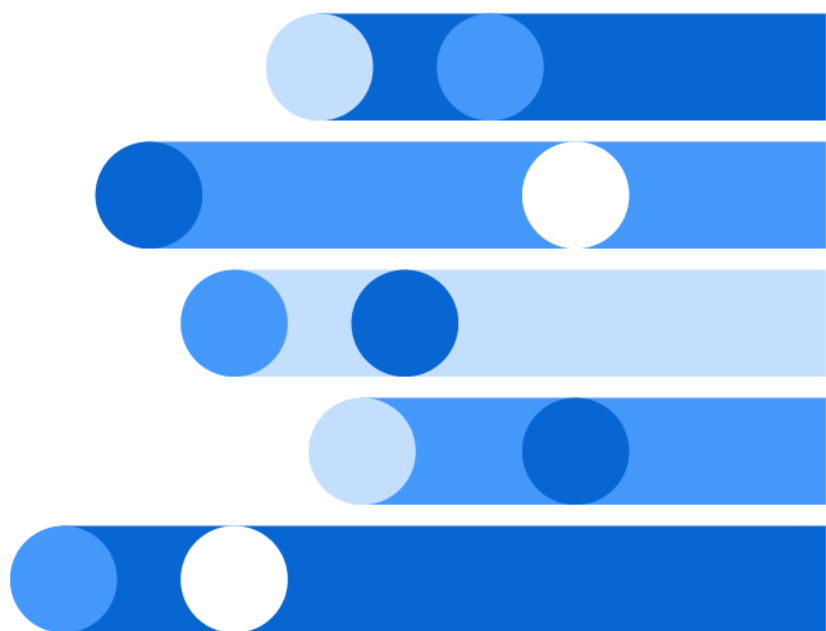




SAS[®] Viya[®] Workbench: Pierwsze kroki



Prawidłowa nota bibliograficzna dla tego podręcznika brzmi następująco: SAS Institute Inc. 2024. SAS® Viya® Workbench: Pierwsze kroki. Cary, NC: SAS Institute Inc.

SAS® Viya® Workbench: Pierwsze kroki

Copyright © 2024, SAS Institute Inc., Cary, NC, USA

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

Dla egzemplarza drukowanego: 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.

Dla e-booka lub tekstu pobranego z Internetu: 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

March 2026

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_002-P1:workbenchgs

Spis treści

Rozdział 1 / Używanie SAS Viya Workbench	1
SAS Viya Workbench - Informacje	1
Co to jest SAS Viya Workbench Advanced?	2
Otwórz SAS Viya Workbench	2
Tworzenie przestrzeni roboczej	3
Zrozumienie statusu przestrzeni roboczej	4
Klonowanie repozytorium przykładów	5
Zmiana zasobów obliczeniowych	6
Usuwanie przestrzeni roboczej	7
Przełączanie zintegrowanych środowisk projektowych	8
Zarządzanie przechowywaniem	9
Zatrzymywanie przestrzeni roboczej	15
Dostęp do dokumentacji SAS-owej	15
Rozdział 2 / Praca z Gitem i SSH	17
Używanie Gita z poziomu Workbenchu	17
Klonowanie publicznego repozytorium GitHub bez instalacji	17
Korzystanie z SSH w Workbenchu	18
Konfigurowanie globalnych ustawień Gita	19
Komunikat: Błąd krytyczny: Wykryto wątpliwą własność	19
Wysyłanie plików do zdalnego repozytorium Gita	20
Rozdział 3 / Praca w kodzie Visual Studio	23
Otwórz przestrzeń roboczą w programie Visual Studio Code	23
Dostęp do SAS Viya Workbench w kodzie Visual Studio	24
Uruchom przykład Pythona	26
Uruchom przykładowy program SAS-owy	27
Dostęp do bibliotek SAS-owych	30
Otwieranie terminala w kodzie Visual Studio	31
Wybór schematu kolorów SAS-owych	32
Utwórz notatnik SAS-owy	32
Korzystanie z funkcji AutoExec w kodzie Visual Studio	34
Rozdział 4 / Praca w notesie Jupyter	35
Otwórz przestrzeń roboczą w notesie Jupyter	35
Dostęp do przestrzeni roboczej w notesie Jupyter	36
Otwieranie terminala w notesie Jupyter	37
Uruchom przykład Pythona	37
Przesyłanie danych do notesu Jupyter	38
Rozdział 5 / Praca w Jupyter Lab	39
Otwieranie przestrzeni roboczej w notatniku Jupyter	39
Dostęp do przestrzeni roboczej w Jupyter Lab	40
Uruchom przykład Pythona	41

Przesyłanie danych do Jupyter Lab	42
Otwieranie terminalu w Jupyter Lab	43
Rozdział 6 / Praca w aplikacji SAS Enterprise Guide	45
Otwórz przestrzeń roboczą w aplikacji SAS Enterprise Guide	45
Dostęp do przestrzeni roboczej w aplikacji SAS Enterprise Guide	46
Rozdział 7 / Praca z procesami SAS-owymi	49
Czym są procesy SAS-owe?	49
Strona procesów w SAS Viya Workbench	50
Tworzenie definicji procesu	50
Przekaż proces	52
Polecenie sas-workbench-execute	53
Rozwiązywanie problemów: Procesy SAS-owe	57
Załącznik 1 / Względy bezpieczeństwa	59
Ładowanie i pobieranie danych	59
Przechowywanie danych uwierzytelnienia użytkownika	59
Załącznik 2 / Wskazówki dotyczące rozwiązywania problemów i często zadawane pytania	61
Jakie są różnice pomiędzy SAS Viya Workbench i platformą SAS Viya?	61
Czy SAS Viya Workbench obsługuje SASPy?	62
Co się stanie, jeśli przestrzeń robocza przekroczy przydzielony limit miejsca?	62
Jak przywrócić połączenie po przekroczeniu limitu czasu w SAS Viya Workbench	63
Rozwiązywanie problemów: Włączanie plików cookie podmiotów zewnętrznych dla SAS Viya Workbench	64
Rozwiązywanie problemów: Możliwy błąd podczas korzystania z pakietów SWAT i SASVIYA	66
Załącznik 3 / Licencje i informacje podmiotów zewnętrznych	67
Wiadomości ogólne	67
Jupyter	68
Jupyter Lab	68
Serwer OpenVSCoDe	69
Matplotlib	70
scikit-learn	70
seaborn	71
Teradata	72

Używanie SAS Viya Workbench

<i>SAS Viya Workbench - Informacje</i>	1
<i>Co to jest SAS Viya Workbench Advanced?</i>	2
<i>Otwórz SAS Viya Workbench</i>	2
<i>Tworzenie przestrzeni roboczej</i>	3
<i>Zrozumienie statusu przestrzeni roboczej</i>	4
<i>Klonowanie repozytorium przykładów</i>	5
<i>Zmiana zasobów obliczeniowych</i>	6
<i>Usuwanie przestrzeni roboczej</i>	7
<i>Przełączanie zintegrowanych środowisk projektowych</i>	8
<i>Zarządzanie przechowywaniem</i>	9
Zrozumienie przechowywania w SAS Viya Workbench	9
Utwórz wolumen	9
Używanie woluminu domowego	10
Zarządzanie woluminami	12
Magazyn NFS	13
Tworzenie zrzutu ekranu	13
Tworzenie woluminu ze zrzutu	14
<i>Zatrzymywanie przestrzeni roboczej</i>	15
<i>Dostęp do dokumentacji SAS-owej</i>	15

SAS Viya Workbench - Informacje

SAS Viya Workbench to aplikacja oparta na chmurze przeznaczona dla programistów i analityków danych. Użytkownicy mogą uzyskać dostęp do SAS Viya Workbench przez przeglądarkę na pulpicie.

W SAS Viya Workbench możesz uruchomić kod SAS-a 9 i analizy platformy SAS Viya w tym samym środowisku, bez konieczności instalowania SAS-a 9 lub platformy SAS Viya. Architektura wielojęzyczna umożliwia natywne uruchamianie

SAS Analytics w innych językach programowania, takich jak Python i R. SAS Viya Workbench jest dla programistów SAS-a, którzy chcą uzyskać dostęp do SAS-a w VS Code oraz dla programistów Pythona lub R, którzy używają VS Code lub Jupyter. Listę procedur SAS obsługiwanych w SAS Viya Workbench znajdziesz w [SAS Viya Workbench Procedures by Name and Capability](#).

W SAS Viya Workbench użytkownicy mogą zarządzać własnym środowiskiem bez dodatkowego wsparcia IT. Użytkownicy mogą także wybrać moc obliczeniową dostosowaną do potrzeb projektu.

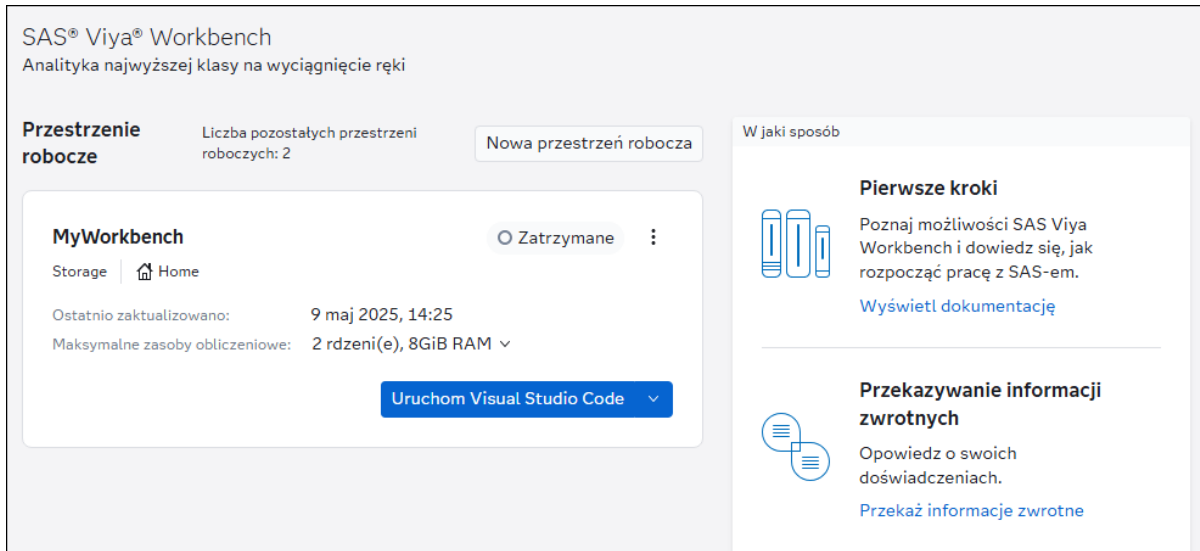
Co to jest SAS Viya Workbench Advanced?

SAS Viya Workbench Advanced pierwotnie zaplanowano na grudzień 2024. Zaktualizowana data wydania wersji i szczegóły zostaną ogłoszone na początku 2026 r. Dalsze informacje na temat planów obsługi procedur w SAS Viya Workbench znajdziesz w ["Workbench Procedures Roadmap" w Workbench Procedures Roadmap](#).

Otwórz SAS Viya Workbench

Aby otworzyć SAS Viya Workbench, sprawdź swoją pocztę e-mail, gdzie uzyskasz instrukcje dotyczące logowania.

Na stronie startowej możesz także tworzyć nowe przestrzenie robocze i zarządzać istniejącymi.



Tworzenie przestrzeni roboczej

- 1 Kliknij **Nowa przestrzeń robocza**.
- 2 Podaj nazwę przestrzeni roboczej.
- 3 Z listy rozwijanej **Maksymalne zasoby obliczeniowe** wybierz liczbę używanych rdzeni. Możesz wybierać spośród 2, 4 lub 16 rdzeni.

Uwaga: Jeśli SAS Viya Workbench administrator włączył obsługę procesora graficznego, dostępna jest również opcja procesora graficznego.

- 4 Z listy rozwijanej **Wolumin główny** wybierz miejsce przechowywania. Aby utworzyć wolumen, patrz ["Utwórz wolumen" na stronie 9](#).

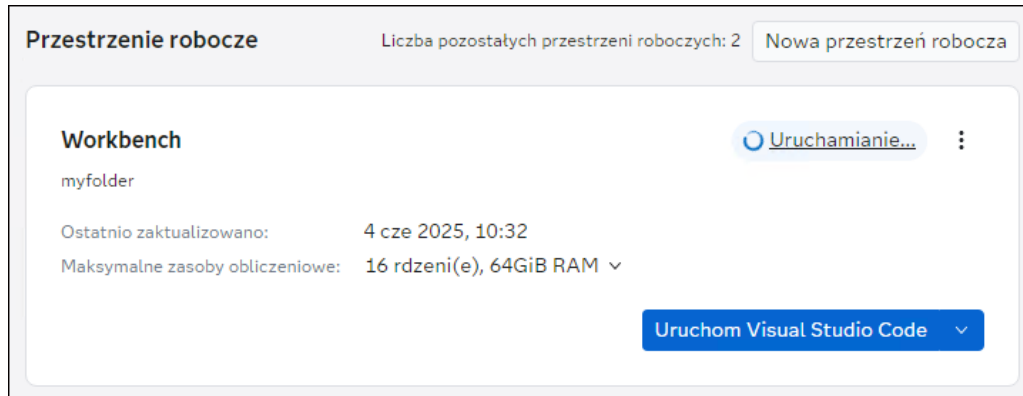
WSKAZÓWKA Domyślny wolumin nazywa się myfolder.

- 5 Wprowadź ścieżkę do folderu montażowego dla przestrzeni roboczej. Folder ten jest automatycznie tworzony w pliku `/workspaces/`.
- 6 Dodaj dowolne dodatkowe woluminy, które chcesz skojarzyć z tym warsztatem. Można także zdefiniować wolumen bazowy.
- 7 Aby użyć zmiennych środowiskowych do zainicjowania sesji warsztatu, kliknij **Nowy**. W Dodaj zmienną środowiskową wpisz nazwę zmiennej środowiskowej i jej wartość. Następnie kliknij **OK**.

Przykłady zmiennych środowiskowych, które można ustawić to `SAS_LOG_CFG_PATH` i `SAS_LOG_OUTPUT_DIR`.

8 Kliknij **OK**.

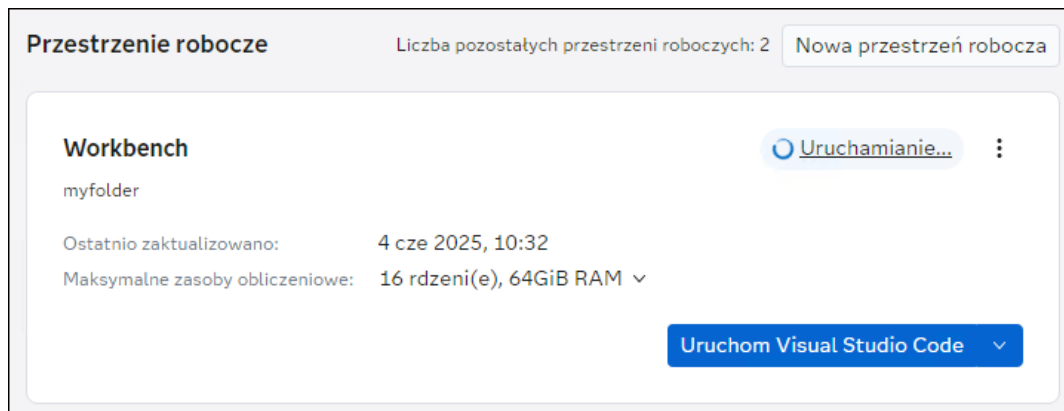
Nowa przestrzeń robocza jest już dostępna na stronie startowej.



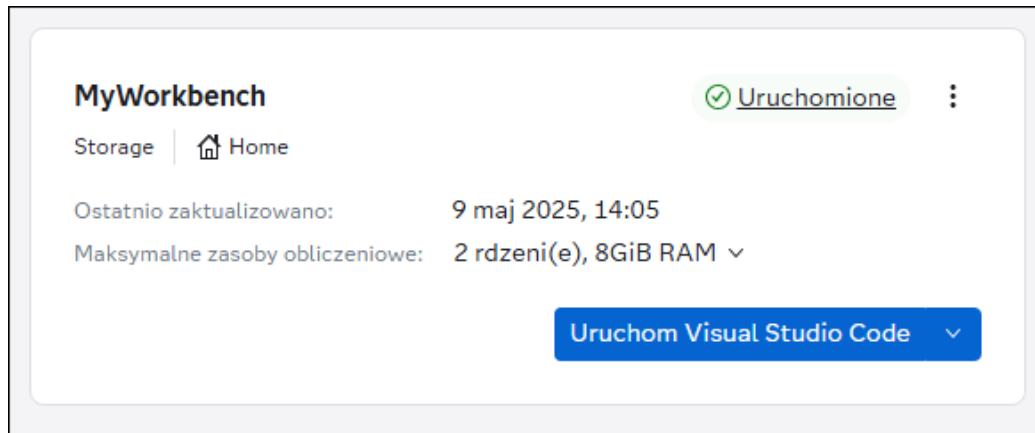
Zrozumienie statusu przestrzeni roboczej

Na stronie startowej SAS Viya Workbench możesz obejrzeć status istniejących przestrzeni roboczych.

