스타일을 사용하려면 사용자 고유의 ODS 대상을 열어야 합니다. 기본 결과에 대한 사용자 정의 스타일을 지정할 수 없습니다. ODS 문에 있는 STYLE= 옵션을 사용하여 사용자 정의 스타일을 지정합니다. 예:

```
filename odsout "&_SASWS_/charts";
ods _all_ close;
ods html path=odsout file="filename.htm" style=style-name;
```

사용자 정의 스타일을 생성하려면 ODS TEMPLATE 프로시저, CSSStyles 또는 STYLE= 옵션을 사용합니다. 자세한 내용은 *SAS Output Delivery System: User's Guide*에서 참조하십시오.

기본값 대신 이미지 출력형식 사용

기본 외에 이미지 형식을 사용하려면 원하는 출력형식을 지정한 다음 사용자 고유의 ODS 대상을 열어야 합니다. 이미지 출력형식 지정하려면 다음을 수행합니다.

- SAS/GRAPH를 사용하여 그래프를 생성하면 OPTIONS 또는 GOPTIONS 문에 있는 DEVICE= 옵션을 지정합니다. 자세한 내용은 *SAS/GRAPH: Reference*에서 참조하십시오.

주: SAS Studio 5.2 이후 버전에서는 ACTIVEX 및 JAVA 그래픽 장치를 사용하지 않는 것이 좋습니다. 이러한 출력 장치를 사용하면 그래프가 예상대로 작동하지 않을 수 있습니다.

- ODS Graphics를 사용하여 그래프를 생성하면 ODS GRAPHICS 문에 있는 OUTPUTFMT= 옵션을 지정합니다. 자세한 내용은 *SAS Output Delivery System: User's Guide*에서 참조하십시오.

드릴다운 그래프 생성

SAS Studio에서 드릴다운 그래프를 생성하려면 사용자 고유의 ODS 대상을 열어야 합니다. 드릴다운 그래프는 복잡한 데이터를 탐색하는 사용자를 위한 편리한 수단을 제공합니다. 드

릴다운 그래프에서 특정 요소는 활성 링크를 포함합니다. 사용자가 연결된 요소를 클릭하면 연결된 새로운 브라우저 창에 기본으로 나타납니다.

자세한 내용은 다음 문서를 참조하십시오.

- SAS/GRAPH를 사용하여 그래프를 생성하려면 *SAS/GRAPH: Reference*에서 참조하십시오.
- 그래프 템플릿 언어를 사용하여 그래프를 생성하려면 *SAS Graph Template Language: User's Guide*에서 참조하십시오.

애니메이션 GIF 또는 SVG 이미지 생성

SAS Studio에서 애니메이션 그래프를 생성하려면 사용자 고유의 ODS 대상을 열어야 합니다. 애니메이션 그래프는 그래프가 웹 브라우저 또는 애니메이션을 지원하는 다른 뷰어에서 보여질 때 일련의 그래프를 자동으로 표시합니다. 애니메이션은 일련의 그래프를 슬라이드 쇼 방식으로 실행하며 각 그래프 사이에 지연이 발생합니다. 시퀀스는 한번만 실행되며 고정된 횟수만큼 루프하고 멈추거나 무한정으로 루프합니다.

자세한 내용은 다음 문서를 참조하십시오.

- SAS/GRAPH를 사용하여 그래프를 생성하려면 *SAS/GRAPH: Reference*에서 참조하십시오.
- 그래프 템플릿 언어를 사용하여 그래프를 생성하려면 *SAS Graph Template Language: User's Guide*에서 참조하십시오.

부록 5

SAS Studio 명령행

명령행 인터페이스 정보	195
표준 영역의 명령	196
대화식 영역의 명령	201
콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅	204

명령행 인터페이스 정보

명령행 인터페이스를 통해 키보드를 사용하여 SAS Studio에 액세스하고 응용 프로그램을 제어할 수 있습니다. 명령행을 열려면 **보기** ⇒ **명령**을 선택하거나 Windows 환경에서는 Alt+M을, MacOS 환경에서는 Option+M을 누릅니다.

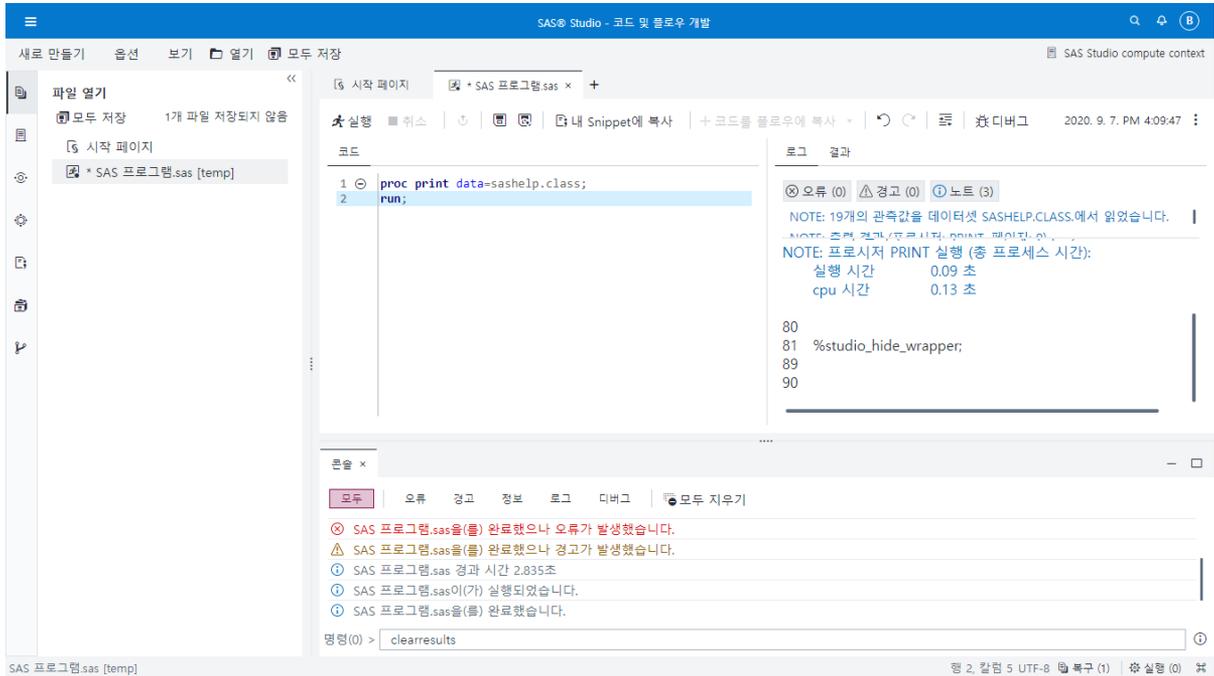
명령에는 옵션 및 인수가 있을 수 있습니다. 옵션에는 앞에 대시가 붙습니다. 공백 문자가 포함된 인수는 따옴표로 묶어야 합니다. 명령의 인수는 지정된 순서로 되어 있어야 하지만, 옵션은 인수 리스트에 혼합되어 임의 순서로 나타날 수 있습니다.

명령행에서 실행할 명령을 입력합니다. 명령행에 텍스트를 입력하면 SAS Studio에서 가능한 명령을 제안합니다. 권장 사항 리스트에서 명령을 선택하려면 위쪽 및 아래쪽 화살표 키를 사용하고 Enter 키를 누릅니다. 명령을 실행하려면 Enter 키를 다시 누릅니다. 여러 명령을 실행하려면 명령을 세미콜론으로 구분하여 입력합니다.

주: 명령 사용은 사용자 인터페이스의 위치에 따라 다릅니다. 예를 들어 플로우 작업에는 명령의 하위 집합만 사용할 수 있습니다.

명령은 순서대로 실행되며 이전 명령이 완료되면 다음 명령이 시작됩니다. 명령이 완료되지 않고 처리가 중지되면 Ctrl+Shift+X를 사용하여 명령 큐를 지웁니다.

명령이 실행될 때 정보가 SAS Studio 콘솔에 추가됩니다. 명령이 실패하면 콘솔에 오류 메시지가 나타납니다. 콘솔을 열려면 **보기** ⇒ **콘솔**을 선택하거나 Alt+C를 누릅니다.



이전 명령을 보려면 Ctrl+위쪽 화살표를 눌러 명령 히스토리에서 한 단계 위로 이동합니다. 명령 히스토리에서 한 단계 아래로 이동하려면 Ctrl+아래쪽 화살표를 누릅니다.

명령 리스트의 실행을 중단하려면 다음을 수행합니다.

- Windows 환경에서는 Ctrl+Shift+X를 누릅니다.
- OS X 환경에서는 Command+Shift+X를 누릅니다.

표준 영역의 명령

다음 명령을 표준 영역에서 사용할 수 있습니다.

테이블 A5.1 표준 영역의 명령

명령 이름	설명
BGSSUBMIT	백그라운드 실행을 시작하고 코드 편집기에서 현재 선택된 코드를 실행합니다. 선택된 코드가 없으면 모든 코드가 실행됩니다.
CANCEL	현재 실행을 중지합니다.
CLEAR	로그 및 결과 탭의 콘텐츠를 삭제합니다.
CLEARCODE	코드 탭에서 모든 코드를 삭제합니다.
CLEARHISTORY	실행 히스토리를 삭제합니다.

명령 이름	설명
CLEARLOG	로그 탭에서 콘텐츠를 삭제합니다.
CLEARRESULTS	결과 탭에서 콘텐츠를 삭제합니다.
CLOSE	선택한 탭을 닫습니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
CLOSEALL	모든 탭을 닫습니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
CONSOLE	SAS Studio용 콘솔을 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
DEBUG	DATA 스텝 디버거를 시작합니다. 주: 이 명령이 작동하려면 커서가 DATA 스텝 코드에 있어야 합니다.
EXPORTTABLE <출력 폴더 경로> <filename.ext> <-csv -dlm -tab><-force>	데이터 뷰가 열려 있고 출력 경로가 지정되어 있으면 테이블 내보내기 창을 엽니다. 테이블 뷰의 파일 이름 및 출력 경로를 지정하면 내보내기 창 이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오. 다음 옵션을 사용할 수 있습니다. <ul style="list-style-type: none"> ■ -csv -dlm -tab은 파일의 내보내기 유형을 지정합니다. 이 옵션을 지정하지 않으면 파일 이름의 확장자에 따라 내보내기 유형이 결정됩니다. ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다.
FIND 타겟 문자열 <- case -c> <-up -u><- word -w>	코드에서 지정된 타겟 문자열을 검색합니다. 다음 옵션을 사용할 수 있습니다. <ul style="list-style-type: none"> ■ -case -c는 검색에서 대소문자를 구분하도록 지정합니다. 기본적으로 대소문자가 무시됩니다. ■ -up -u는 위로 검색합니다. 기본적으로 파일에서 아래쪽으로 검색합니다. ■ -word -w는 전체 단어 일치 검색을 지정합니다. 기본적으로 검색은 단어 경계를 확인하지 않습니다. <p>주: 타겟 문자열의 다음 인스턴스로 이동하려면 NEXT 명령을 사용합니다. 타겟 문자열의 이전 인스턴스로 이동하려면 PREV 명령을 사용합니다.</p>
GOTOLINE <행 번호>	지정된 행 번호로 커서를 이동합니다. 행 번호를 지정하지 않으면 SAS Studio에서 이 정보를 묻는 메시지가 표시됩니다.
KEYS <-edit>	바로 가기 키 창을 엽니다. edit 옵션을 지정하면 기존 바로 가기를 편집할 수 있는 바로 가기 키 관리 창이 열립니다. 이 창에서 바로 가기를 가져오거나 내보내고 모든 바로 가기를 재설정할 수도 있습니다.

명령 이름	설명
	주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEWFLOW	새로운 플로우를 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEWIMPORT	작업 영역의 새로운 탭에서 데이터 가져오기 창을 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEWPROGRAM	작업 영역에 새로운 프로그램 탭을 추가합니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEWQUERY	작업 영역의 새로운 탭에서 질의 창을 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEWTASKTEMPLATE	작업 영역의 새로운 탭에서 빈 작업 템플릿을 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
NEXT	FIND 명령에 지정된 다음 패턴을 찾습니다.
OPENENCODED <파일 경로/filename.ext> <-encoding>	옵션을 지정하지 않으면 파일 열기 창이 열립니다. 파일 이름을 지정하면 이 창이 열리지 않습니다. 특정 파일을 열려면 경로와 파일 이름을 지정합니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오. -encoding 옵션을 사용하여 파일의 인코딩을 지정합니다(예: -UTF8). 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
OPENFILE <파일 경로/filename.ext>	파일 이름을 지정하지 않으면 파일 열기 창이 열립니다. 파일 이름을 지정하면 이 창이 열리지 않습니다. 특정 파일을 열려면 경로와 파일 이름을 입력합니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 확인하십시오. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
PREFERENCES	우선 설정 창을 엽니다. 주: 플로우 작업 시 이 명령을 사용할 수 있습니다.
PREV	FIND 명령에 지정된 이전 패턴을 찾습니다.
REPLACE 타겟 문자열 대체 문자열<-case -c><-up -u><-word -w><-all -a>	지정된 타겟 문자열을 코드에서 검색하고 지정된 콘텐츠로 바꿉니다. 다음 인수를 사용할 수 있습니다. <ul style="list-style-type: none"> ■ -case -c는 검색에서 대소문자를 구분하도록 지정합니다. 기본적으로 대소문자가 무시됩니다. ■ -up -u는 위로 검색합니다. 기본적으로 파일에서 아래쪽으로 검색합니다.

명령 이름	설명
	<ul style="list-style-type: none"> ■ -word -w는 전체 단어 일치 검색을 합니다. 기본적으로 검색은 단어 경계를 확인하지 않습니다. ■ -all -a는 타겟 문자열의 모든 항목을 바꿉니다.
RESET	<p>SAS Studio 세션을 재설정합니다.</p> <p>주: 플로우 작업 시 이 명령을 사용할 수 있습니다.</p>
SAVE <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p> <p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오.</p> <p>파일 이름을 지정하지 않으면 파일의 원래 이름이 유지됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다. 다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVEALL	<p>열려 있는 모든 파일을 저장합니다.</p>
SAVECODE <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p> <p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오.</p> <p>파일 이름을 지정하지 않으면 원래 파일 이름이 사용됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다. 다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVELOG <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p> <p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 확인하십시오.</p> <p>파일 이름을 지정하지 않으면 원래 파일 이름이 사용됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다. 다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVEPACKAGE <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p>

명령 이름	설명
	<p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 확인하십시오.</p> <p>파일 이름을 지정하지 않으면 원래 파일 이름이 사용됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다.</p> <p>다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVERESULTS <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p> <p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 확인하십시오.</p> <p>파일 이름을 지정하지 않으면 원래 파일 이름이 사용됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다.</p> <p>다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVESUMMARY <출력 경로> <filename.ext> <-force> <-encoding>	<p>출력 경로를 지정하지 않으면 다른 이름으로 저장 창이 열립니다.</p> <p>출력 경로를 지정하면 이 창이 열리지 않습니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오.</p> <p>파일 이름을 지정하지 않으면 파일의 원래 이름이 유지됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다.</p> <p>다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SEARCH <-max -restore -off>	<p>검색 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ -max는 미리 보기 영역을 최대화합니다. ■ -restore는 미리 보기 영역을 복원합니다. ■ -off는 검색 탭을 닫습니다. <p>주: 플로우 작업 시 이 명령을 사용할 수 있습니다.</p>
SELECTCODE	코드 탭을 선택합니다.
SELECTLOG	로그 탭을 선택합니다.
SELECTOUTPUT	출력 탭을 선택합니다.
SELECTRESULTS	결과 탭을 선택합니다.