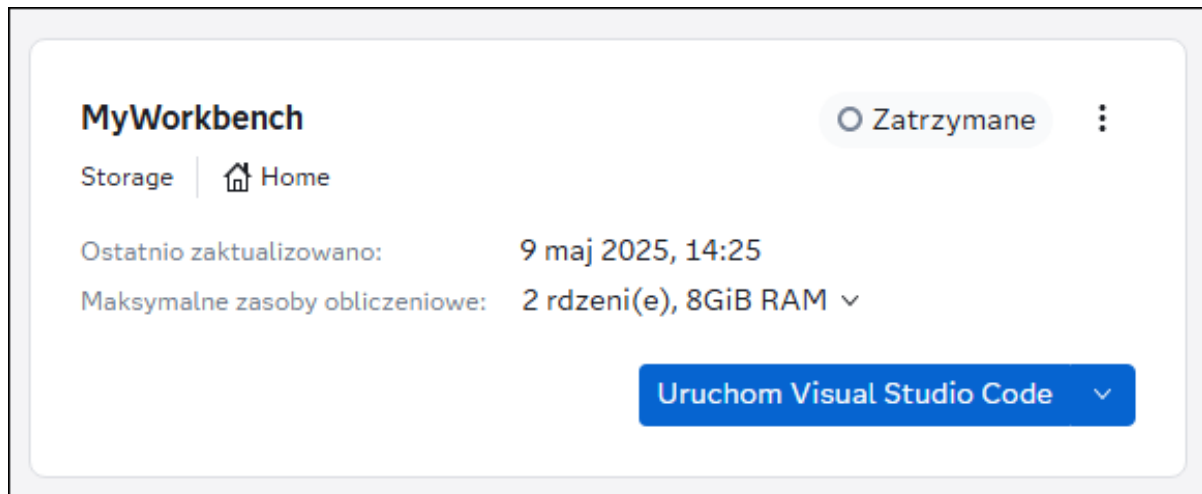
- Uruchamianie — ten status pojawia się po kliknięciu **Start** lub po otwarciu przestrzeni roboczej.



- Uruchomiono — Przestrzeń robocza jest uruchomiona.



- Zatrzymano — przestrzeń robocza została zatrzymana. Aby zatrzymać przestrzeń roboczą, kliknij  i wybierz **Zatrzymaj Przestrzeń roboczą**.



Na stronie startowej możesz także sprawdzić, czy przestrzeń robocza zawiera jakieś błędy. Aby uzyskać więcej informacji na temat błędu, najedź kursorem na ikonę Błąd. W okienku podręcznym pojawi się krótkie podsumowanie błędu. Dalsze informacje znajdziesz klikając **Wyświetl informacje w logu**.

Aby wyświetlić status wszystkich przestrzeni roboczych, użyj logu aktywności.

Klonowanie repozytorium przykładów

Przestrzeń robocza uruchamia się po otwarciu go w zintegrowanym środowisku programistycznym. Domyślnie przestrzeń robocza jest pusta.

W tej dokumentacji założono, że sklonowałeś przykłady SAS Viya Workbench z [GitHuba](#).

Aby sklonować te przykłady i udostępnić je w sesji przestrzeni roboczej:

- 1 Otwórz przestrzeń roboczą w zintegrowanym środowisku programistycznym.

2 Otwórz terminal i wprowadź następujące polecenie:

```
git clone https://github.com/sassoftware/sas-viya-workbench-examples.git
```

Nie musisz być zalogowany do GitHuba, aby skopiować przykłady do swojej przestrzeni roboczej.

Teraz przestrzeń robocza zawiera następujące foldery:

- **data** zawiera źródła danych dla przykładów. Wczytaj plik README.md, aby uzyskać informacje o każdym źródle danych.
- **python** zawiera przykładowe notatniki Pythona. Wczytaj plik README.md, aby wyświetlić notatniki uporządkowane według algorytmu, zbioru danych i biblioteki.
- **sas** zawiera kod SAS-owy przykładów. Wczytaj plik README.md, aby zobaczyć programy uporządkowane według procedury SAS-owej i zbioru SAS-owego.

Zmiana zasobów obliczeniowych

Podczas początkowego tworzenia przestrzeni roboczej, podaje się maksymalną liczbę zasobów obliczeniowych. Liczba ta ustanawia ograniczenia przestrzeni roboczej.

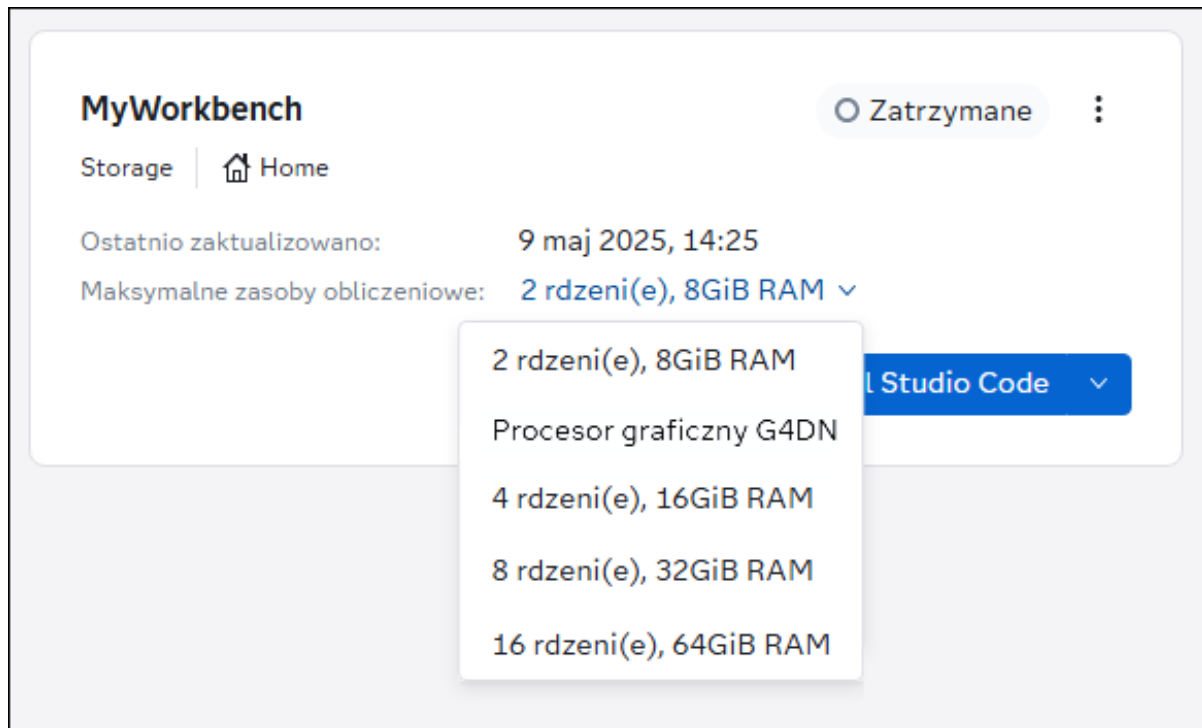
Aby zmienić ilość zasobów obliczeniowych kliknij  obok pola **Maksymalne zasoby obliczeniowe**.

Można wybrać spośród poniższych opcji:

- **2 rdzenie, 8GB RAM**
- **Procesor graficzny G4DN**

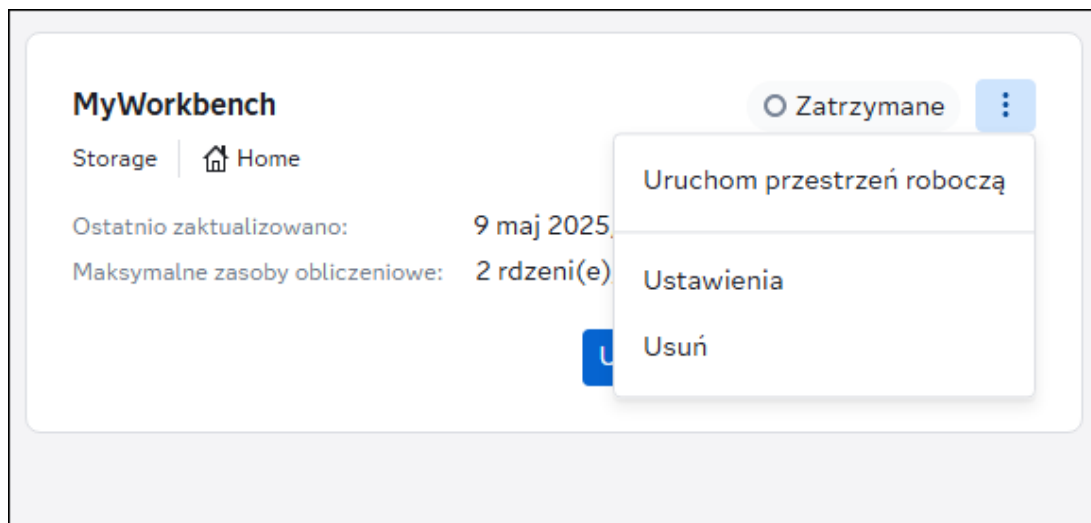
.....
Uwaga: Opcja ta jest dostępna tylko jeśli administrator włączył obsługę procesora graficznego przy otwieraniu SAS Viya Workbench.

- **4 rdzenie, 16GB RAM**
- **8 rdzeni, 32 GB RAM**
- **16 rdzeni, 64 GB RAM**



Usuwanie przestrzeni roboczej

- 1 Na stronie startowej kliknij  dla przestrzeni roboczej, które chcesz usunąć i wybrać **Usuń**.



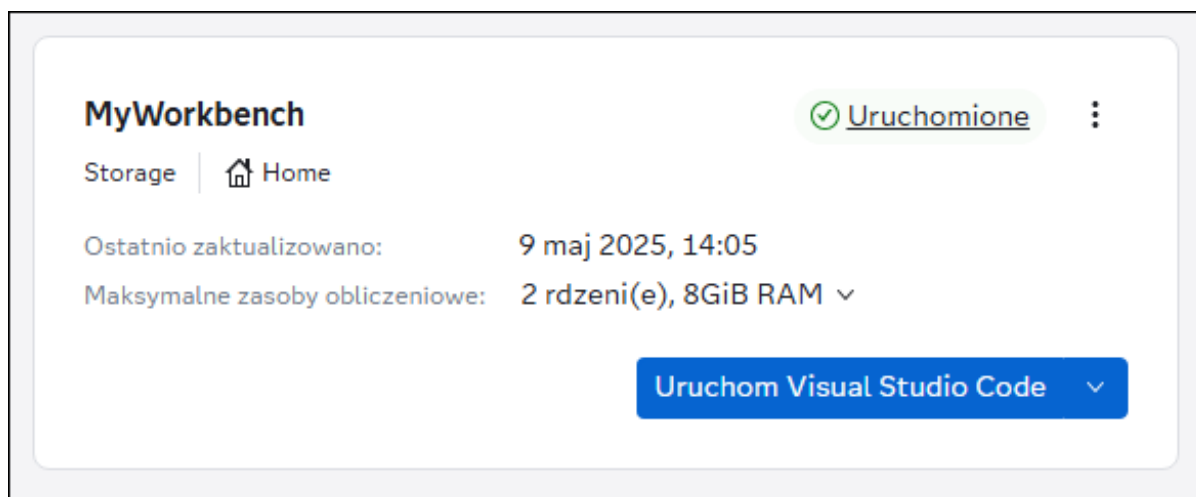
- 2 Po wyświetleniu monitu kliknij **Usuń**.

Przestrzeń robocza nie jest już dostępna na stronie startowej.

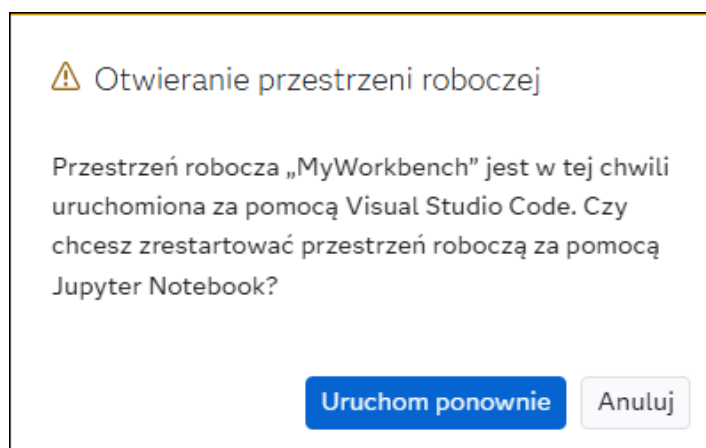
Przełączanie zintegrowanych środowisk projektowych

Przestrzeń robocza może być otwarta tylko w jednym środowisku programistycznym na raz.

Ze strony startowej SAS Viya Workbench możesz łatwo przełączać środowiska programistyczne dla swojej przestrzeni roboczej. W tym przykładzie przestrzeń robocza działa w VS Code.



Aby przełączyć się do innego środowiska programistycznego, kliknij  i wybierz nowe środowisko programistyczne. Jeśli przestrzeń robocza działa obecnie w innym środowisku programistycznym, zostanie wyświetlony monit o jego ponowne uruchomienie, aby można było ją otworzyć w nowym środowisku.



Zarządzanie przechowywaniem

Zrozumienie przechowywania w SAS Viya Workbench

Workbench może być skojarzony z jednym lub więcej woluminami. Posiadanie wielu woluminów skojarzonych ze środowiskiem roboczym umożliwia łatwy dostęp do danych z różnych źródeł.

SAS Viya Workbench umożliwia tworzenie zrzutów obrazu woluminów. Zrzuty ekranu umożliwiają utworzenie kopii zapasowej miejsca przechowywania i szybkie utworzenie nowych woluminów przy użyciu istniejących zrzutów. Zrzuty ekranu umożliwiają wprowadzanie zmian w środowisku roboczym i, jeśli to konieczne, przywracanie poprzedniej wersji środowiska roboczego.

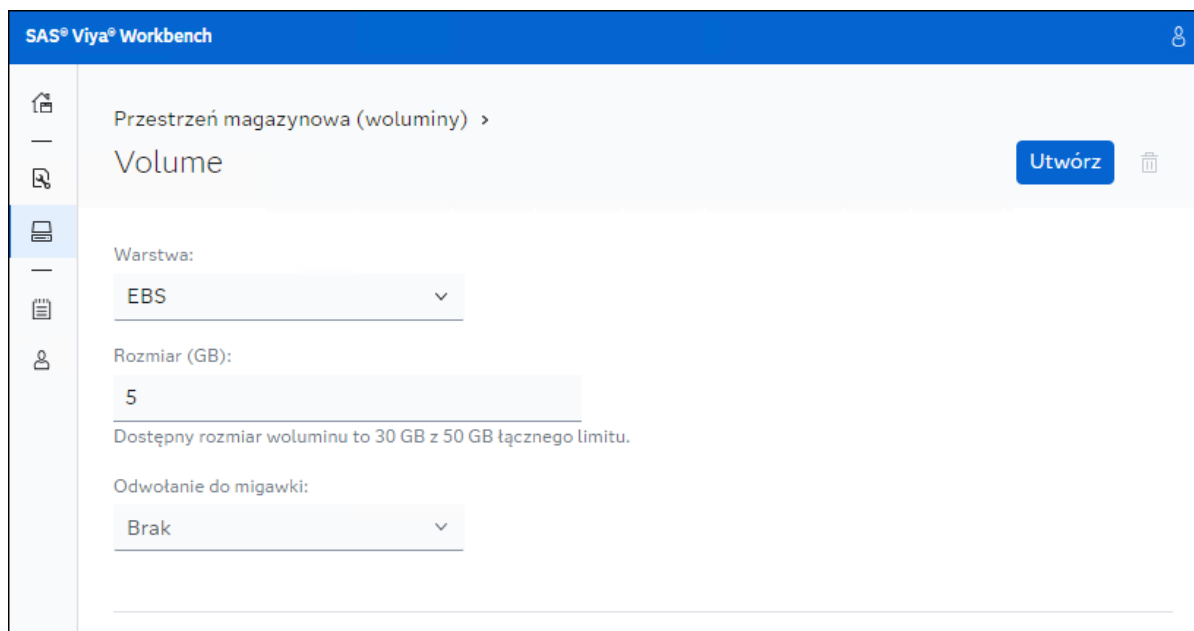
Utwórz wolumen

Kiedy tworzysz warsztat, możesz wybrać dla niego główny wolumin i dowolne dodatkowe woluminy.