명령 이름	설명
SUBMISSIONS <-max -restore -off>	<p>실행 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> -max는 미리 보기 영역을 최대화합니다. -restore는 미리 보기 영역을 복원합니다. -off는 실행 탭을 닫습니다. <p>주: 플로우 작업 시 이 명령을 사용할 수 있습니다.</p>
SUBMIT	현재 선택한 코드 또는 코드가 선택되어 있지 않으면 편집기의 모든 코드를 실행합니다.
VIEWTABLE <library.table> <-last>	<p>파일 이름을 지정하지 않으면 테이블 선택 창이 열립니다.</p> <p>표시할 테이블을 지정하려면 library.table 인수를 사용합니다. 테이블 선택 창은 열리지 않습니다.</p> <p>테이블을 지정하지 않았을 때 -last 옵션을 사용하면 현재 세션에서 생성된 마지막 테이블이 열립니다.</p> <p>주: 플로우 작업 시 이 명령을 사용할 수 있습니다.</p>
ZOOM <-max> <-restore>	<p>탭 영역의 크기를 지정합니다. 다음 인수를 사용할 수 있습니다.</p> <ul style="list-style-type: none"> -max는 탭 영역의 크기를 최대화하도록 지정합니다. -restore는 탭 영역을 기본 크기로 되돌립니다. <p>주: 플로우 작업 시 이 명령을 사용할 수 있습니다.</p>

대화식 영역의 명령

다음 명령을 대화식 영역에서 사용할 수 있습니다.

테이블 A5.2 대화식 영역의 명령

명령 이름	설명
CANCEL	현재 실행을 중지합니다.
CLEAR	로그 및 결과 탭의 콘텐츠를 삭제합니다.
CLEARCODE	코드 탭에서 모든 코드를 삭제합니다.
CLEARLOG	로그에서 콘텐츠를 삭제합니다.
CLEARRESULTS	결과 탭에서 콘텐츠를 삭제합니다.
CLOSE	선택한 탭을 닫습니다.

명령 이름	설명
CLOSEALL	모든 탭을 닫습니다.
CONSOLE	명령행을 위한 콘솔을 엽니다.
FIND <i>타겟 문자열</i> <-case -c> <-up -u> <-word -w>	<p>코드에서 지정된 타겟 문자열을 검색합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> -case -c는 검색에서 대소문자를 구분하도록 지정합니다. 기본적으로 대소문자가 무시됩니다. -up -u는 위로 검색합니다. 기본적으로 파일에서 아래쪽으로 검색합니다. -word -w는 전체 단어 일치를 검색합니다. 기본적으로 검색은 단어 경계를 확인하지 않습니다.
GOTOLINE <행 번호>	지정된 행 번호로 커서를 이동합니다. 행 번호를 지정하지 않으면 SAS Studio에서 이 정보를 묻는 메시지가 표시됩니다.
KEYS <-edit>	바로 가기 키 창을 엽니다. edit 옵션을 지정하면 기존 바로 가기를 편집할 수 있는 바로 가기 키 관리 창이 열립니다. 이 창에서 바로 가기를 가져오거나 내보내고 모든 바로 가기를 재설정할 수도 있습니다.
NEWPROGRAM	작업 영역에 새로운 프로그램 탭을 추가합니다.
NEXT	FIND 명령에 지정된 다음 패턴을 찾습니다.
OPENENCODED <파일 경로/filename.ext> <-encoding>	<p>옵션을 지정하지 않으면 파일 열기 창이 열립니다.</p> <p>파일 이름을 지정하면 이 창이 열리지 않습니다. 특정 파일을 열려면 경로와 파일 이름을 지정합니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오.</p> <p>-encoding 옵션을 사용하여 파일의 인코딩을 지정합니다(예: -UTF8).</p>
OPENFILE <파일 경로/filename.ext>	<p>파일 이름을 지정하지 않으면 파일 열기 창이 열립니다.</p> <p>파일 이름을 지정하면 이 창이 열리지 않습니다. 특정 파일을 열려면 경로와 파일 이름을 입력합니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅” (204페이지)에서 참조하십시오.</p>
PREFERENCES	우선 설정 창을 엽니다.
PREV	FIND 명령에 지정된 이전 패턴을 찾습니다.
REPLACE <i>타겟 문자열</i> <i>대체 문자열</i> <-case -c> <-up -u> <-word -w> <-all -a>	<p>지정된 타겟 문자열을 코드에서 검색하고 지정된 콘텐츠로 바꿉니다. 다음 인수를 사용할 수 있습니다.</p> <ul style="list-style-type: none"> -case -c는 검색에서 대소문자를 구분하도록 지정합니다. 기본적으로 대소문자가 무시됩니다.

명령 이름	설명
	<ul style="list-style-type: none"> ■ -up -u는 위로 검색합니다. 기본적으로 파일에서 아래쪽으로 검색합니다. ■ -word -w는 전체 단어 일치를 검색합니다. 기본적으로 검색은 단어 경계를 확인하지 않습니다. ■ -all -a는 타겟 문자열의 모든 항목을 바꿉니다.
RESET	SAS Studio 세션을 재설정합니다.
SAVE <상위 폴더 경로> <filename.ext> <-force> <-encoding>	<p>옵션을 지정하지 않으면 다른 이름으로 저장 창이 열립니다. 파일 이름을 지정하면 이 창이 열리지 않습니다. 특정 파일을 저장하려면 폴더 이름을 입력합니다. 자세한 내용은 “콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅 (204페이지)”에서 확인하십시오.</p> <p>파일 이름을 지정하지 않으면 파일의 원래 이름이 유지됩니다. 파일 이름을 변경하려면 filename.ext 인수를 지정해야 합니다.</p> <p>다음 옵션을 지정할 수도 있습니다.</p> <ul style="list-style-type: none"> ■ -force는 기존 파일을 동일한 이름으로 덮어씁니다. ■ -encoding은 -UTF8과 같은 파일의 인코딩을 지정합니다.
SAVEALL	열려 있는 모든 파일을 저장합니다.
SEARCH <-max -restore -off>	<p>검색 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ -max는 미리 보기 영역을 최대화합니다. ■ -restore는 미리 보기 영역을 복원합니다. ■ -off는 검색 탭을 닫습니다.
SHOWLOG <-max -restore -off>	<p>로그 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ -max는 미리 보기 영역을 최대화합니다. ■ -restore는 미리 보기 영역을 복원합니다. ■ -off는 로그 탭을 닫습니다.
SHOWRESULTS <-max -restore -off>	<p>결과 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ -max는 미리 보기 영역을 최대화합니다. ■ -restore는 미리 보기 영역을 복원합니다. ■ -off는 결과 탭을 닫습니다.
SUBMISSIONS <-max -restore -off>	<p>실행 탭을 표시합니다. 다음 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ -max는 미리 보기 영역을 최대화합니다. ■ -restore는 미리 보기 영역을 복원합니다. ■ -off는 실행 탭을 닫습니다.
SUBMIT	현재 선택한 코드 또는 코드가 선택되어 있지 않으면 편집기의 모든 코드를 실행합니다.