Aby wyświetlić listę istniejących woluminów, kliknij  w lewym okienko nawigacji SAS Viya Workbench.

Aby utworzyć wolumen:

- 1 Kliknięcie **Nowy wolumin**.
- 2 Wpisz nazwę woluminu.
- 3 Wybierz warstwę (EBS lub EFS) dla warsztatu.
- 4 Wybierz rozmiar (GB) środowiska roboczego.
- 5 (Opcjonalne) Wybierz odwołanie do zrzutu dla warsztatu. Nowy wolumin użyje ustawień z tego odwołania do zrzutu.
- 6 Kliknij **Utwórz**.



Aby usunąć wolumin z **Objętość** kliknij  i wybierz **Usuń wolumin**.



Uwaga: Kiedy usuwasz wolumin, rzuty ekranu skojarzone z tym woluminem nie są usuwane. Możesz na przykład utworzyć kopię zapasową woluminu, tworząc rzut, a następnie usunąć wolumin. Ponieważ rzut nadal istnieje, możesz utworzyć z niego nowy wolumin.

Używanie woluminu domowego

Korzyści ze stosowania woluminu domowego

W aplikacji SAS Viya Workbench, wolumin domowy umożliwia współdzielenie rozszerzeń i ustawień konfiguracyjnych między platformami roboczymi. Wolumin startowy zawiera również plik autoexec.

Założmy, że masz wolumin domowy i instalujesz rozszerzenie SAS w Visual Studio Code dla warsztatu. Wszelkie ustawienia konfiguracyjne dla tego rozszerzenia są zapisywane w woluminie macierzystym, więc ustawienia te są dostępne dla wszystkich środowisk roboczych korzystających z woluminu macierzystego. Jeśli nie używasz woluminu domowego do zapisywania konfiguracji i ustawień rozszerzenia, musisz podać te ustawienia dla każdego warsztatu, w którym używasz tego rozszerzenia.

Administratorzy aplikacji SAS Viya Workbench mogą przydzielać użytkownikom wolumin domowy. Użytkownicy SAS Viya Workbench mogą uzyskać dostęp do tego woluminu domowego z sesji końcowej w swoim IDE. Zarówno „\$HOME” jak i „\$SAS_USER” wskazują na `/home/sas` do wszystkich przestrzeni roboczych.

Pierwsze przydzielenie woluminu startowego może potrwać kilka minut. Po utworzeniu woluminu startowego wszystkie kolejne stanowiska pracy automatycznie będą z niego korzystać.

Jaka jest różnica między woluminem a woluminem startowym?

Tworząc warsztat, trzeba podać wolumin. W tej lokalizacji zapisywane są informacje o przestrzeni roboczej oraz wszelkie dane, kod i inne pliki.

Wolumin startowy jest opcjonalny. Jednak korzystanie z woluminu startowego oznacza, że możesz współdzielić rozszerzenia i ustawienia konfiguracyjne między przestrzeniami roboczymi.

Wolumin startowy zawiera informacje dla wielu przestrzeni roboczych, a zatem w woluminie startowym może znajdować się wiele miejsc przechowywania. Przykładowo masz dwie przestrzenie robocze, Workbench_A i Workbench_B. Przestrzeń robocza_A używa `myfolder` jako swojego woluminu. Przestrzeń robocza_B używa `sales` jako swojego woluminu. Zarówno `myfolder` jak i `sales` znajdują się w woluminie startowym.

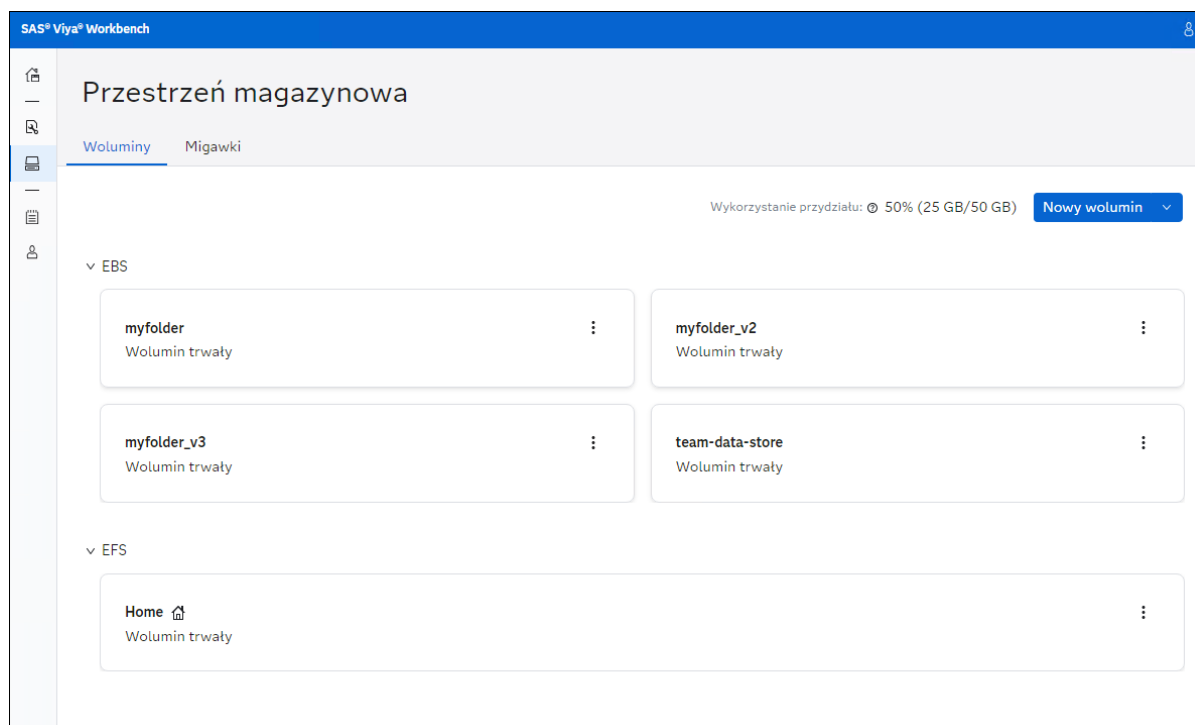
Utwórz wolumin startowy

- 1 W SAS Viya Workbench kliknij .
- 2 Na stronie **Przechowywanie**, wybierz **Nowy wolumin** ⇒ **Nowy wolumin startowy**.

Uwaga: Możesz mieć tylko jeden wolumin startowy. Jeśli wolumin startowy już istnieje, zostanie usunięty po utworzeniu nowego. Wszystkie konfiguracje i ustawienia w istniejącym woluminie startowym zostaną usunięte.

- 3 Podaj rozmiar woluminu startowego.
- 4 Kliknij **Utwórz**.

Wolumin startowy pojawi się na karcie **Woluminy**.



W tym przykładzie myfolder myfolder_v2, myfolder_v3 oraz team-data-store są woluminami przestrzeni roboczej. Woluminem startowym jest Start.

WAŻNE Przestrzenie robocze utworzone po utworzeniu woluminu startowego automatycznie z niego korzystają. Przestrzenie robocze utworzone przed utworzeniem woluminu startowego nie korzystają z niego. Musisz usunąć i ponownie utworzyć wszystkie istniejące przestrzenie robocze, aby mogły one korzystać z woluminu startowego.

Jak uzyskać dostęp do woluminu startowego

Wolumin startowy znajduje się w `/home/sas`. Możesz przeglądać wolumin startowy, korzystając z sesji terminalowej w swoim IDE.

Zarządzanie woluminami

Użytkownicy Workbenchu mogą tworzyć woluminy, kiedy użytkownik tworzy przestrzeń roboczą.

Aby wyświetlić listę woluminów w Twojej instalacji, kliknij  i kliknij kartę **Woluminy**.

Woluminy mogą być definiowane przez użytkownika końcowego lub administratora. Wolumin startowy (nazywany też katalogiem startowym) nosi nazwę Start. Dalsze informacje znajdziesz w [“Jaka jest różnica między woluminem a woluminem startowym?” na stronie 11](#).

Na stronie **Woluminy** można wykonać następujące zadania:

- edytować ustawienia woluminu. Kliknij  a potem **Otwórz**, aby zmodyfikować nazwę miejsca przechowywania lub rozmiar. Rozmiar to 5 GB. Można zwiększyć rozmiar woluminu, ale nie można go zmniejszyć. Można także dodać lub usunąć odwołanie do zrzutu.
- usunąć wolumin. Kliknij  a następnie **Usuń wolumin**.
- utworzyć zrzut.

Magazyn NFS

Można umieścić National File Share (dane) bezpośrednio w SAS Viya Workbench.

- 1 W Microsoft Marketplace lub w konsoli AWS zapisz adres IP SAS Viya Workbench bramki NAT płaszczyzny danych.
- 2 W swojej subskrypcji Azure lub AWS dodaj adres IP do listy dozwolonych serwerów NFS.
- 3 Wyślij te informacje email'em do zespołu SAS Viya Workbench Ops (workbenchops@sas.com):
 - nazwa hosta / adres IP serwera NFS
 - nazwę udziału NFS
 - podkatalog SAS Viya Workbench

Domyślnie użytkownicy mają przypisany identyfikator GUID 001. Administrator może używać panelu administratora w SAS Viya Workbench do pominięcia identyfikatora GUID dla określonego użytkownika.

Tworzenie zrzutu ekranu

Aby utworzyć zrzut istniejącego woluminu:

- 1 Kliknij  i kliknij kartę **Woluminy**.

- 2 Kliknij  a następnie **Nowy zrzut ekranu**.

Zrzut pojawi się na karcie **Zrzuty ekranu**.

Przestrzeń magazynowa		
Woluminy <u>Migawki</u>		
 Filtruj	 	 Usuń
☐ Nazwa	Data utworzenia	
☐ myfolder_v1	5 cze 2025, 10:28 	
☐ myfolder_v2	5 cze 2025, 10:28 	
☐ myfolder_v3	5 cze 2025, 10:28 	
☐ myfolder_v4	5 cze 2025, 10:28 	

Aby wyświetlić zrzuty ekranu skojarzone z określonym woluminem:

- 1 Otwórz kartę **Wolumeny**.
- 2 Przy woluminie, który chcesz wyświetlić, kliknij  a potem **Otwórz**.

Przestrzeń magazynowa (woluminy) > myfolder  Zapisz 	
Warstwa: EBS Rozmiaru woluminu nie można zmniejszyć, lecz można zwiększyć. Rozmiar (GB): 5 Dostępny rozmiar woluminu to 30 GB z 50 GB łącznego limitu. Utworzył: pupusr3@sas-test-pupreactdev.com pupusr3@sas-test-pupreactdev.com Data utworzenia: 30 maj 2025, 15:07	Migawki + Nowy 
Nazwa	Data utworzenia
myfolder_v6	5 cze 2025, 10:52 
myfolder_v5	5 cze 2025, 10:52 
myfolder_v4	5 cze 2025, 10:28 
myfolder_v3	5 cze 2025, 10:28 
myfolder_v2	5 cze 2025, 10:28 
myfolder_v1	5 cze 2025, 10:28 

Tworzenie woluminu ze zrzutu

- 1 Kliknij  i kliknij kartę **Zrzuty ekranu**.
- 2 Kliknij nazwę zrzutu.
- 3 Na stronie zrzutu *name-of-snapshot*, kliknij **Nowy wolumin ze zrzutu**.

Nowy wolumin jest tworzony przy użyciu ustawień z wybranego zrzutu.

Zatrzymywanie przestrzeni roboczej

Aby zatrzymać przestrzeń roboczą, kliknij  i wybierz **Zatrzymaj przestrzeń roboczą** na stronie startowej SAS Viya Workbench.

.....
Uwaga: Zamknięcie przeglądarki lub środowiska programistycznego (Jupyter lub VS Code) nie powoduje zatrzymania przestrzeni roboczej. Przestrzeń robocza nadal działa. Możesz sprawdzić status swojej przestrzeni roboczej na stronie głównej SAS Viya Workbench. Dalsze informacje znajdziesz w ["Zrozumienie statusu przestrzeni roboczej" na stronie 4](#).
.....

Dostęp do dokumentacji SAS-owej

Aby uzyskać dostęp do dokumentacji SAS-owej, kliknij **Wyświetl dokumentację** na stronie startowej SAS Viya Workbench.

Jeśli korzystasz ze środowiska programistycznego (VS Code lub Jupyter), możesz uzyskać dostęp do dokumentacji z pliku README.md.

Praca z Gitem i SSH

<i>Używanie Gita z poziomu Workbenchu</i>	17
<i>Klonowanie publicznego repozytorium GitHub bez instalacji</i>	17
<i>Korzystanie z SSH w Workbenchu</i>	18
<i>Konfigurowanie globalnych ustawień Gita</i>	19
<i>Komunikat: Błąd krytyczny: Wykryto wątpliwą własność</i>	19
<i>Wysyłanie plików do zdalnego repozytorium Gita</i>	20

Używanie Gita z poziomu Workbenchu

Narzędzie wiersza poleceń `git` jest wstępnie zainstalowane w Twojej przestrzeni roboczej. Aby utworzyć lokalną kopię repozytorium, wprowadzić zmiany i przesłać je do oryginalnego repozytorium, należy wykonać kilka czynności konfiguracyjnych. Te etapy konfiguracji mogą się różnić w zależności od hosta Gita. Na przykład, aby przekazać zmiany do repozytorium GitHub, trzeba najpierw mieć konto GitHub. Jednak aby po prostu skopiować repozytorium do przestrzeni roboczej, nie musisz wykonywać żadnych czynności konfiguracyjnych. Informacje jak pominąć proces konfiguracji i skopiować repozytorium przykładów Workbenchu znajdziesz w ["Klonowanie publicznego repozytorium GitHub bez instalacji"](#) na stronie 17.

Klonowanie publicznego repozytorium GitHub bez instalacji

Klonowanie repozytorium polega na utworzeniu lokalnej kopii wszystkich plików i powiązanych metadanych w repozytorium. Możesz sklonować repozytorium, uruchamiając następujące polecenie:

```
git clone https://path/to/repository.git
```

Na przykład poniższe polecenie klonuje zestaw przykładów, których możesz użyć w swojej przestrzeni roboczej:

```
git clone https://github.com/sassoftware/sas-viya-workbench-examples.git
```

Nie musisz być zalogowany do GitHuba, aby skopiować przykłady do swojej przestrzeni roboczej.

Korzystanie z SSH w Workbenchu

Secure Shell (SSH) to protokół komunikacyjny wykorzystujący szyfrowanie kluczem publicznym w celu ustanowienia bezpiecznego kanału komunikacji pomiędzy klientem i serwerem. Konfiguruje protokół SSH w swojej przestrzeni roboczej, tworzysz tożsamość, która może służyć do uwierzytelniania na serwerze. Możesz skonfigurować dane uwierzytelniające SSH w swojej przestrzeni roboczej w taki sam sposób, jak w dowolnym innym systemie operacyjnym typu Unix. Poniższe kroki pokazują jeden ze sposobów konfiguracji protokołu SSH. Dokładne kroki konfiguracji różnią się w zależności od zasad bezpieczeństwa obowiązujących w Twojej witrynie, ale poniższe kroki przedstawiają typową strukturę generowania klucza SSH dla środowiska Workbench:

- 1 Otwórz terminal dla wybranego edytora.
- 2 Uruchom `ssh-keygen`. Odpowiedz twierdząco na monit, aby zaakceptować domyślną lokalizację pliku dla tożsamości lub podaj alternatywną ścieżkę. Odpowiedz na monit, aby zdefiniować hasło, aby używać tej tożsamości, lub naciśnij Enter, aby nie podawać hasła.
- 3 (Opcjonalnie) Skopiuj swój klucz publiczny na serwer, który chcesz uwierzytelnić.

Etap generowania klucza może się różnić w zależności od polityki bezpieczeństwa Twojej witryny. Jedną z powszechnych odmian jest wybór innego schematu szyfrowania, np. ED25519, który można określić za pomocą `ssh-keygen -t ed25519` na etapie generowania klucza.

Lokalizacja pliku jest istotnym czynnikiem w przypadku przestrzeni roboczych. Domyślnie plik tworzony jest w formacie `~/.ssh`. Folder `~/.ssh` odpowiada katalogowi głównemu, dzięki czemu jest dostępny we wszystkich przestrzeniach roboczych i wystarczy wykonać konfigurację tylko raz.

Istnieje wiele sposobów skopiowania klucza publicznego na serwer. W przypadku aplikacji internetowych, np. GitHub, często można znaleźć interfejs umożliwiający kopiowanie i wklejanie zawartości pliku klucza publicznego, np. postępując zgodnie z instrukcjami na tej stronie: [Dodanie nowego klucza SSH do konta GitHub](#).

Alternatywnie, aby uwierzytelnić się jako użytkownik na innym komputerze typu Unix, możesz skopiować tożsamość za pomocą następującego polecenia: `ssh-copy-id -i /home/sas/.ssh/id_ed25519 user@host`. Opcja `host` odpowiada

komputerowi hosta, z którym chcesz się połączyć. Opcja `user` odpowiada użytkownikowi, jako który chcesz się zalogować na komputerze-goście.

Uwaga: Przed utworzeniem tożsamości SSH zapoznaj się z polityką bezpieczeństwa swojej witryny.

Konfigurowanie globalnych ustawień Gita

Aby zatwierdzenia były ważne, Git wymaga podania nazwy użytkownika i adresu e-mail. Możesz ustawić swoją nazwę użytkownika i adres e-mail, uruchamiając następujące dwa polecenia:

```
git config --global user.email „ty@przykład.com”
git config --global user.name „Imię i nazwisko”
```

Po pomyślnym wykonaniu polecenia na terminalu nie zostaje wypisany żaden wynik. Możesz potwierdzić konfigurację, uruchamiając `git config --global --list` i sprawdzając, czy `user.email` i `user.name` to wartości, których się spodziewasz.

Gdy podasz opcję `--global` podczas ustawiania wartości konfiguracji, ustawienia te będą miały zastosowanie do wszystkich poleceń Git, chyba że zostaną nadpisane lokalną konfiguracją. Twoje globalne konfiguracje są zapisywane w pliku `.gitconfig` w Twoim katalogu głównym. Ponieważ jest to część Twojego katalogu głównego, ustawienia te zostaną zachowane dla wszystkich instancji Workbench.

Komunikat: Błąd krytyczny: Wykryto wątpliwą własność”

Niektóre polecenia Gita powodują wyświetlenie następującego komunikatu:

```
błąd krytyczny: wykryto wątpliwą własność w repozytorium w path
Aby dodać wyjątek dla tego katalogu:
git config --global --add bezpieczny.katalog ścieżka
```

Rozwiązaniem jest przesłanie polecenia zaproponowanego w wiadomości. Konkretnie, dodaj repozytorium jako bezpieczną ścieżkę, uruchamiając następujące polecenie:

```
git config --global --add safe.directory ścieżka
```

Komunikat ten jest konsekwencją sposobu, w jaki Workbench zarządza własnością plików. Git pojawia się w tym przypadku, gdy użytkownik przesyłający polecenia Git jest innym użytkownikiem niż użytkownik będący właścicielem plików w repozytorium Git. Domyślnie użytkownikiem, który przesyła polecenia Git, jest użytkownik **sas**. Pliki są jednak nominalnie własnością użytkownika proxy wskazanego przez UID. Pliki są własnością użytkownika proxy, nawet jeśli użytkownik **SAS-a** uruchomił polecenia w celu utworzenia plików.

Wysyłanie plików do zdalnego repozytorium Gita

Gdy masz lokalne repozytorium Gita, możesz śledzić zmiany w plikach i przesyłać zmiany do zdalnego repozytorium, takiego jak GitHub.

- 1 Zmiany w plikach w repozytorium.
- 2 Użyj `git add nazwa_pliku`, aby dodać plik do obszaru danych pośrednich. Powtórz dla wszystkich plików, które chcesz dodać.
- 3 Użyj `git commit` aby zatwierdzić wszystkie pliki z repozytorium lokalnego.
- 4 Użyj `git push <remote> <branch>`, aby wysłać zmiany z gałęzi lokalnej do zdalnej.

Jednak zanim będzie można wysłać zmiany, muszą być spełnione następujące warunki:

- Musi istnieć repozytorium zdalne.
- Musisz być upoważniony do wysyłania do repozytorium zdalnego.
- Lokalne repozytorium musi wiedzieć, jak uzyskać dostęp do zdalnego repozytorium.

Pierwsze dwa elementy różnią się w zależności od tego, kto udostępnia zdalne repozytoria Gita. Na przykład w przypadku GitHub można użyć internetowego interfejsu użytkownika do utworzenia i rozdzielenia repozytoriów. Aby uwierzytelnić się w GitHubie, musisz mieć konto i sposób uwierzytelnienia z wiersza poleceń do swojego konta, na przykład poprzez skonfigurowanie tokenów SSH lub GitHuba. Proces różni się dla różnych hostów Gita. Informacje na temat tych elementów znajdziesz w [Dokumentacji GitHuba](#), u administratora Gita lub w [podręczniku Gita](#).

Aby upewnić się, że lokalne repozytorium wie o zdalnym repozytorium, możesz potrzebować wykonać dodatkowe kroki w celu dodania tego zdalnego. Często zdalne repozytorium jest definiowane automatycznie podczas klonowania repozytorium. Zgodnie z konwencją zdalne repozytorium nosi nazwę "origin" i używa protokołu oraz adresu URL użytych do sklonowania repozytorium. Jeśli jednak nie

chcesz wysyłać wiadomości do repozytorium zdalnego, które sklonowałeś, powinieneś zmodyfikować definicje zdalne. Możesz dodawać, usuwać lub pobierać informacje o repozytoriach zdalnych za pomocą poleceń `git remote`. Następujące polecenia `git remote` służą do zarządzania repozytoriami zdalnymi:

- `git remote --verbose` tworzy listę nazw i adresów URL wszystkich repozytoriów zdalnych. Każde z nich jest często wyświetlany na liście dwukrotnie, raz ze względu na miejsce pobierania zmian i drugi raz ze względu na miejsce, do którego je wysyła. Zdalne funkcje pobierania i wysyłania często używają tego samego adresu URL.
- `git remote add nameURL` dodaje repozytrium zdalne do listy znanych repozytoriów zdalnych. Używa protokołu i nazwy podanych w części adresu URL.
- `git remote remove name` usuwa repozytorium zdalne z listy znanych repozytoriów zdalnych.

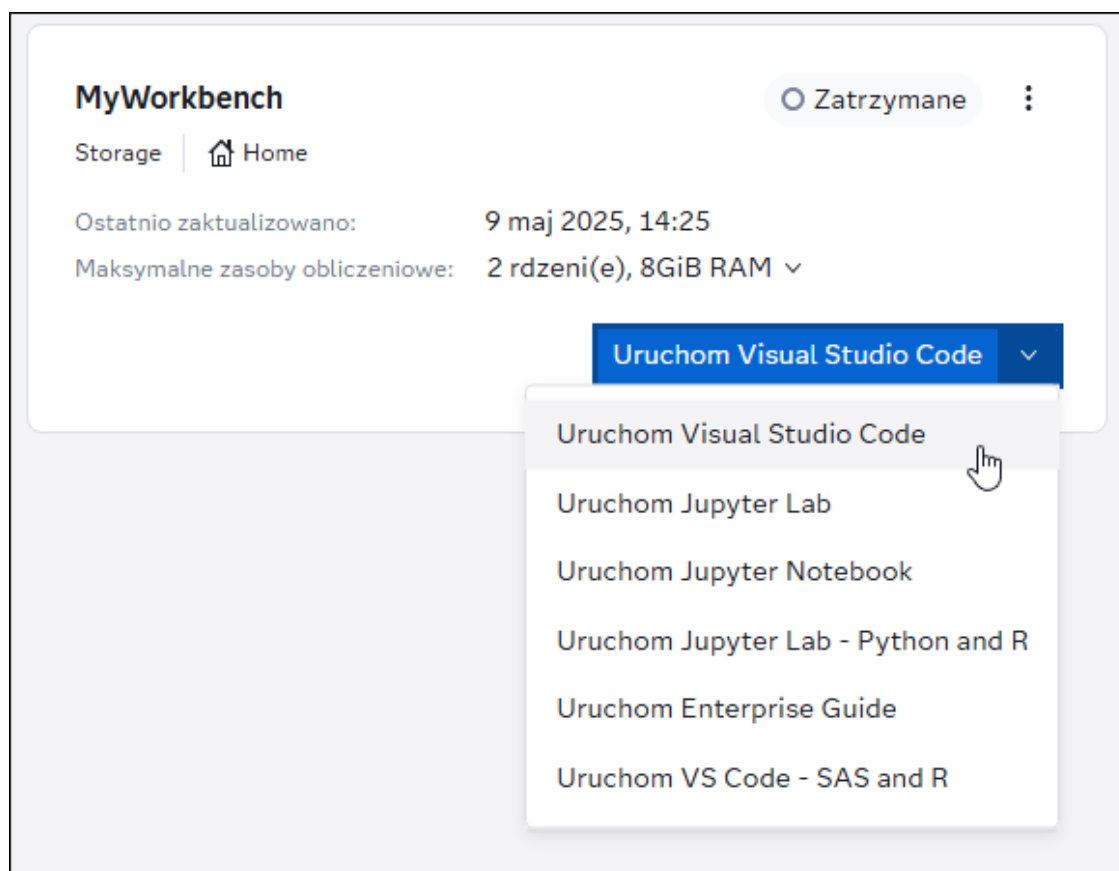
Dalsze informacje na temat używania polecenia `git remote` znajdziesz w [dokumentacji zdalnego repozytorium Gita](#).

Praca w kodzie Visual Studio

<i>Otwórz przestrzeń roboczą w programie Visual Studio Code</i>	23
<i>Dostęp do SAS Viya Workbench w kodzie Visual Studio</i>	24
<i>Uruchom przykład Pythona</i>	26
<i>Uruchom przykładowy program SAS-owy</i>	27
<i>Dostęp do bibliotek SAS-owych</i>	30
<i>Otwieranie terminala w kodzie Visual Studio</i>	31
<i>Wybór schematu kolorów SAS-owych</i>	32
<i>Utwórz notatnik SAS-owy</i>	32
<i>Korzystanie z funkcji AutoExec w kodzie Visual Studio</i>	34

Otwórz przestrzeń roboczą w programie Visual Studio Code

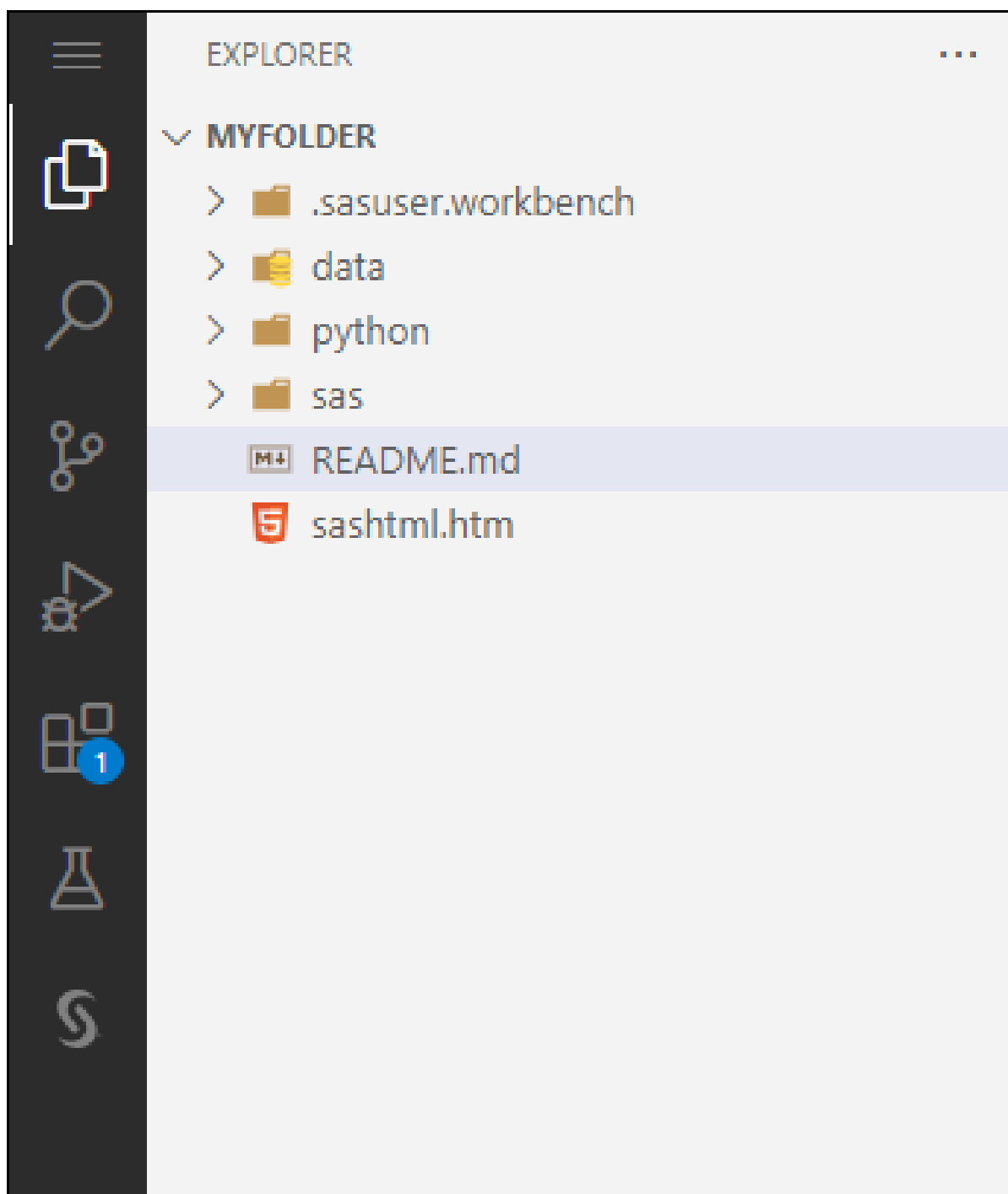
Można otworzyć przestrzeń roboczą ze strony startowej. Z listy rozwijanej wybierz **Uruchom kod Visual Studio**.



Uwaga: W kodzie Visual Studio obecnie obsługiwany jest tylko język angielski.

Dostęp do SAS Viya Workbench w kodzie Visual Studio

Po otwarciu przestrzeni roboczej w element wizualny Studio Code, Eksplorator wyświetla folder zawierający przestrzeń roboczą. Miejscem przechowywania podanym podczas tworzenia przestrzeni roboczej jest MYFOLDER.