명령 이름	설명
VIEWTABLE <library.table> <-last>	테이블 선택 창을 엽니다. 표시할 테이블을 지정하려면 library.table 인수를 사용합니다. 테이블을 지정하지 않았을 때 -last 옵션을 사용하면 현재 세션에서 생성된 마지막 테이블이 열립니다.
ZOOM <-max><-restore>	탭 영역의 크기를 지정합니다. 다음 인수를 사용할 수 있습니다. <ul style="list-style-type: none"> -max는 탭 영역의 크기를 최대화하도록 지정합니다. -restore는 탭 영역을 기본 크기로 되돌립니다.

콘텐츠 경로 지정: SAS 콘텐츠 또는 SAS 컴퓨팅

이러한 명령에 대한 파일 경로를 지정할 때 디렉터리 경로 인수에 sascontent 또는 sascompute로 접두어를 지정하여 파일 또는 폴더의 위치를 지정해야 합니다.

테이블 A5.3 명령 이름 및 예

명령 이름	예
EXPORTTABLE	exporttable "sascontent:/Users/userId/My Folder/Import" "air.csv" exporttable "sascompute:/your-server.com/userId/SAS Studio Content/Import"
OPENFILE	openfile "sascontent:/Public/air.csv" openfile "sascompute:/your-server.com/userId/SAS Studio Content/Import/air.csv"
OPENENCODED	openencoded "sascontent:/Public/air.csv" -encoding openencoded "sascompute:/your-server.com/userId/SAS Studio Content/Import/air.csv" -encoding
SAVE	save "sascontent:/Users/userId/My Folder" "air.csv" -force -encoding save "sascompute:/your-server.com/userId/SAS Studio Content" "air.csv" -force -encoding
SAVECODE	savecode "sascontent:/Users/userId/My Folder" "air.sas" -force -encoding savecode "sascompute:/your-server.com/userId/SAS Studio Content" "air.sas" -force -encoding
SAVELOG	savelog "sascontent:/Users/userId/My Folder" "air.log" -force -encoding savelog "sascompute:/your-server.com/userId/SAS Studio Content" "air.log" -force -encoding
SAVERESULTS	saveresults "sascontent:/Users/userId/My Folder" "air.html" -force -encoding saveresults "sascompute:/your-server.com/userId/SAS Studio Content" "air.html" -force -encoding
SAVEPACKAGE	savepackage "sascontent:/Users/userId/My Folder" "air.pkg" -force -encoding savepackage "sascompute:/your-server.com/userId/SAS Studio Content" "air.pkg" -force -encoding

명령 이름	예
SAVESUMMARY	<pre>savesummary "sascontent:/Users/userId/My Folder" "air.summary" -force -e savesummary "sascompute:/your-server.com/userId/SAS Studio Content" "air</pre>

.....

주: 특정 컨트롤의 구문(경로 정보 포함)의 경우 를 클릭하여 명령줄 인터페이스에서 각 명령에 대한 도움말을 확인합니다.

.....

부록 6

SAS Studio로 전환

SAS Studio 3.x 사용자를 위한 팁	208
차이점 개요	208
프로세스 플로우 업데이트	208
파일 저장	208
질의	209
작업	209
데이터 가져오기	209
우선 설정	209
구성 속성	210
테마	210
일반 사용자 인터페이스 및 기능 차이	210
SAS Enterprise Guide 사용자를 위한 팁	211
차이점 개요	211
SAS@9 Content Assessment 도구 사용	211
SAS Studio에서 SAS Enterprise Guide 프로젝트 열기	211
프로세스 플로우 차이점	212
질의 차이점	212
데이터 가져오기	213
작업 차이	213
SAS Data Integration Studio 사용자를 위한 팁	213
SAS Data Integration Studio 및 SAS Studio	213
플로우 생성	214
작업을 플로우로 마이그레이션	214
라이브러리 및 파일 등록	214
플로우에서 노드 사용	215
SAS Universal Viewer 사용자를 위한 팁	215
SAS Universal Viewer 및 SAS Studio	215
SAS 데이터셋 업로드	216

SAS Studio 3.x 사용자를 위한 팁

차이점 개요

SAS Studio 2021.1에는 오버플로우 메뉴에서 새로운 오버플로우 기능과 오버플로우 메뉴에서 사용할 수 있는 신규 및 기존 옵션이 많이 포함된 업데이트되고 단순화된 사용자 인터페이스가 포함되어 있습니다. 대부분의 경우 작업(task), 질의 및 작업(job)은 SAS Studio 2021.1에서 동일하게 작동하지만,

기본 파일 저장소가 크게 변경되었습니다. 기본 파일 저장소 위치는 SAS 콘텐츠이며 관리자는 파일 시스템에 대한 액세스를 구성해야 합니다.

프로세스 플로우 업데이트

SAS Studio 3.x에서 사용할 수 있었던 프로세스 플로우는 특정 칼럼, 테이블, 작업 및 프로그램에 대한 참조를 기반으로 반복 가능한 프로세스 플로우를 생성할 수 있습니다. SAS Studio 2021.1부터 플로우는 데이터와 작업을 나타내는 노드를 사용하여 데이터에 대한 일련의 작업을 지원하는 메타데이터 구동 설계 도구입니다. 플로우에 "스텝"을 추가하여 노드를 생성합니다.

플로우는 테이블, 칼럼 및 작업에 대한 참조를 하드코딩하지 않고도 일반화된 데이터 플로우를 만들어 복잡한 프로세스를 관리할 수 있도록 설계되었습니다. 노드 속성을 사용하여 플로우에 있는 모든 노드의 특정 속성을 업데이트할 수 있습니다. 포트는 작업 노드의 입력 소스 및 출력 타겟을 나타내며 각 포트와 관련된 매크로 변수를 통해 프로그래밍 방식으로 액세스할 수 있습니다. 플로우 콘텐츠를 보다 유연하게 정의할 수 있도록 노드와 포트를 사용하여 메타데이터 기반 정의를 제공할 수 있습니다.

플로우에서 사용할 수 있는 노드 유형은 탐색 영역의 **스텝** 섹션에서 액세스할 수 있습니다. 새로운 SAS Studio 릴리스가 나올 때마다 **스텝** 섹션에 추가 스텝이 더해집니다. 자세한 내용은 [SAS Studio: 플로우 작업의 "스텝 및 노드 이해"](#)에서 참조하십시오.

현재 저장된 SAS Studio 3.x 프로세스 플로우를 SAS Studio 2021.1 플로우로 변환할 수 없지만 프로세스 플로우의 노드에서 SAS 프로그램을 생성할 수 있습니다. 자세한 내용은 [SAS Studio 3.8: 사용 설명서의 "프로세스 플로우에서 코드 생성"](#)에서 참조하십시오.

파일 저장

SAS Studio 2021.1 사용자의 경우 기본 파일 저장 위치는 SAS 콘텐츠입니다. SAS 콘텐츠 위치는 SAS Drive와 같은 다른 SAS Viya 플랫폼의 응용 프로그램과 통합되므로 응용 프로그램 및 다른 SAS 사용자와 파일을 원활하게 공유할 수 있습니다.