Po sklonowaniu repozytorium przykładów, należy przejrzeć następujące foldery:

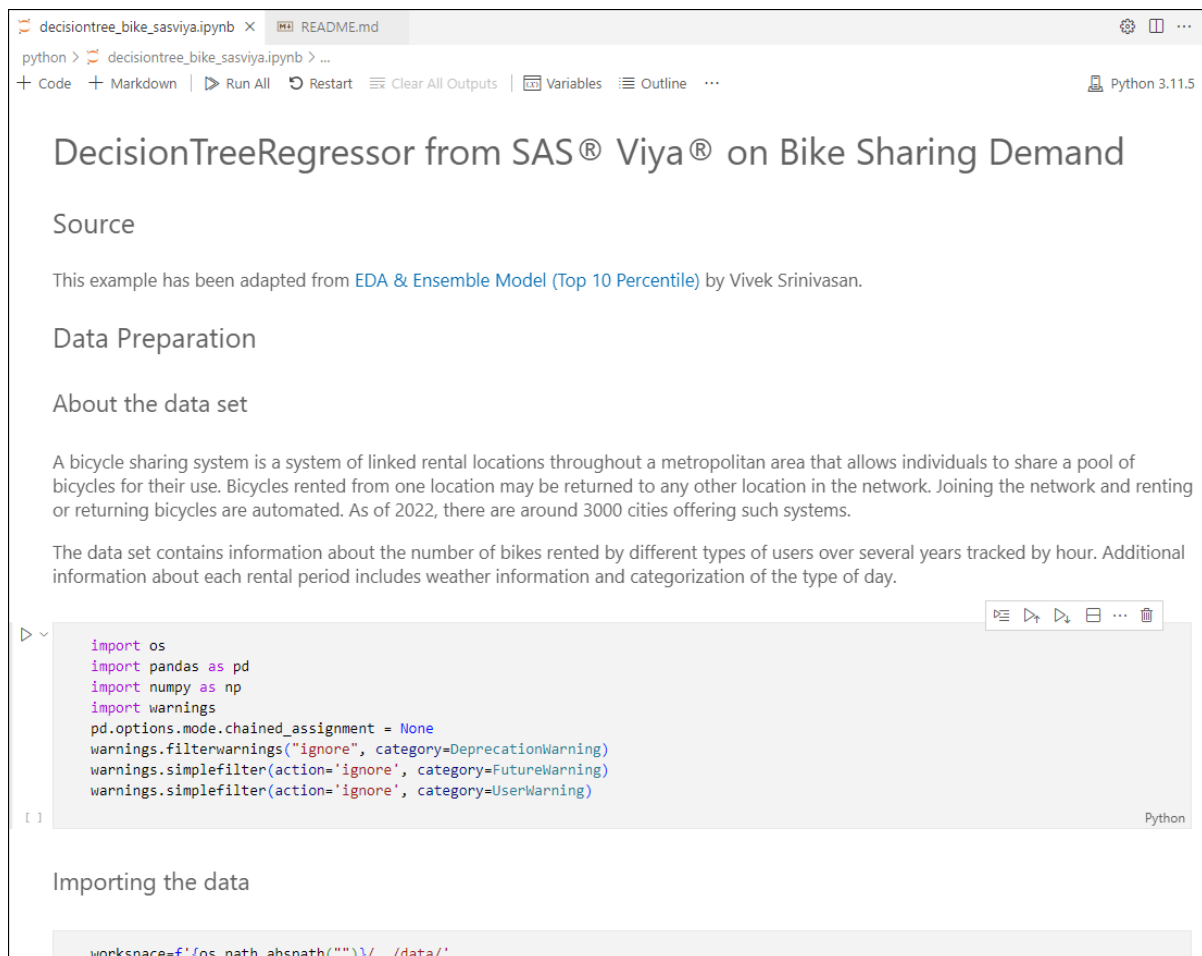
- **data** zawiera źródła danych dla przykładów. Wczytaj plik README.md, aby uzyskać informacje o każdym źródle danych.
- **python** zawiera przykładowe notatniki Pythona. Wczytaj plik README.md, aby wyświetlić notatniki uporządkowane według algorytmu, zbioru danych i biblioteki.
- **sas** zawiera kod SAS-owy przykładów. Wczytaj plik README.md, aby zobaczyć programy uporządkowane według procedury SAS-owej i zbioru SAS-owego.

Dalsze informacje znajdziesz w “Klonowanie repozytorium przykładów” na stronie 5.

Uruchom przykład Pythona

SAS Viya Workbench zapewnia dostęp do kilku przykładowych notatników. Aby wyświetlić przykład, rozwiń folder `python` i dwukrotnie kliknij plik, który chcesz otworzyć. W tym przykładzie kliknij dwukrotnie `decisiontree_bike_sasviya.ipynb`.

Cały przykład zostanie otwarty w obszarze roboczym VS Code.



```
import os
import pandas as pd
import numpy as np
import warnings
pd.options.mode.chained_assignment = None
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.simplefilter(action='ignore', category=FutureWarning)
warnings.simplefilter(action='ignore', category=UserWarning)
```

```
workspace=f'{os.path.abspath("")}/../data/'
```

Aby uruchomić cały przykład, kliknij **Uruchom wszystko**. Dane wynikowe każdego bloku kodu pojawiają się po pojedynczej komórce.

Sample of the first few rows

```
bikeData.head(5)
```

[5] Python

	date	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	cc
0	1/1/2011	1	0	1	0	0	6	0	1	0.24	0.2879	0.81	0.0	3	13	
1	1/1/2011	1	0	1	1	0	6	0	1	0.22	0.2727	0.80	0.0	8	32	
2	1/1/2011	1	0	1	2	0	6	0	1	0.22	0.2727	0.80	0.0	5	27	
3	1/1/2011	1	0	1	3	0	6	0	1	0.24	0.2879	0.75	0.0	3	10	
4	1/1/2011	1	0	1	4	0	6	0	1	0.24	0.2879	0.75	0.0	0	1	

Variable data types

```
bikeData.dtypes
```

[4] Python

Variable	dtype
date	object
season	int64
year	int64
month	int64
hour	int64
holiday	int64
weekday	int64
workingday	int64
weather	int64
temp	float64
atemp	float64

Aby wykonać pojedynczą komórkę, kliknij strzałkę po lewej stronie komórki. Możesz także użyć ikon po prawej stronie, aby uruchomić kod linijka po linijce, wykonać poprzednie i kolejne komórki oraz uzyskać dodatkowe opcje.

Sample of the first few rows

```
bikeData.head(5)
```

[5] Python

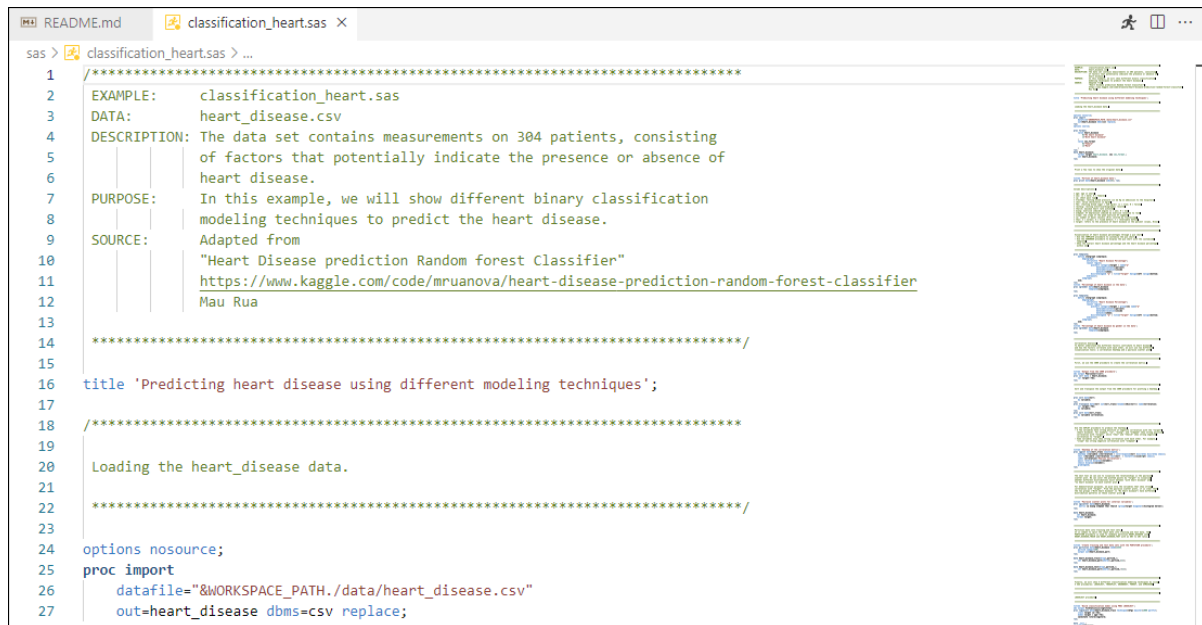
	date	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	cc
0	1/1/2011	1	0	1	0	0	6	0	1	0.24	0.2879	0.81	0.0	3	13	
1	1/1/2011	1	0	1	1	0	6	0	1	0.22	0.2727	0.80	0.0	8	32	
2	1/1/2011	1	0	1	2	0	6	0	1	0.22	0.2727	0.80	0.0	5	27	
3	1/1/2011	1	0	1	3	0	6	0	1	0.24	0.2879	0.75	0.0	3	10	
4	1/1/2011	1	0	1	4	0	6	0	1	0.24	0.2879	0.75	0.0	0	1	

Uruchom przykładowy program SAS-owy

SAS Viya Workbench zapewnia dostęp do kilku przykładowych programów SAS-owych. Aby wyświetlić program SAS-owy, rozwiń folder **sas** i kliknij dwukrotnie

plik, który chcesz otworzyć. W tym przykładzie kliknij dwukrotnie **classification_heart.sas**.

Cały przykład zostanie otwarty w obszarze roboczym VS Code.



```

1  /*****
2  EXAMPLE:  classification_heart.sas
3  DATA:    heart_disease.csv
4  DESCRIPTION: The data set contains measurements on 304 patients, consisting
5              of factors that potentially indicate the presence or absence of
6              heart disease.
7  PURPOSE:   In this example, we will show different binary classification
8              modeling techniques to predict the heart disease.
9  SOURCE:    Adapted from
10             "Heart Disease prediction Random forest Classifier"
11             https://www.kaggle.com/code/mruanova/heart-disease-prediction-random-forest-classifier
12             Mau Rua
13
14  *****/
15
16  title 'Predicting heart disease using different modeling techniques';
17
18  /*****
19
20  Loading the heart_disease data.
21
22  *****/
23
24  options nosource;
25  proc import
26      datafile="&WORKSPACE_PATH./data/heart_disease.csv"
27      out=heart_disease dbms=csv replace;
  
```

Aby uruchomić cały program, kliknij . Aby uruchomić blok kodu, wybierz kod, który chcesz uruchomić i kliknij .

Wszelkie dane wynikowe pojawiają się na karcie **Rezultaty**. Log SAS-owy jest dostępny w okienku **Wynik**.

The screenshot displays the SAS Studio environment with the following components:

- Editor:** Contains the SAS program `classification_heart.sas`. The code includes a title, a description of the data, and a `proc import` statement to load `heart_disease.csv` into a SAS dataset.
- Output Window:** Shows the results of the SAS execution, including a table of the first five observations of the `heart_disease` dataset.
- Log Window:** Displays the SAS log, providing details on the execution time and the number of observations read.

SAS Program Code:

```

1 /*****
2 EXAMPLE:   classification_heart.sas
3 DATA:     heart_disease.csv
4 DESCRIPTION: The data set contains me
5             of factors that potentia
6             heart disease.
7 PURPOSE:   In this example, we will
8             modeling techniques to p
9 SOURCE:     Adapted from
10            "Heart Disease predictio
11            https://www.kaggle.com/c
12            Mau Rua
13 *****/
14
15
16 title 'Predicting heart disease using
17
18 /*****
19
20 Loading the heart_disease data.
21
22 *****/
23
24 options nosource;
25 proc import
26     datafile="&WORKSPACE_PATH./data/he
27     out=heart_disease dbms=csv replace

```

Output Window:

Predicting heart disease using different modeling techniques
Portion of heart_disease data

Obs.	target	sex	age	trestbps	chol	thalch	oldpeak
1	No heart disease	Male	63	145	233	150	
2	With heart disease	Male	67	160	286	108	
3	With heart disease	Male	67	120	229	129	
4	No heart disease	Male	37	130	250	187	
5	No heart disease	Female	41	130	204	172	

Log Window:

Predicting heart disease using different modeling techniques
Percentage of heart disease in the data

NOTE: DATA statement used (Total process time):

NOTE: PROCEDURE SGPLOT used (Total process time):

real time 0.71 seconds

17 SAS System

poniedziałek, 24 czerwca 2024 06:29:00

cpu time 0.09 seconds

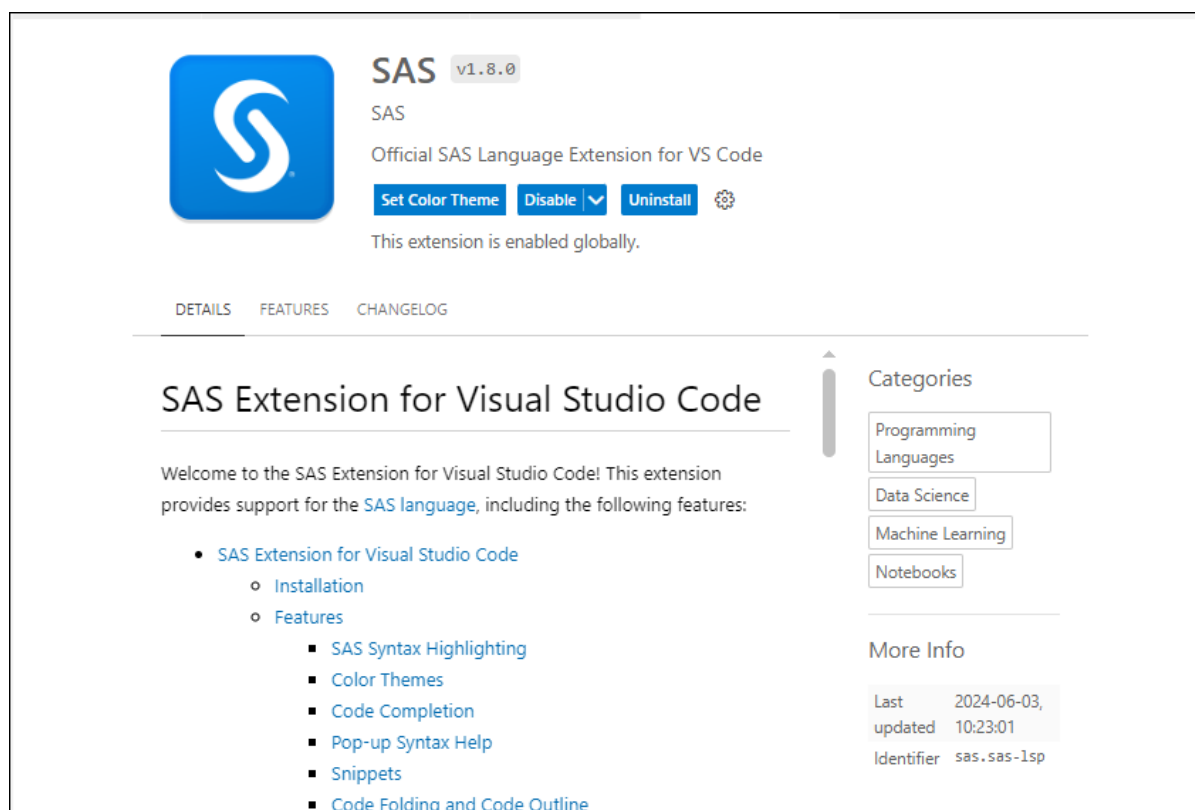
NOTE: There were 10 observations read from the data set WORK.ALLMETHODS.

542

543 title;

544 ;*';*';*/;run;quit;ods html5 close;

Dalsze informacje na temat pracy z kodem SAS-owym w VS Code znajdziesz w dokumentacji rozszerzenia SAS-owego dla kodu Visual Studio.

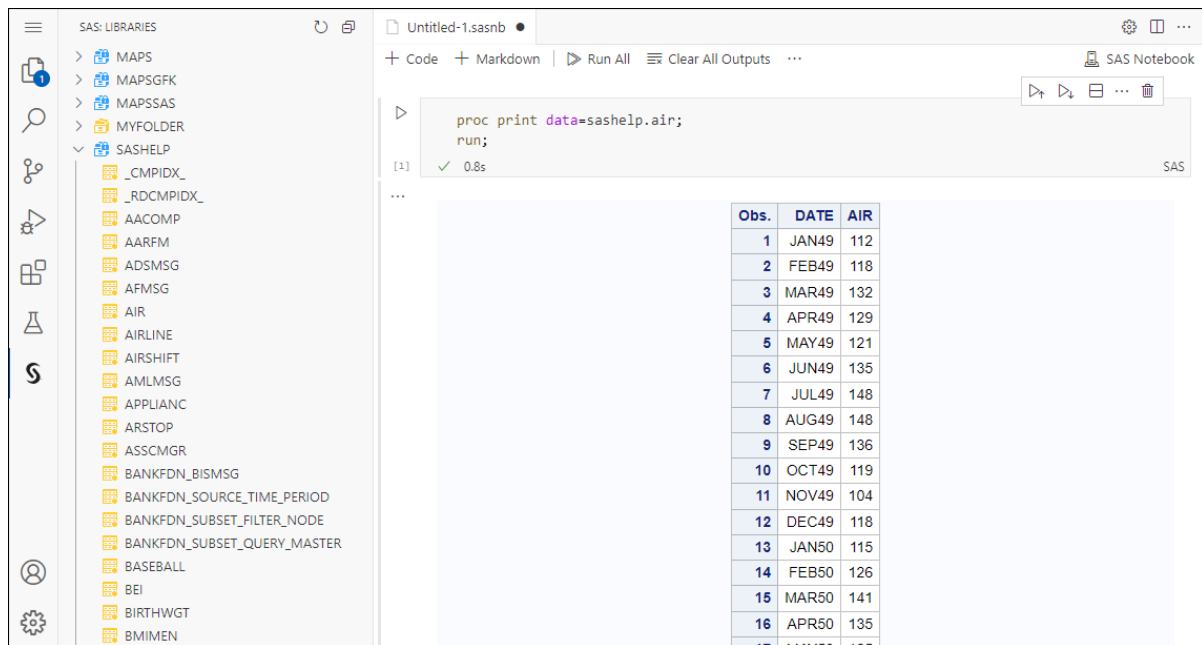


Dostęp do bibliotek SAS-owych

Następujące biblioteki SAS-owe są dostępne poprzez SAS Viya Workbench:

- MAPS
- MAPSGFK
- MAPSSAS
- SASHELP
- SASUSER
- WORK

Aby uzyskać dostęp do bibliotek SAS-owych, kliknij .



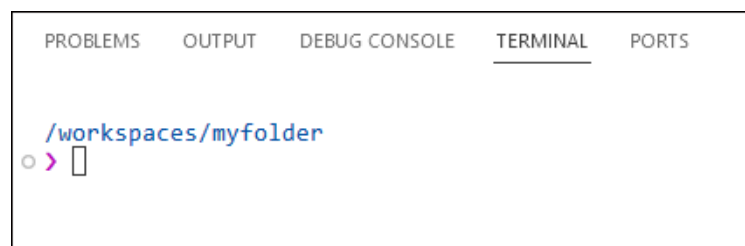
Rozwiń bibliotekę SAS-ową, aby zobaczyć zbiory danych, do których możesz odwołać się z poziomu programów SAS-owych lub notatnika Pythona.

Otwieranie terminala w kodzie Visual Studio

SAS Viya Workbench zawiera pełne środowisko programistyczne, w tym możliwość wywoływania poleceń powłoki.

Aby otworzyć terminal w kodzie Visual Studio, kliknij  i wybierz **Terminal** ⇒ **Nowy terminal**.

Domyślnie terminal otwiera powłokę Bash w katalogu wskazanym jako folder montażowy podczas tworzenia przestrzeni roboczej.



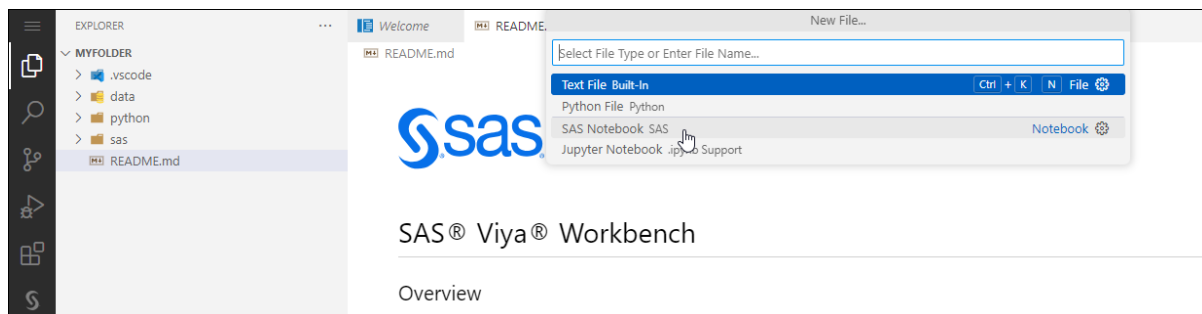
Wybór schematu kolorów SAS-owych

SAS opracował trzy schematy: SAS Light, SAS Dark i SAS High Contrast, które są zalecane ze względu na ułatwienia dostępu dla osób niepełnosprawnych.

- 1 Kliknij  i wybierz **Plik** ⇒ **Preferencje** ⇒ **Schemat** ⇒ **Schemat kolorów**
- 2 Wybierz SAS-owy schemat kolorystyczny.

Utwórz notatnik SAS-owy

Aby utworzyć notatnik SAS-owy, kliknij  i wybierz **Plik** ⇒ **Nowy plik**. Jako typ pliku wybierz **Notatnik SAS-owy**.



Po uruchomieniu sekcji kodu SAS-owego, wynik pojawi się bezpośrednio pod blokiem kodu.

```

output;
procname = "&currentProc";
type = "test";
accuracy = &&acc_test_&currentProc;
output;
%end;
run;
proc sgplot data=allMethods;
vbar procname / response=accuracy group=type nostatlabel datalabel
groupdisplay=cluster dataskin=preserved;
xaxis display=(nolabel);
yaxis grid;
run;
%mend;

title2 'Compare accuracy across all 5 procedures';
%CreateComparison;

title;

```

Predicting heart disease using different modeling techniques
Portion of heart_disease data

Obs.	target	sex	age	trestbps	chol	thalch	oldpeak	ca	cp	exang	slope	thal	restecg	fbs
1	No heart disease	Male	63	145	233	150	2.3	0	3	0	0	1	0	1
2	With heart disease	Male	67	160	286	108	1.5	3	0	1	1	2	0	0
3	With heart disease	Male	67	120	229	129	2.6	2	0	1	1	0	0	0
4	No heart disease	Male	37	130	250	187	3.5	0	2	0	0	2	1	0
5	No heart disease	Female	41	130	204	172	1.4	0	1	0	2	2	0	0

Predicting heart disease using different modeling techniques
Percentage of heart disease in the data

Heart Disease Percentage

Aby wyświetlić log SAS-owy, kliknij i wybierz **Zmień prezentację**.

Z listy rozwijanej wybierz **application/vnd.sas.compute.log.lines** Wizualizator logu SAS-owego.

Select mimetype to render for current output

- application/vnd.sas.compute.log.lines Currently Active
- SAS Log Renderer (SAS.sas-lsp)
- application/vnd.sas.ods.html5
- SAS HTML Renderer (SAS.sas-lsp)

```

length procname $16. type $8.;
%do i = 1 %to %sysfunc(countw(&allprocs));
%let currentProc = %scan(&allprocs,&i);
procname = "&currentProc";

```

Teraz log SAS-owy jest wyświetlany pod każdym blokiem kodu.

```

        accuracy = &&acc_test_&currentProc;
        output;
    %end;
run;
proc sgplot data=allMethods;
    vbar procname / response=accuracy group=type nostatlabel datalabel
                    groupdisplay=cluster dataskin=priest;
    xaxis display=(nolabel);
    yaxis grid;
run;
%mend;

title2 'Compare accuracy across all 5 procedures';
%CreateComparison;

title;
[1] ✓ 39.8s
SAS
... 545 title;footnote;ods _all_ close;
546 ods graphics on;
547 ods html5 style=HTMLEncore options(bitmap_mode='inline' svg_mode='inline');
NOTE: Writing HTML5 Body file: sashtml1.htm
548 %let _SASPROGRAMFILE = %nrquote(%nrstr(Untitled-1.sasnb));
549 /*****
550 EXAMPLE:      classification_heart.sas
551 DATA:        heart_disease.csv
552 DESCRIPTION:  The data set contains measurements on 304 patients, consisting
553                of factors that potentially indicate the presence or absence of
554                heart disease.
555 PURPOSE:      In this example, we will show different binary classification
556                modeling techniques to predict the heart disease.
557 SOURCE:       Adapted from
558                "Heart Disease prediction Random forest Classifier"
559                https://www.kaggle.com/code/muannova/heart-disease-prediction-random-forest-classifier

```

Korzystanie z funkcji AutoExec w kodzie Visual Studio

Za pomocą rozszerzenia SAS-owego w kodzie Visual Studio, możesz skonfigurować plik autoexec, który będzie uruchamiany po rozpoczęciu sesji. Wszelkie zmiany wprowadzone w pliku autoexec wymagają zamknięcia i ponownego otwarcia sesji, aby zaczęły obowiązywać.

Dalsze informacje znajdziesz w [SAS Autoexec Settings](#) w *SAS Extension for Visual Studio Code Documentation*.

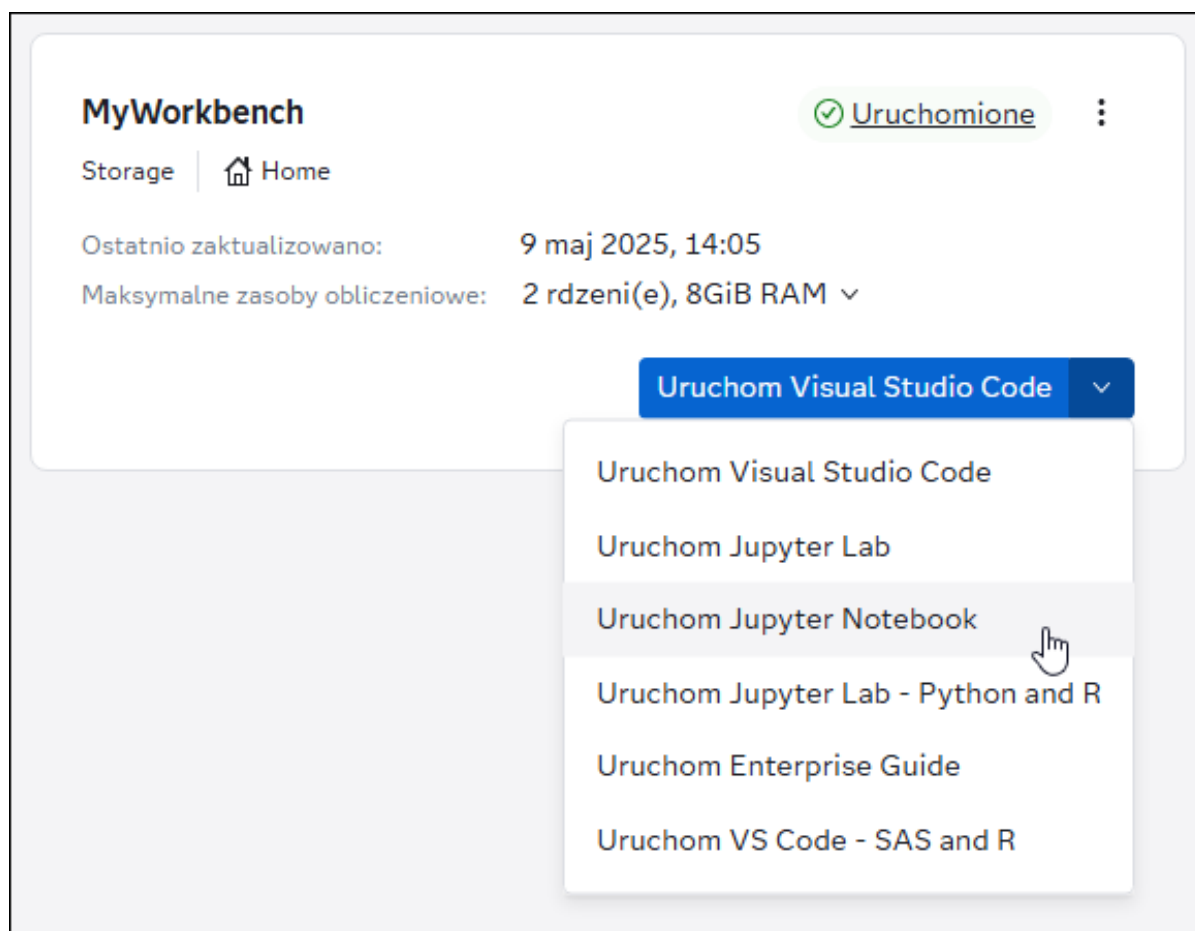
Zapisz plik autoexec w katalogu domowym (`/home/sas`), aby udostępnić ustawienia w wielu przestrzeniach roboczych.

Praca w notesie Jupyter

<i>Otwórz przestrzeń roboczą w notesie Jupyter</i>	35
<i>Dostęp do przestrzeni roboczej w notesie Jupyter</i>	36
<i>Otwieranie terminalu w notesie Jupyter</i>	37
<i>Uruchom przykład Pythona</i>	37
<i>Przesyłanie danych do notesu Jupyter</i>	38

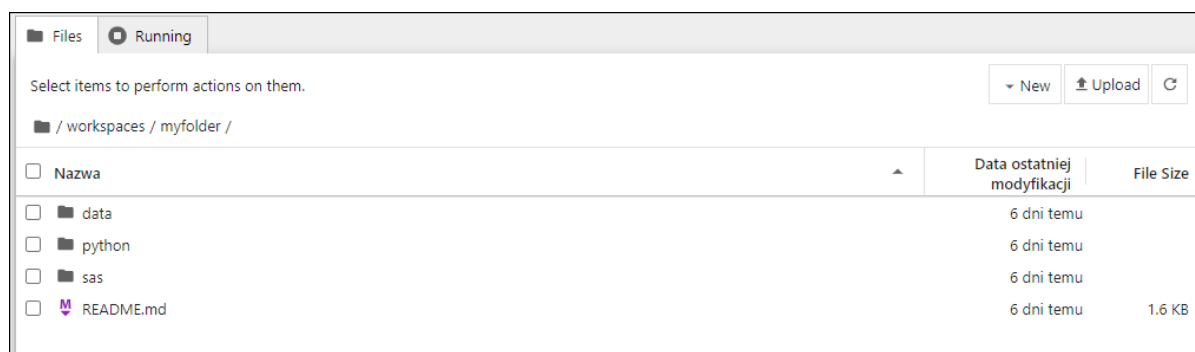
Otwórz przestrzeń roboczą w notesie Jupyter

Można otworzyć przestrzeń roboczą ze strony startowej. Z listy rozwijalnej wybierz **Uruchom notatnik Jupyter**.



Dostęp do przestrzeni roboczej w notesie Jupyter

Po otwarciu przestrzeni roboczej w notesie Jupyter karta **Pliki** wyświetla folder zawierający przestrzeń roboczą. Miejsce przechowywania, **myfolder** jest tym, co określiłeś podczas tworzenia przestrzeni roboczej.



Po sklonowaniu repozytorium przykładów, należy przejrzeć następujące foldery:

- **data** zawiera źródła danych dla przykładów. Wczytaj plik README.md, aby uzyskać informacje o każdym źródle danych.
- **python** zawiera przykładowe notatniki Pythona. Wczytaj plik README.md, aby wyświetlić notatniki uporządkowane według algorytmu, zbioru danych i biblioteki.
- **sas** zawiera kod SAS-owy przykładów. Wczytaj plik README.md, aby zobaczyć programy uporządkowane według procedury SAS-owej i zbioru SAS-owego.

Dalsze informacje znajdziesz w [“Klonowanie repozytorium przykładów” na stronie 5](#).

Otwieranie terminalu w notesie Jupyter

SAS Viya Workbench zawiera pełne środowisko programistyczne, w tym możliwość wywoływania poleceń powłoki.

Aby otworzyć terminal, wybierz **Nowy** ⇒ **Terminal**.

Domyślnie terminal otwiera powłokę Bash w katalogu wskazanym jako folder montażowy podczas tworzenia przestrzeni roboczej.



Uruchom przykład Pythona

SAS Viya Workbench zapewnia dostęp do kilku przykładowych notatników. Aby wyświetlić przykład dwukrotnie kliknij folder **python** i dwukrotnie kliknij plik, który chcesz otworzyć.

Aby uruchomić cały przykład, wybierz **Uruchom** ⇒ **Uruchom wszystkie komórki**.

Aby uruchomić zaznaczony tekst, użyj opcji w menu **Uruchom**.

Dane wynikowe bloku kodu pojawiają się pod tą komórką.

Importing the data

```
[2]: workspace=f'{os.path.abspath("")}/../data/'
bikeData = pd.read_csv(workspace + "bike_sharing_demand.csv")
bikeData.shape
```

[2]: (17379, 16)

Sample of the first few rows

```
[3]: bikeData.head(5)
```

	date	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	count
0	1/1/2011	1	0	1	0	0	6	0	1	0.24	0.2879	0.81	0.0	3	13	16
1	1/1/2011	1	0	1	1	0	6	0	1	0.22	0.2727	0.80	0.0	8	32	40
2	1/1/2011	1	0	1	2	0	6	0	1	0.22	0.2727	0.80	0.0	5	27	32
3	1/1/2011	1	0	1	3	0	6	0	1	0.24	0.2879	0.75	0.0	3	10	13
4	1/1/2011	1	0	1	4	0	6	0	1	0.24	0.2879	0.75	0.0	0	1	1

Variable data types

```
[4]: bikeData.dtypes
```

```
[4]: date          object
season          int64
year            int64
month           int64
hour            int64
holiday          int64
weekday          int64
workingday       int64
weather          int64
temp            float64
atemp           float64
humidity         float64
windspeed       float64
casual           int64
registered       int64
```

Przesyłanie danych do notesu Jupyter

Na stronie głównej przestrzeni roboczej w notesie Jupyter kliknij **Załaduj**. Wybierz źródło danych zapisane na Twoim komputerze lokalnym i kliknij **Otwórz**. Źródło danych pojawi się teraz jako element na liście.