Git 레파지토리 작업과 같은 다른 작업을 수행하기 위해 파일 시스템에 대한 액세스도 필요한 경우 Kubernetes 및 SAS Studio 관리자는 배포 중에 파일 시스템에 대한 액세스를 생성할 수 있습니다. SAS Studio 5.x에서는 기본적으로 SAS 콘텐츠 및 파일 시스템 저장 위치

에 모두 액세스할 수 있습니다. 자세한 내용은 [SAS Studio: Administrator's Guide](#)의 “Creating Persistent File Storage”에서 참조하십시오.

질의

SAS Studio 2021.1에서는 독립 실행형 및 플로우 질의를 모두 생성할 수 있으며 다른 칼럼과 값을 기반으로 하는 계산된 칼럼을 질의에 추가할 수 있습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “질의”](#)에서 참조하십시오.

다음 기준이 충족되면 이전 버전의 SAS Studio에서 저장된 질의를 SAS Studio 2021.1에서 실행할 수 있습니다.

- 원래 질의에 사용된 데이터는 SAS Studio 2021.1에서 사용할 수 있습니다.
- SAS Studio 2021.1에서 사용할 수 있는 데이터에서는 원래 질의에 사용된 칼럼을 찾을 수 없습니다.
- 원래 질의에 포함된 모든 테이블에는 정의된 조인이 있어야 합니다.

저장된 질의를 마이그레이션할 수 없는 경우 SAS Studio는 질의를 프로그램 파일로 변환하고 질의를 마이그레이션할 수 없는 이유를 설명하는 메모를 프로그램에 포함시킵니다.

작업

SAS Studio 3.x에서 사용할 수 있었던 사용자 정의 작업을 포함한 모든 작업을 사용할 수 있습니다. SAS Studio 2021.1에서 사용자 정의 작업을 생성할 수도 있습니다. 자세한 내용은 [“사용자 정의 작업 생성” \(169페이지\)](#)에서 참조하십시오.

SAS Studio 2021.1에서는 작업을 플로우의 스텝으로 포함할 수 없습니다. 그러나 작업 메뉴의 **코드를 플로우에 복사** 버튼을 사용하여 작업에서 자동으로 생성된 코드를 열려 있는 플로우의 새로운 SAS 프로그램 노드에 복사할 수 있습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “플로우에 코드 복사”](#)에서 참조하십시오.

데이터 가져오기

저장된 가져오기 작업 파일(*.CTL)은 SAS Studio 2021.1에서 계속 작동합니다. 플로우의 일부로 데이터를 가져올 때는 가져오기 스텝을 사용합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “외부 파일에서 데이터 가져오기”](#)에서 참조하십시오.

우선 설정

SAS Studio 3.x와 SAS Viya 플랫폼의 SAS Studio에서 **우선 설정** 창의 콘텐츠가 재구성되었습니다. 코드 편집기의 우선 설정과 같은 일부 우선 설정은 일반적인 옵션이며 SAS Studio에만 국한되지 않습니다.

SAS Studio 3.7 또는 이전 버전에서 업그레이드하는 경우 **SAS 변수 이름 정책**의 기본값이 **ANY**로 변경되었습니다. 기본적으로 변수 이름은 빈 칸을 포함한 최소 하나의 문자를 포함하여 시작해야 하고 널(null) 바이트는 포함되지 않도록 지정합니다.

구성 속성

SAS Studio 관리자는 구성 속성을 사용하여 소프트웨어를 사용자 정의할 수 있습니다. SAS Studio 3.x에서는 관리자가 **config.properties** 파일에 이러한 사용자 정의를 지정했습니다. SAS Studio를 다시 설치하거나 다시 구성하면 **config.properties** 파일의 모든 사용자 정의가 손실됩니다. 새로운 SAS Studio 릴리스로 업그레이드하거나 SAS Studio를 다시 구성하기 전에 **config.properties** 파일에 적용한 변경 사항을 기록하십시오. SAS Studio를 다시 설치하거나 다시 구성한 후 SAS Environment Manager에서 이러한 구성 속성을 재설정해야 합니다.

SAS Studio 3.x의 구성 속성은 **webdms**로 시작합니다. 이러한 구성 속성 중 일부는 SAS Viya 플랫폼에서 사용되지 않으며, SAS Viya 플랫폼의 SAS Studio에서 구성 속성은 **sas.studio**로 시작됩니다. 사용 가능한 구성 속성 리스트에 대한 자세한 내용은 [SAS Studio: Administrator's Guide](#)의 "Configuration Properties for SAS Studio"에서 참조하십시오.

테마

SAS Studio 2021.1은 모든 SAS Viya 플랫폼의 제품과 동일한 기본 기업 테마를 사용합니다. 자세한 내용은 "글로벌 설정 생성" (172페이지)에서 참조하십시오.

일반 사용자 인터페이스 및 기능 차이

사용자 인터페이스를 단순화하기 위해 SAS Studio 기능에 대한 일부 옵션이 해당 기능에 대한 도구 모음에서 오버플로우 메뉴로 이동되었습니다(도구 모음에서 로 표시됨).

다음은 SAS Studio 3.x 이후에 추가된 몇 가지 새로운 기능입니다.

- DATA 스텝 디버거 자세한 내용은 "DATA 스텝 디버거 사용" (55페이지)에서 참조하십시오.
- 표현식 빌더 자세한 내용은 "표현식 구성" (187페이지)에서 참조하십시오.
- 작업 정의 및 스케줄링 자세한 내용은 SAS Studio 개발자 가이드: 작업하기에서 참조하십시오.
- 콘솔 창 자세한 내용은 "콘솔 사용" (24페이지)에서 확인하십시오.
- 명령행 인터페이스 자세한 내용은 "명령행 인터페이스 정보" (195페이지)에서 참조하십시오.
- 대화식 영역 자세한 내용은 "영역 이해" (31페이지)에서 확인하십시오.
- FedSQL 질의 자세한 내용은 "FedSQL 질의 생성" (102페이지)에서 참조하십시오.
- 바로 가기 키 관리 자세한 내용은 "바로 가기 키 관리" (34페이지)에서 확인하십시오.
- 탐색 영역의 파일 열기 섹션 자세한 내용은 "열린 파일 액세스" (8페이지)에서 확인하십시오.
- Git 분기 및 병합 자세한 내용은 6장, "SAS Studio의 Git 통합 이해" (133페이지)에서 참조하십시오.

SAS Enterprise Guide 사용자를 위한 팁

차이점 개요

SAS Studio와 SAS Enterprise Guide는 모두 리포트, 그래프 및 차트를 생성할 수 있는 SAS에 포인트앤클릭 인터페이스를 제공합니다. SAS 서버 및 데이터에 액세스합니다. 데이터를 분석합니다. 두 제품 모두 분석 및 리포트에 바로 사용할 수 있는 작업, 색상 코드 SAS 언어 편집기, Git의 가장 일반적인 일부 기능에 대한 액세스도 포함합니다.

SAS Studio와 SAS Enterprise Guide의 기본 차이점은 SAS Studio가 웹 브라우저를 통해 SAS 코드를 쓰고 실행할 수 있는 도구라는 데 있습니다. SAS Enterprise Guide는 컴퓨터에 설치하는 Microsoft Windows 클라이언트 응용 프로그램입니다.

SAS®9 Content Assessment 도구 사용

SAS®9 Content Assessment는 SAS® 9.4 시스템의 다양한 특성을 이해하는 데 도움이 되도록 설계된 응용 프로그램 컬렉션입니다. 각 응용 프로그램은 SAS®9 시스템에서 관련 정보를 검사하고 주요 상세 정보를 수집하며, 평가의 각 부분에서 결과를 생성합니다.

새로운 릴리스로 업그레이드하기 전에 SAS®9 Content Assessment 도구를 실행해야 합니다. 이 도구는 업그레이드 할 때 발생할 수 있는 문제를 식별합니다. 자세한 내용은 [SAS Content Assessment의 "Executing the Import EG Projects Application"](#)에서 참조하십시오.