The screenshot shows the Jupyter Notebook file browser interface. At the top, there are tabs for 'Files' and 'Running'. Below the tabs, there is a text prompt 'Select items to perform actions on them.' and buttons for 'New', 'Upload', and a refresh icon. The main area displays a list of files and folders in the workspace, with columns for 'Nazwa' (Name), 'Data ostatniej modyfikacji' (Last modified date), and 'File Size'.

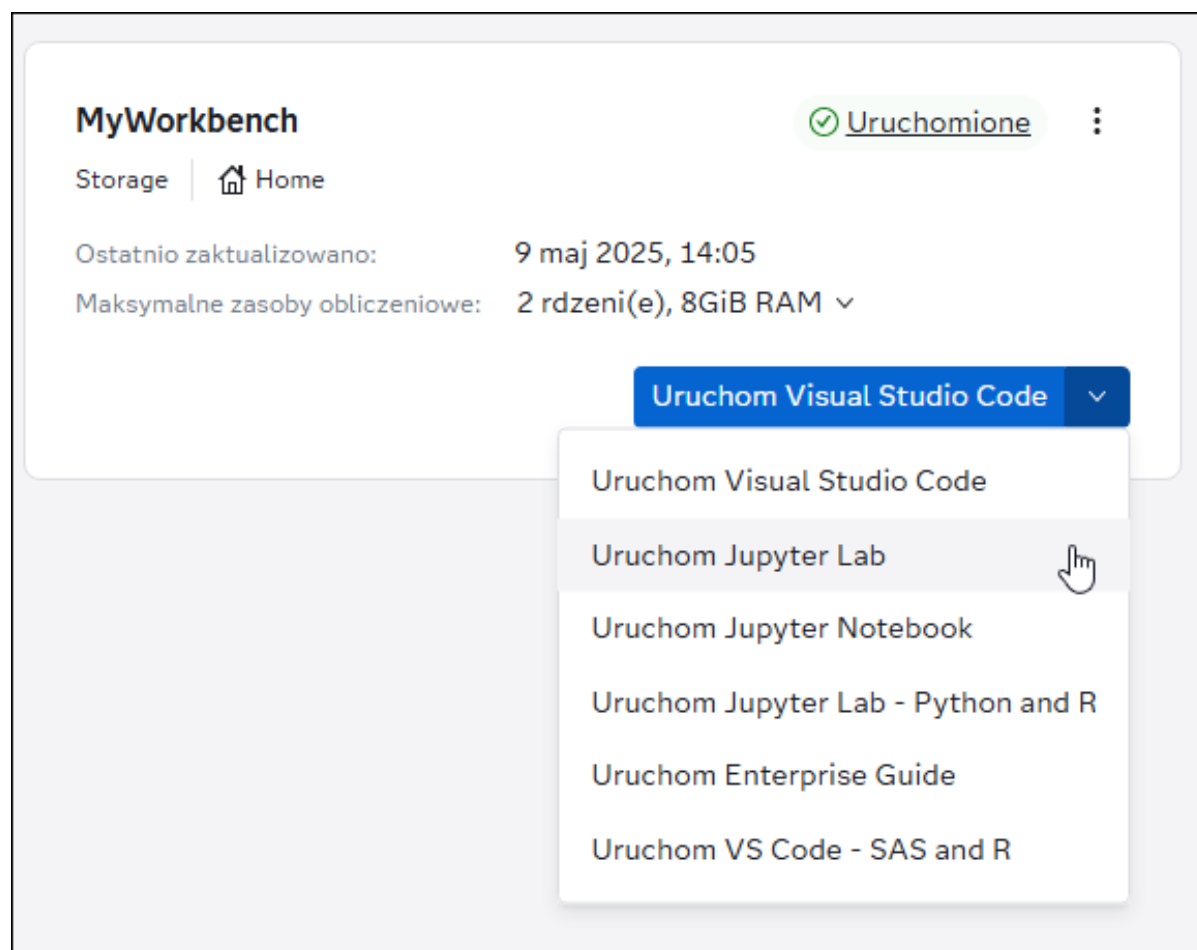
Nazwa	Data ostatniej modyfikacji	File Size
data	6 dni temu	
python	6 dni temu	
sas	6 dni temu	
banking.csv	2 godziny temu	6 KB
README.md	6 dni temu	1.6 KB

Praca w Jupyter Lab

<i>Otwieranie przestrzeni roboczej w notatniku Jupyter</i>	39
<i>Dostęp do przestrzeni roboczej w Jupyter Lab</i>	40
<i>Uruchom przykład Pythona</i>	41
<i>Przesyłanie danych do Jupyter Lab</i>	42
<i>Otwieranie terminalu w Jupyter Lab</i>	43

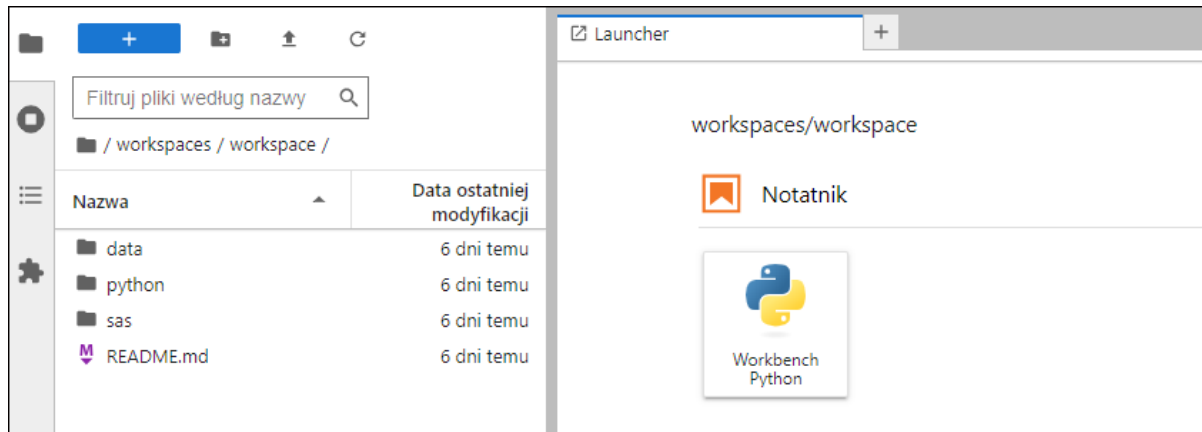
Otwieranie przestrzeni roboczej w notatniku Jupyter

Można otworzyć przestrzeń roboczą ze strony startowej. Z listy rozwijanej wybierz **Uruchom Jupyter Lab**.



Dostęp do przestrzeni roboczej w Jupyter Lab

Kiedy otwierasz przestrzeń roboczą w Jupyter Lab, karta **Przeglądarka plików** wyświetla zawartość tej przestrzeni roboczej.



Po sklonowaniu repozytorium przykładów, należy przejrzeć następujące foldery:

- **data** zawiera źródła danych dla przykładów. Wczytaj plik README.md, aby uzyskać informacje o każdym źródle danych.
- **python** zawiera przykładowe notatniki Pythona. Wczytaj plik README.md, aby wyświetlić notatniki uporządkowane według algorytmu, zbioru danych i biblioteki.
- **sas** zawiera kod SAS-owy przykładów. Wczytaj plik README.md, aby zobaczyć programy uporządkowane według procedury SAS-owej i zbioru SAS-owego.

Dalsze informacje znajdziesz w ["Klonowanie repozytorium przykładów"](#) na stronie 5.

Uruchom przykład Pythona

Przykładowe repozytorium SAS Viya Workbench zapewnia dostęp do kilku przykładowych notatników. Aby wyświetlić przykład, dwukrotnie kliknij folder **python** i dwukrotnie kliknij plik, który chcesz otworzyć.

Aby uruchomić cały przykład, wybierz **Uruchom** ⇨ **Uruchom wszystkie komórki**. Skorzystaj z opcji w menu **Uruchom** do uruchamiania poszczególnych komórek.

Dane wynikowe pojawiają się bezpośrednio pod blokiem kodu.

The screenshot shows a Jupyter Lab notebook titled 'decisiontree_bike_sasviya.ipyn'. The interface includes a toolbar with icons for saving, adding files, and running code. The notebook content consists of three code cells:

- Cell [2]: `bikeData.shape` returns `(17379, 16)`.
- Cell [3]: `bikeData.head(5)` displays a sample of the first five rows of the dataset. The output is a table with 14 columns: `date`, `season`, `year`, `month`, `hour`, `holiday`, `weekday`, `workingday`, `weather`, `temp`, `atemp`, `humidity`, and `windspeed`.
- Cell [4]: `bikeData.dtypes` displays the data types for each variable in the dataset.

Sample of the first few rows

	date	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed
0	1/1/2011	1	0	1	0	0	6	0	1	0.24	0.2879	0.81	
1	1/1/2011	1	0	1	1	0	6	0	1	0.22	0.2727	0.80	
2	1/1/2011	1	0	1	2	0	6	0	1	0.22	0.2727	0.80	
3	1/1/2011	1	0	1	3	0	6	0	1	0.24	0.2879	0.75	
4	1/1/2011	1	0	1	4	0	6	0	1	0.24	0.2879	0.75	

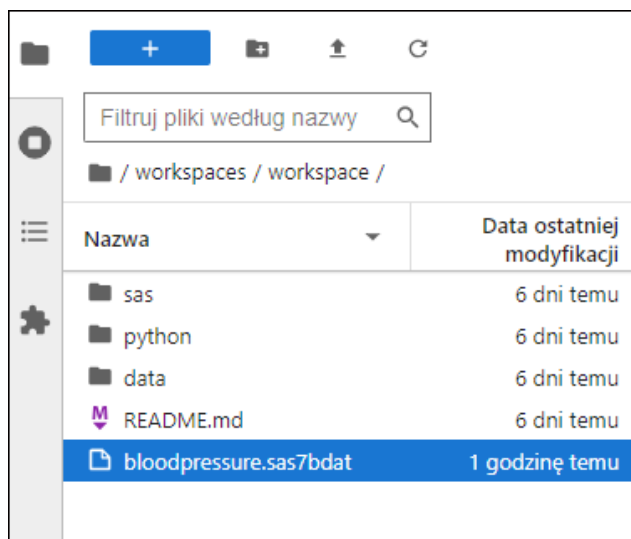
Variable data types

```

[4]: date          object
     season        int64
     year          int64
     month         int64
     hour          int64
     holiday       int64
     weekday       int64
     workingday    int64
     weather       int64
     temp          float64
     atemp         float64
     humidity      float64
     windspeed     float64
     casual        int64
     registered    int64
     count         int64
     dtype: object
  
```

Przesyłanie danych do Jupyter Lab

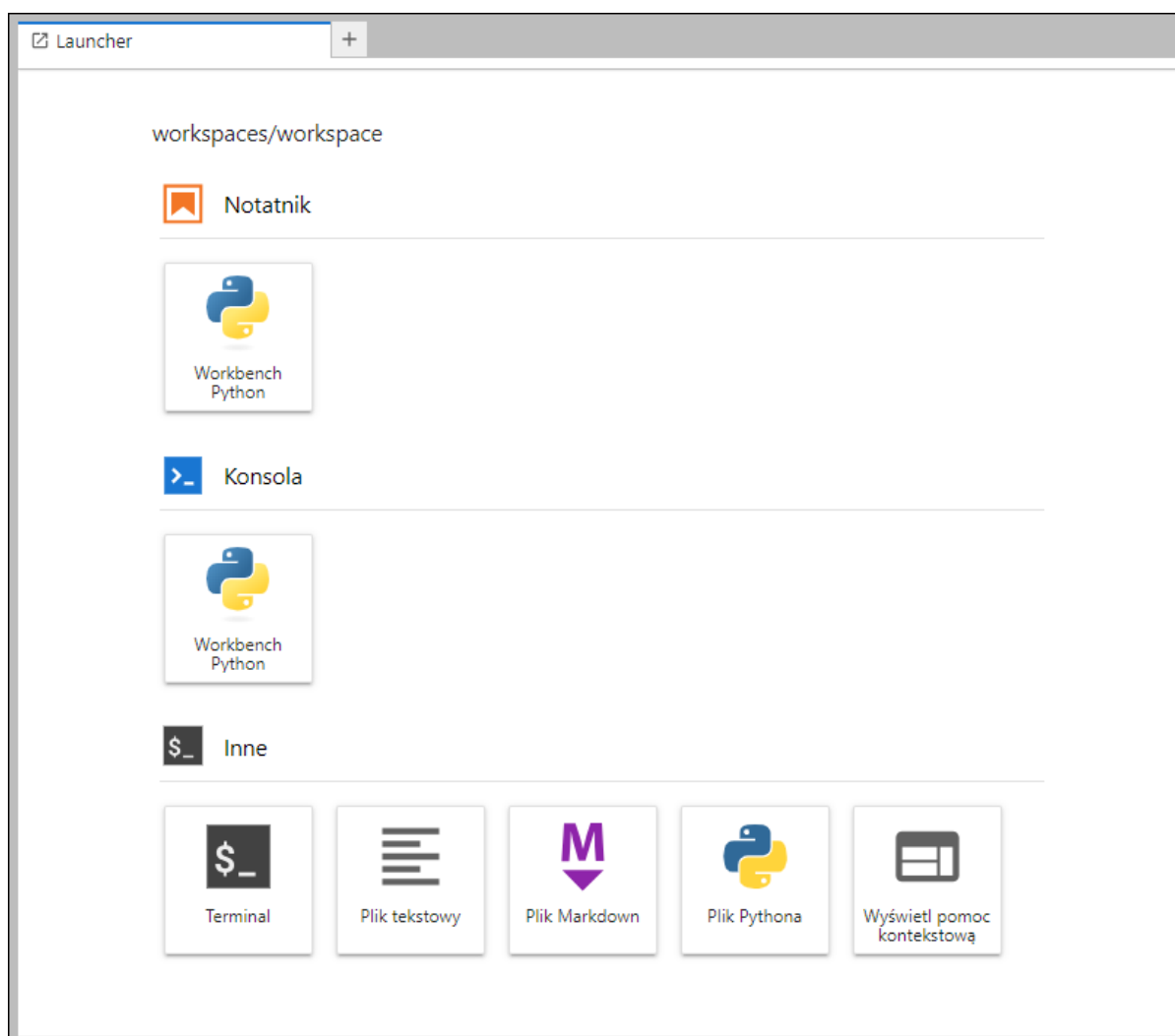
Aby załadować źródło danych, kliknij . Przesłane dane pojawią się teraz w **Przeglądarce plików**.



Otwieranie terminalu w Jupyter Lab

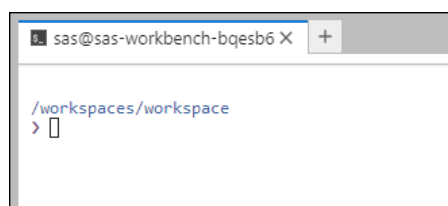
SAS Viya Workbench zawiera pełne środowisko programistyczne, w tym możliwość wywoływania poleceń powłoki.

Możesz otworzyć terminal ze strony uruchamiania Jupyter Lab. Jeśli nie widzisz strony uruchamiania, wybierz **Plik** ⇒ **Nowy launcher**.



Na stronie modułu uruchamiającego możesz otworzyć terminal, wybierając płytkę **Terminal**.

Domyślnie terminal otwiera powłokę Bash w katalogu wskazanym jako folder montażowy podczas tworzenia przestrzeni roboczej.