SAS Studio에서 SAS Enterprise Guide 프로젝트 열기

현재 SAS Enterprise Guide 프로젝트를 SAS Studio에서 직접 열 수는 없지만 SAS Enterprise Guide 프로젝트의 작업을 SAS Studio 2021.1로 전환하는 다음과 같은 두 가지 옵션을 사용할 수 있습니다.

- SAS Environment Manager를 사용하여 SAS Enterprise Guide 프로젝트를 가져오고 프로젝트에서 각 프로세스 플로우에 대한 SAS Studio 플로우 파일(*.flw)을 생성할 수 있습니다. 대부분의 SAS Enterprise Guide 질의는 SAS Studio에서 질의의 노드로 변환됩니다. 질의의 노드로 변환할 수 없는 질의는 SAS 프로그램 노드로 변환됩니다. 기타 참조 프로그램(*.sas)을 제외한 SAS Enterprise Guide 프로세스 플로우의 모든 실행 가능한 노드는 SAS Studio 플로우에서 SAS 프로그램 노드로 변환됩니다. 노드 간 링크는 유지됩니다. 자세한 내용은 [SAS Viya Platform: Content Migration From SAS 9.4의 "SAS Enterprise Guide"](#)에서 참조하십시오.
- 프로젝트에서 코드를 내보내고 SAS Studio 2021.1에서 코드를 엽니다. 자세한 내용은 [SAS Enterprise Guide: 사용 설명서의 "모든 코드 내보내기"](#)에서 확인하십시오.
- 프로젝트를 SAS Studio 3.8 프로세스 플로우로 변환한 다음 프로세스 플로우에서 SAS 프로그램을 생성합니다. 프로젝트를 프로세스 플로우로 변환하는 데 대한 자세한 내용은

SAS Studio 3.8: *사용 설명서*의 “Generating Code from a Process Flow”에서 참조하십시오.

주:

- SAS Studio에서 액세스가 불가능한 SAS Enterprise Guide 프로젝트의 데이터에 대한 참조를 업데이트할 필요가 있습니다.
- SAS Enterprise Guide에서 개발하거나 변환한 코드를 사용하여 SAS Studio에서 작업을 시작할 때 코드를 평가하고 업데이트하여 새로운 CAS 프로그래밍 언어를 활용할 수 있는지 확인해야 합니다. 자세한 내용은 [Programming Documentation for SAS Viya](#)에서 참조하십시오.

프로세스 플로우 차이점

프로세스 플로우는 SAS Enterprise Guide에서 사용할 수 있으며 SAS Enterprise Guide에서 프로젝트를 생성할 때 자동으로 생성됩니다. 프로젝트에서 여러 추가 프로세스 플로우를 생성할 수도 있습니다. SAS Enterprise Guide의 프로세스 플로우를 사용하면 특정 칼럼, 테이블, 작업 및 프로그램에 대한 참조를 기반으로 하는 둘 이상의 개체 간 관계를 표시할 수 있습니다.

SAS Studio 2021.1에 포함된 플로우는 데이터와 작업을 나타내는 노드를 사용하여 데이터에 대한 일련의 작업을 지원하는 메타데이터 구동 설계 도구입니다. 플로우에 "스텝"을 추가하여 노드를 생성합니다. 플로우는 테이블, 칼럼 및 작업에 대한 참조를 하드코딩하지 않고도 일반화된 데이터 플로우를 만들어 복잡한 프로세스를 관리할 수 있도록 설계되었습니다. 노드 속성을 사용하여 플로우에 있는 모든 노드의 특정 속성을 업데이트할 수 있습니다. 포트는 작업 노드의 입력 소스 및 출력 타겟을 나타내며 각 포트와 관련된 매크로 변수를 통해 프로그래밍 방식으로 액세스할 수 있습니다. 플로우 콘텐츠를 보다 유연하게 정의할 수 있도록 노드와 포트를 사용하여 메타데이터 기반 정의를 제공할 수 있습니다.

플로우에서 사용할 수 있는 노드 유형은 탐색 영역의 **스텝** 섹션에서 액세스할 수 있습니다. 새로운 SAS Studio 릴리스가 나올 때마다 **스텝** 섹션에 추가 스텝이 더해집니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “플로우란?”](#)에서 참조하십시오.

질의 차이점

SAS Enterprise Guide 및 SAS Studio의 질의는 거의 동일한 기능을 제공합니다. 두 제품의 질의는 하나 이상의 테이블을 기반으로 할 수 있으며 다른 칼럼 또는 값을 기반으로 하는 컴퓨팅 칼럼 또는 계산된 칼럼을 포함할 수 있습니다. SAS Enterprise Guide에서 질의 빌더를 사용하여 프로젝트의 일부로 질의를 생성합니다. SAS Studio 2021.1에서는 독립 실행형 질의와 플로우 질의를 모두 생성할 수 있으며, PROC SQL 대신 PROC FEDSQL을 사용하여 질의를 생성하도록 선택할 수 있습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “질의”](#)에서 참조하십시오.

데이터 가져오기

SAS Enterprise Guide 및 SAS Studio를 사용하면 텍스트, CSV, Microsoft Excel 및 기타 여러 PC 기반 데이터베이스 파일을 가져와서 SAS 데이터셋을 생성할 수 있습니다. SAS Enterprise Guide는 데이터 가져오기 마법사를 사용하여 파일을 가져옵니다. SAS Studio에는 두 가지 유형의 가져오기 기능이 있습니다. 빠른 가져오기를 사용하면 몇 가지 옵션을 선택하여 파일 시스템 또는 SAS 콘텐츠에서 데이터를 빠르게 가져올 수 있습니다. 데이터를 플로우로 가져올 수도 있습니다. 이 가져오기 기능은 보다 강력하며 추가 옵션을 제공합니다. 자세한 내용은 “SAS Studio에 데이터 가져오기 정보” (118페이지)에서 참조하십시오.

작업 차이

SAS Enterprise Guide와 SAS Studio에는 모두 간단한 데이터 리스트에서 복잡한 분석 프로시저에 이르기까지 SAS 프로시저에 대한 그래픽 인터페이스인 SAS 작업이 포함되어 있습니다. SAS Enterprise Guide에서 다음과 같은 유형의 작업을 수행할 수 있습니다.

- SAS Enterprise Guide와 함께 제공되는 사전 정의된 작업.
- SAS Studio 작업. 사이트에 SAS Enterprise Guide 및 SAS Studio가 있는 경우 SAS Enterprise Guide에서 SAS Studio 작업에 액세스할 수 있습니다.
- 사용자 정의 작업. 사이트에 대한 사용자 정의 작업을 개발하고 SAS Enterprise Guide에서 사용할 수 있습니다.

SAS Enterprise Guide 작업과 SAS Studio 작업의 기본 기술은 다르므로 SAS Studio에서 작업 인터페이스를 사용하여 SAS Enterprise Guide 작업을 열 수 없습니다.

사전 정의된 작업, SAS Studio 작업 또는 사용자 정의 작업을 SAS Enterprise Guide에서 SAS Studio로 전환하려면 다음과 같이 수행합니다.

- 1 SAS Enterprise Guide에서 SAS 코드를 내보냅니다.
- 2 SAS Studio에서 내보낸 SAS 코드를 엽니다.

SAS Data Integration Studio 사용자를 위한 팁

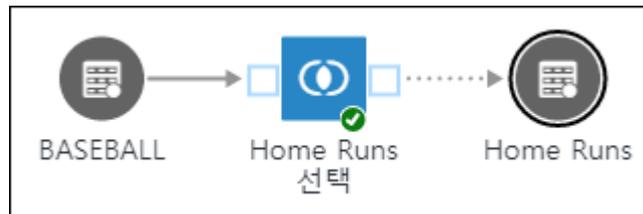
SAS Data Integration Studio 및 SAS Studio

SAS Studio의 데이터 관리 기능은 현재 SAS Data Integration Studio를 직접 대체하지 않습니다. 새로운 SAS Studio 릴리스마다 보다 많은 데이터 관리 기능을 사용할 수 있습니다.

플로우 생성

SAS Studio 플로우는 SAS Data Integration Studio 작업과 유사합니다. 플로우는 실행 순서를 지정하기 위해 연결할 수 있는 일련의 데이터 및 작업 노드에서 작성됩니다. 예를 들어 다음 숫자는 야구 데이터 테이블에서 각 선수에 대해 홈런 수를 선택하는 플로우를 표시합니다.

그림 A6.1 홈런 데이터를 선택하는 플로우



- BASEBALL은 야구팀에 대한 데이터의 입력 테이블로 지정한 테이블 노드입니다.
- Home Runs 선택은 BASEBALL 테이블에서 각 선수에 대한 홈런 수를 선택하는 질의 노드입니다.
- Home Runs는 질의에 대한 출력 테이블을 지정하는 테이블 노드입니다.

플로우 작업에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [SAS Studio: 플로우 작업의 “플로우 생성”](#)
- [SAS Studio: 플로우 작업의 “플로우 열기”](#)
- [SAS Studio: 플로우 작업의 “플로우 탭 이해”](#)
- [SAS Studio: 플로우 작업의 “플로우 실행”](#)

작업을 플로우로 마이그레이션

SAS Data Integration Studio 또는 SAS Management Console의 SAS 패키지로 내보내기 마법사를 사용하여 작업을 내보낼 수 있습니다. 내보내기에 대한 자세한 내용은 [SAS Viya Platform: Content Migration From SAS 9.4의 “SAS Data Integration Studio Jobs and Transformations”](#)에서 참조하십시오.

이러한 작업은 SAS Viya 플랫폼의 SAS Environment Manager에서 가져올 수 있습니다. 가져오는 동안 작업은 SAS Studio 플로우로 변환됩니다. Hadoop 변환을 위해 생성된 코드가 Hadoop REST API를 사용하도록 업데이트됩니다.

라이브러리 및 파일 등록

테이블 또는 파일은 플로우에서 소스 또는 타겟이 될 수 있습니다. 탐색 영역의 라이브러리 섹션에서는 사용자에게 사용 가능한 SAS 라이브러리에 액세스할 수 있습니다. 자세한 내용은 [“라이브러리 작업” \(11페이지\)](#)에서 참조하십시오.

플로우는 파일로 저장됩니다. 탐색 영역의 **탐색기** 섹션에서 폴더 바로 가기, 서버 파일 시스템 및 SAS Content Server 위치의 파일과 폴더에 액세스할 수 있습니다. 자세한 내용은 [“탐색기 사용 ” \(8페이지\)](#)에서 참조하십시오.

플로우에서 노드 사용

SAS Studio 플로우의 노드는 SAS Data Integration Studio 작업에서의 변환과 유사합니다. 플로우 노드는 탐색 영역의 **스텝** 섹션에서 액세스할 수 있습니다. 해당 섹션의 노드만 플로우에서 사용할 수 있습니다. SAS Studio Basic에는 다음 노드 유형이 포함되어 있습니다.

- 내보내기 노드를 사용하여 구분된 파일(*.csv, *.dlm, *.tab, *.tsv), 텍스트 파일 또는 Microsoft Excel 파일(*.xls, *.xlsx)에 데이터를 저장할 수 있습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “데이터를 외부 파일로 내보내기”](#)에서 참조하십시오.
- 파일 노드는 플로우에서 외부 소스 파일을 지정합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “외부 파일을 플로우에 추가”](#)에서 참조하십시오.
- 가져오기 노드는 외부 파일을 SAS 데이터셋으로 변환합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “외부 파일에서 데이터 가져오기”](#)에서 참조하십시오.
- 테이블 노드는 플로우의 SAS 데이터셋을 참조합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “SAS 라이브러리에서 플로우로 테이블 추가”](#)에서 참조하십시오.
- SAS 프로그램 노드는 새로운 SAS 프로그램, 저장된 SAS 프로그램 또는 플로우의 스크립트를 지정합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “SAS 프로그램 스텝: 플로우에서의 SAS 코드 작성”](#)에서 참조하십시오.
- 질의 노드는 지정한 기준에 따라 하나 이상의 테이블에서 데이터를 추출합니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “질의”](#)에서 참조하십시오.
- 정렬 노드를 사용하면 하나 이상의 칼럼 값을 기준으로 플로우에서 출력 데이터를 정렬할 수 있습니다. 자세한 내용은 [SAS Studio: 플로우 작업의 “정렬”](#)에서 참조하십시오.

플로우 캔버스 아래의 노드 속성을 사용하여 앞의 항목에 설명된 대로 노드의 속성 및 콘텐츠를 지정할 수 있습니다.

SAS Universal Viewer 사용자를 위한 팁

SAS Universal Viewer 및 SAS Studio

SAS Universal Viewer와 SAS Studio를 사용하여 SAS 데이터셋을 보고, 데이터를 필터링하며, 특정 칼럼을 표시하거나 숨기고, 데이터를 정렬하고, SAS 프로그램을 볼 수 있습니다.

SAS Studio와 SAS Universal Viewer의 주요 차이점은 SAS Studio에서 데이터를 보기 위해 먼저 데이터를 SAS Studio 인터페이스에 업로드해야 하는 데 있습니다.

SAS 데이터셋 업로드

로컬 컴퓨터에 저장된 SAS 데이터셋을 업로드하고 보려면 다음을 수행하십시오.

- 1 탐색 영역의 **탐색기** 섹션을 클릭하고 데이터를 업로드할 폴더로 이동합니다. 탐색기 섹션 도구 모음에서 **T**를 클릭합니다.
- 2 **파일 업로드** 창에서 **+**를 클릭하고 업로드할 데이터를 검색합니다. **열기**를 클릭한 다음 **업로드**를 클릭합니다. 업로드된 데이터는 탐색 영역의 **탐색기** 섹션에 있는 파일 리스트에 추가됩니다.
- 3 데이터셋을 보려면 데이터셋 이름을 더블 클릭하십시오. 테이블 뷰어에서 데이터셋이 열립니다.

데이터 보기, 정렬 또는 필터링에 대한 자세한 내용은 [“테이블 뷰어 정보” \(105페이지\)](#)에서 참조하십시오.

SAS 데이터셋 외에도 다른 유형의 데이터 파일을 업로드할 수 있습니다. 자세한 내용은 [“데이터 가져오기” \(118페이지\)](#)에서 참조하십시오.

부록 7

SAS Studio 기능 요약

SAS Studio 라이선스 정보	217
플로우 기능 요약	217
사용자 정의 스텝 기능 요약	222

SAS Studio 라이선스 정보

SAS Studio는 다음 세 가지 라이선스로 제공됩니다.

- SAS Studio Basic
- SAS Studio Analyst
- SAS Studio Engineer

각 라이선스의 기능은 조금씩 다릅니다.

플로우 기능 요약

이 테이블에는 각 SAS Studio 라이선스에서 플로우에 사용할 수 있는 스텝이 나열되어 있습니다.

테이블 A7.1 플로우 스텝 요약

스텝	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
사용자 정의 스텝	사이트의 사용자가 특정 작업을 완료할 수 있는	✓	✓	✓

스텝	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	사용자 인터페이스를 생성할 수 있습니다.			
내보내기	데이터를 외부 파일로 내보냅니다.	✓	✓	✓
파일	외부 파일을 참조합니다.	✓	✓	✓
Geocoding	주소 데이터를 지도 위치로 변환하는 방법을 제공합니다.	✓	✓	✓
가져오기	외부 파일을 SAS 데이터셋으로 변환합니다. 구분된 파일 및 Microsoft Excel 파일 가져오기를 지원합니다. 고정 너비 파일 가져오기 및 외부 파일에서 칼럼 구조 메타 데이터를 로드하는 기능을 지원합니다.	✓	✓	✓
칼럼 관리	입력 테이블에서 칼럼 서브셋을 선택하고 출력 테이블에 칼럼을 쓸 수 있습니다.	✓	✓	✓
Python 프로그램	플로우에서 Python 프로그램 작성 및 실행을 지원합니다.	✓	✓	✓
질의	지정한 기준에 따라 하나 이상의 테이블에서 데이터를 추출합니다.	✓	✓	✓
중복 제거	소스 테이블에서 중복 행을 제거할 수 있습니다.	✓	✓	✓
SAS 프로그램	플로우에서 SAS 프로그램 작성	✓	✓	✓

스텝	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	및 실행을 지원 합니다.			
정렬	하나 이상의 칼럼 값을 기준으로 데이터를 정렬할 수 있습니다.	✓	✓	✓
테이블	소스 테이블을 참조하거나 출력 테이블의 물리적 위치를 지정합니다.	✓	✓	✓
주소 확인 및 주소 지오코딩 - Loqate	주소 정보를 확인하고 Loqate Verify Address API에서 Geocode 좌표를 얻을 수 있습니다.	✓	✓	✓
이메일 주소 확인 - Loqate	Loqate Verify Email Addresses API를 사용하는 이메일 주소를 확인할 수 있습니다.	✓	✓	✓
전화 번호 확인 - Loqate	Loqate Verify Phone Numbers API를 사용하는 전화 번호를 확인할 수 있습니다.	✓	✓	✓
막대 그래프	다른 그래프 변수의 값 사이에서 숫자 값 또는 통계량을 비교하는 가로 또는 세로 막대 그래프를 생성합니다.		✓	✓
막대-선 그래프	막대 선 그래프 오버레이가 있는 수직 막대 그래프를 생성합니다.		✓	✓
행 분기	지정한 조건에 따라 테이블을		✓	✓

스텝	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	두 개의 출력 테이블로 분할합니다.			
칼럼 계산	입력 테이블을 기반으로 출력 테이블을 생성합니다. 출력 테이블에서 칼럼을 생성하거나 수정할 수 있습니다.		✓	✓
행 필터	입력 테이블에서 행의 서브셋을 선택하고 단일 출력 테이블에 행을 쓸 수 있습니다.		✓	✓
행 삽입	입력 테이블에서 출력 테이블로 행을 삽입합니다.		✓	✓
일원빈도분석	데이터에서 빈도 테이블을 생성합니다. 이 스텝을 사용하여 이항 및 카이제곱 검정을 수행할 수도 있습니다.		✓	✓
데이터 순위화	테이블의 행에서 하나 이상의 숫자 변수에 대한 순위를 계산하고 출력 테이블에 순위를 포함합니다.		✓	✓
데이터 전치	입력 테이블의 선택된 칼럼을 출력 테이블의 행으로 바꿉니다.		✓	✓
행 병합 (UNION)	여러 소스의 데이터를 단일 타겟 테이블로 결합할 수 있습니다.		✓	✓
의사결정 실행	게시된 결정을 SAS			✓

스텝	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	Intelligent Decisioning에서 플로우로 추가할 수 있습니다.			
SCD 구현	SCD(저속 변경 차원) 프로세스를 사용하여 데이터로 타겟 차원 테이블에 로드할 수 있습니다.			✓
테이블 로드	소스 테이블에서 타겟 테이블로 로드할 수 있습니다. 삽입, 업데이트 또는 두 가지 방법을 모두 적용하여 새 테이블을 생성할 수 있습니다.			✓
테이블 병합	소스 테이블의 값을 기반으로 타겟 테이블의 칼럼을 변경할 수 있습니다. 소스 및 타겟 테이블은 ANSI SQL 표준을 지원하는 DBMS 테이블이어야 합니다. 테이블 병합 스텝은 업데이트 및 삽입 작업을 한 스텝으로 결합하고 In-Database 처리만 지원합니다.			✓

이 테이블에는 SAS Studio의 각 버전에서 사용할 수 있는 플로우 기능이 요약되어 있습니다.

테이블 A7.2 플로우 기능 요약

기능	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
플로우 노트	플로우에 대한 메모를 만들거나	✓	✓	✓

기능	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	플로우 노드에 메모를 추가합니다.			
공개 API	프로그램 또는 플로우를 식별하고 프로그램 또는 플로우에서 생성된 SAS 코드를 반환합니다.	✓	✓	✓
스윙레인	노드를 스윙레인으로 그룹화하고 스윙레인이 실행되는 순서를 지정하여 플로우의 실행 순서를 제어합니다.	✓	✓	✓
코드 최적화	특정 스텝 조합은 자동으로 최적화될 수 있습니다.		✓	✓
하위 플로우	플로우의 SAS 콘텐츠에서 플로우를 참조하는 기능.		✓	✓

사용자 정의 스텝 기능 요약

사용자 정의 스텝은 표준 영역에서 사용할 수 있습니다. 사용자 정의 스텝은 대화식 영역에서 사용할 수 없습니다.

다음 테이블에는 사용자 정의 스텝에 사용할 수 있는 기능이 나열되어 있습니다.

테이블 A7.3 사용자 정의 스텝 기능 요약

기능	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
계단식 프롬프트	프롬프트 계층을 사용하여 컨트롤 간에 데이터 종	✓	✓	✓

기능	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	속 관계를 생성할 수 있습니다.			
출력 칼럼 구조 정의	플로우에서 출력 테이블 생성 스텝을 실행하지 않고도 칼럼의 계단식 배열 방법을 알 수 있도록 포트를 New, Dropped, Keep으로 표시합니다.	✓	✓	✓
종속 관계	스텝 작성자가 다른 프롬프트의 값을 기반으로 프롬프트의 상태를 지정할 수 있습니다.	✓	✓	✓
디자이너는 디자이너, 프롬프트 UI 및 프로그램 탭으로 구성됩니다.	스텝 작성자가 사용자 정의 스텝을 생성하거나 편집할 수 있습니다.	예(디자이너 탭 제외) 주: 디자이너 탭은 디자이너의 끌어서 놓기 방식의 사용자 인터페이스입니다. SAS Studio Basic에서는 이 탭을 사용할 수 없습니다. 사용자가 프롬프트 UI에서 JSON 코드를 사용하여 사용자 정의 작업을 만들어야 합니다.	✓	✓
독립 실행형 사용자 정의 스텝	플로우 외부에서 사용자 정의 스텝을 실행합니다.	✓	✓	✓
SAS 서버 또는 SAS 콘텐츠에 사용자 정의 스텝 업로드	SAS 서버 또는 SAS 콘텐츠에서 사용자 정의 스텝에 액세스할 수 있습니다.	✓	✓	✓
플로우에 사용자 정의 스텝 추가	플로우에 사용자 정의 스텝을 추		✓	✓

기능	설명	SAS Studio Basic	SAS Studio Analyst	SAS Studio Engineer
	가할 수 있습니다.			