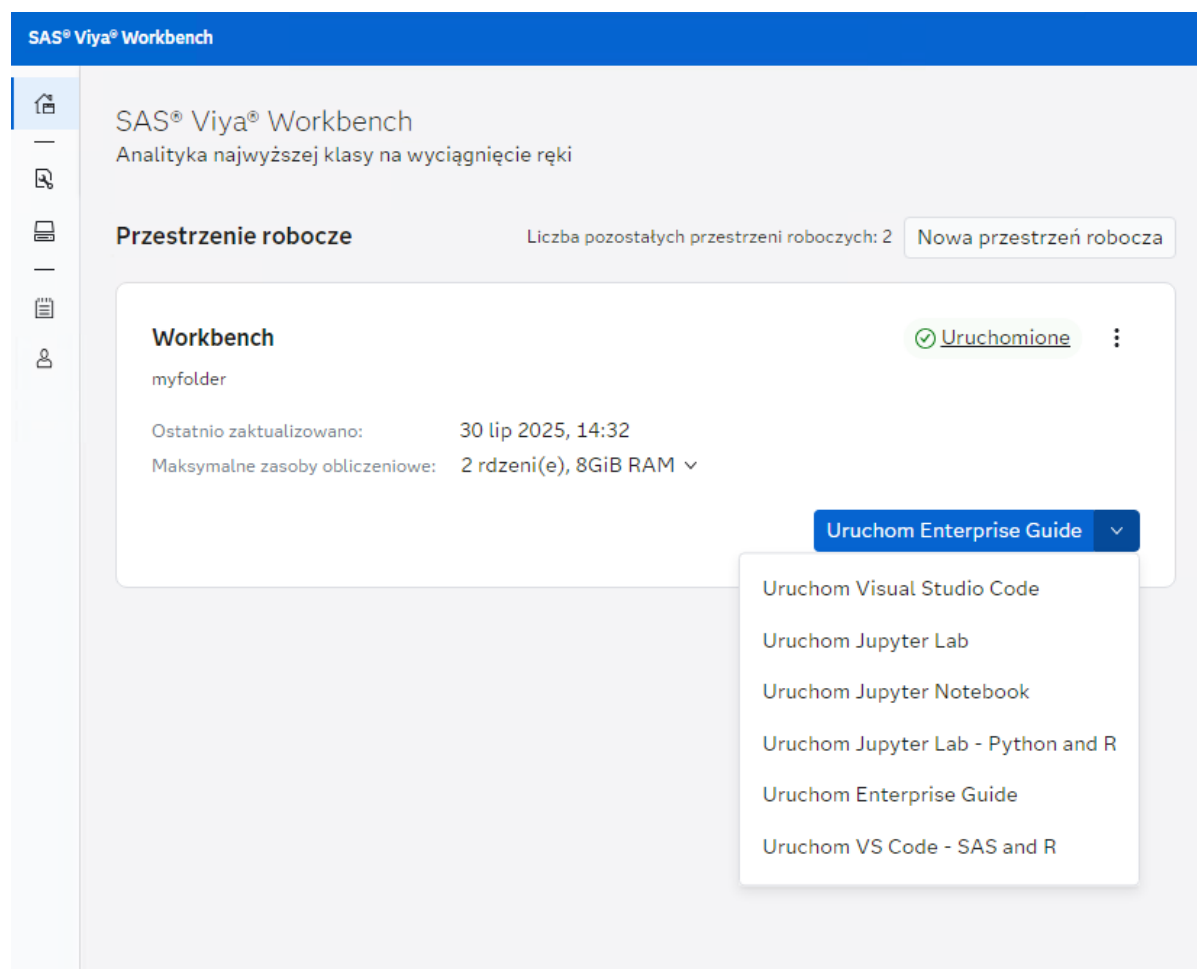


Praca w aplikacji SAS Enterprise Guide

<i>Otwórz przestrzeń roboczą w aplikacji SAS Enterprise Guide</i>	45
<i>Dostęp do przestrzeni roboczej w aplikacji SAS Enterprise Guide</i>	46

Otwórz przestrzeń roboczą w aplikacji SAS Enterprise Guide

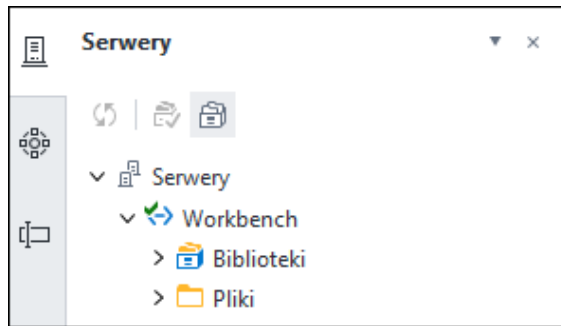
Można otworzyć przestrzeń roboczą ze strony startowej. Z listy rozwijanej wybierz **Uruchom Enterprise Guide**.



Dalsze informacje znajdziesz w ["Definiowanie połączenia z serwerem SAS Viya Workbench"](#) w *SAS Enterprise Guide: Przewodnik użytkownika*.

Dostęp do przestrzeni roboczej w aplikacji SAS Enterprise Guide

Po zdefiniowaniu połączenia z przestrzenią roboczą, pojawi się ono na karcie **Serwery**. Można uzyskać dostęp do bibliotek i plików SAS-owych.



Dalsze informacje na temat zarządzania połączeniem z przestrzenią roboczą znajdziesz w ["Definiowanie połączenia z serwerem SAS Viya Workbench" w SAS Enterprise Guide: Przewodnik użytkownika](#).

Praca z procesami SAS-owymi

<i>Czym są procesy SAS-owe?</i>	49
<i>Strona procesów w SAS Viya Workbench</i>	50
<i>Tworzenie definicji procesu</i>	50
<i>Przełącz proces</i>	52
<i>Polecenie sas-workbench-execute</i>	53
Format polecenia sas-workbench-execute	53
Opcje polecenia sas-workbench-execute	53
Przykład: Polecenie sas-workbench-execute	55
<i>Rozwiązywanie problemów: Procesy SAS-owe</i>	57
Wyświetlanie logów procesów	57
Dlaczego po uruchomieniu procesu pojawia się błąd?	57
Co oznacza status Oczekujące?	57

Czym są procesy SAS-owe?

Możesz używać procesów do raportowania internetowego, wykonywania analiz, tworzenia aplikacji internetowych i dostarczania zawartości do klientów takich jak SAS Studio. Klient pracuje z procesami zawierającymi kod SAS-owy. Procesy te mogą uzyskiwać dostęp do dowolnego źródła danych SAS-owych lub pliku zewnętrznego i tworzyć tabele, pliki lub inne dane docelowe obsługiwane przez SAS-a.

Proces SAS-owy składa się z programu i jego definicji.

- Program zawiera kod SAS-owy, Pythona lub R, który chcesz uruchomić.
- Definicja procesu zawiera takie informacje jak: nazwa procesu, autor oraz data i godzina utworzenia.

W SAS Viya Workbench można utworzyć definicję procesu i przekazać do wykonania istniejące procesy.

Strona procesów w SAS Viya Workbench

Aby utworzyć proces, wyświetl istniejące procesy SAS-owe lub przełącz proces do wykonania, kliknij .

Strona Procesy zawiera dwie karty:

- **Definicje procesów** wyświetla listę wszystkich procesów, które utworzyłeś lub masz uprawnienie do edycji. Na tej karcie można także utworzyć definicję procesu.
- **Przekazywanie procesów** pokazuje historię przekazanych do wykonania procesów i ich status. Na tej karcie można wyświetlić logi przekazania do wykonania i ponownie przekazać proces do wykonania, korzystając z parametrów podanych podczas ostatniego przekazania procesu.

Uwaga: Twój administrator SAS Viya Workbench może wyświetlić wszystkie procesy SAS-owe i ich status na tych dwóch kartach.

Tworzenie definicji procesu

- 1 Aby otworzyć stronę Procesy, kliknij .

- 2 Na karcie **Definicje procesów** kliknij **Nowy**.

Uwaga: Aby zmodyfikować definicję procesu, kliknij nazwę definicji. Dostępne są te same opcje. Nie ma natomiast wersjonowania procesów. Możesz mieć tylko jedną definicję procesu o tej samej nazwie.

- 3 Wpisz nazwę definicji procesu.
- 4 W wierszu **polecenie** wpisz polecenie uruchamiające proces.

Oto kilka przykładowych poleceń:

- `sas-workbench-execute /workspaces/myfolder/sales.sas`. Dalsze informacje na temat opcji dostępnych dla tego polecenia znajdziesz w ["Polecenie sas-workbench-execute" na stronie 53](#).
- `python /workspaces/myfolder/sales.py`

- Oto przykład użycia baska i Pythona do utworzenia definicji procesu:

```
a=Workbench
echo "Hello $a"
echo "print('Hello from Python $a')" > /workspaces/myfolder/hello.py
python3 /workspaces/myfolder/hello.py
```

WAŻNE Aby podać plik zawierający kod procesu SAS-owego, trzeba podać pełną ścieżkę. Aby ustalić pełną ścieżkę do pliku, najedź kursorem na nazwę pliku w zintegrowanym środowisku programistycznym dla Twojego warsztatu.

- 5 Podaj ustawienia przekazywania do wykonania dla procesu. Pola te są wypełnione wartościami domyślnymi. Ustawienia te można zmienić dla definicji procesu.

Uwaga: Procesy mają dostępne takie same ustawienia jak Workbench. Kliknijcie **Kopiuj z**, aby skopiować ustawienia z warsztatu lub z istniejącej definicji procesu.

- Z listy rozwijanej **Maksymalne zasoby obliczeniowe** wybierz liczbę używanych rdzeni. Możesz wybierać spośród 2, 4 lub 16 rdzeni.
- Z listy rozwijanej **Wolumin główny** wybierz miejsce przechowywania. Aby utworzyć wolumen, patrz [“Utwórz wolumen” na stronie 9](#).

WSKAZÓWKA Domyślny wolumin nazywa się myfolder. Kiedy uruchomisz proces, logi wynikowe są przechowywane w wybranej przestrzeni magazynowej. Zazwyczaj woluminy pamięci EBS nie mogą być używane jednocześnie przez proces i przez przestrzeń roboczą.

- Wprowadź ścieżkę do folderu montażowego dla przestrzeni roboczej. Folder ten jest automatycznie tworzony w pliku **/workspaces/**.
 - Dodaj dowolne dodatkowe woluminy, które chcesz skojarzyć z tym warsztatem. Można także zdefiniować wolumen bazowy.
 - Wybierz środowisko wykonywania procesu. Usługi i uprawnienia dostępne dla procesów SAS-owych zależą od używanego środowiska.
 - Aby użyć zmiennych środowiskowych w definicji procesu, kliknij **Nowy**. W Dodaj zmienną środowiskową wpisz nazwę zmiennej środowiskowej i jej wartość. Następnie kliknij **OK**.
- 6 (Opcjonalne) Podaj harmonogram przekazywania procesów do wykonania. Dostępne są następujące opcje harmonogramowania:
 - **Brak** - Domyślnie nie jest tworzony harmonogram i definicja procesu jest uruchamiana tylko gdy przekażesz ją w SAS Viya Workbench.
 - **Rekurencja prosta** - Możesz podać, czy proces jest uruchamiany codziennie czy co tydzień, oraz odstęp dla każdego przekazania do wykonania. Można także podać porę dnia, o której ma być uruchomiony proces.

- **Wyrażenie Crona** - Użyj wyrażenia Crona, aby przekazać proces do wykonania.

WAŻNE Obecnie SAS Viya Workbench obsługuje tylko strefę czasową UTC dla procesów cron.

- 7 Zapisz definicję procesu.

.....
Uwaga: Należy zapisać definicję procesu przed przekazaniem go.
.....

Przełącz proces

- 1 Aby otworzyć stronę Procesy, kliknij  .
- 2 Przełącz do wykonania definicję procesu.
 - Na karcie **Definicje procesów** kliknij nazwę definicji procesu. Na stronie zawierającej informacje dotyczące tej definicji procesu, kliknij **Przełącz do wykonania**.
 - Jeśli wcześniej przekazano proces, kliknij **Przełącz ponownie** na stronie Przekazywanie procesów.
- 3 Domyślnie proces używa zasobów obliczeniowych i zmiennych środowiskowych podanych w definicji procesu. Aby użyć tych ustawień domyślnych, kliknij **OK** w oknie Przełącz proces.

Możesz użyć opcji w Przełącz proces, aby zmodyfikować maksymalną liczbę zasobów obliczeniowych i zmiennych środowiskowych dla danego przekazania procesu. Ustawienia te są używane tylko dla tego wykonania i nie są zapisywane w definicji procesu.

Po przekazaniu procesu jego status pojawi się w prawym okienku. Komunikatami o statusie są: W toku, Ukończono lub Błąd. W niektórych przypadkach możesz zobaczyć komunikat o oczekiwaniu. Dalsze informacje znajdziesz w ["Co oznacza status Oczekujące?" na stronie 57](#).

Polecenie sas-workbench-execute

Format polecenia sas-workbench-execute

Tworząc definicję procesu, można podać jedno lub więcej poleceń sas-workbench-execute, aby przekazać kod SAS-owy do serwera obliczeniowego. Polecenie tworzy nową sesję na podanym serwerze obliczeniowym i przekazuje program do nowej sesji. Po zakończeniu polecenia log SAS-owy jest dodawany do wynikowego logu procesu.

Możesz uruchomić polecenie sas-workbench-execute, korzystając z pełnej ścieżki do pliku zawierającego kod SAS-owy lub podając kod SAS-owy w wierszu.

- Aby odwołać się do pliku, użyj następującego formatu: `sas-workbench-execute <options> /path/to/file_to_run.sas`. Przykładowo `sas-workbench-execute /workspaces/myfolder/sales.sas` Przekazuje program sales.sas jako proces.
- Aby wstawić kod SAS-owy w wierszu, użyj następującego formatu: `sas-workbench-execute <options> '<inline sas code to run>'`. Oto przykład: `sas-workbench-execute 'proc print data=sashelp.class; run;'`.

Opcje polecenia sas-workbench-execte

Oto kilka dodatkowych opcji polecenia sas-workbench-execute:

Tabela 7.1 Opcje polecenia sas-workbench-execute

Opcja	Typ	Wymagane	Opis
--autoexec	array [string]	fałsz	Określa kod SAS-owy wykonywany podczas inicjowania sesji. Można podać tę opcję wielokrotnie, aby podać zawartość pliku autoexec lub ścieżkę do pliku zawierającego kod autoexec. Dalsze informacje znajdziesz w createSession na developer.sas.com.
--option	array [string]	fałsz	Podaje opcje systemowe SAS-a, które mają być zastosowane do sesji serwera obliczeniowego SAS-a. Opcję tę można podać wielokrotnie w wierszu poleceń.

Opcja	Typ	Wymagane	Opis
			Dalsze informacje znajdziesz w createSession na developer.sas.com .
--endpoint	array [string]	fałsz	Określa adres URL serwera obliczeniowego SAS-a. Jeśli punkt końcowy nie zostanie podany, używana jest wartość zmiennej środowiskowej WORKBENCH_COMPUTE_ENDPOINT. Jeśli nie podano żadnej z tych wartości, używana jest wartość PUP_COMPUTE_HOST. Uwaga: Opcje --endpoint i --serverId są zazwyczaj używane razem do uruchomienia sas-workbench-execute na serwerze SAS-owym, który nie jest częścią SAS Viya Workbench. Opcji tych nie trzeba używać, jeśli sas-workbench-execute polecenie względem domyślnego serwera obliczeniowego dla SAS Viya Workbench lub z terminala SAS Viya Workbench.
--ignore-code-server-connection-profile	array [string]	fałsz	Określa, że serwer nie używa wartości dla --endpoint, --serverId, --option, i --autoexec znajdujących się w pliku \$HOME/.local/share/code-server/Machine/settings.json. Wartości --endpoint i --serverId podane za pomocą wiersza poleceń nadpisują wszystkie wartości w pliku settings.json. Jeśli dodatkowe wartości --option i --autoexec są podane w wierszu poleceń, są one dołączane do wartości wczytanych z pliku settings.json.
--serverId	array [string]	fałsz	Określa identyfikator serwera obliczeniowego SAS-a. Domyślną wartością jest 0001.
--variable	array [string]	fałsz	Określa zmienne SAS-owe ustawiane dla sesji serwera obliczeniowego SAS-a. Dalsze informacje znajdziesz w createJob na developer.sas.com.

Przykład: Polecenie sas-workbench-execute

Tabela 7.2 Przykłady: Polecenie sas-workbench-execute

Wyjaśnienie	Przykładowy kod
Przykład opcji wielokrotnych z kodem wbudowanym na końcu w pojedynczych cudzysłowach.	<code>sas-workbench-execute --option fullstimer --option fmterr 'proc print data=sashelp.class; run;'</code>
Przykład opcji wielokrotnych w programie SAS-owym zamiast w kodzie wierszowym.	<code>sas-workbench-execute --option fullstimer --option fmterr /workspaces/myfolder/sales.sas</code>
Przykład, gdzie --autoexec wskazuje na plik tymczasowy. Uwaga: Wartości podane w pliku uruchamianym po instrukcjach domyślnego pliku autoexec.sas in \$HOME/.local/share/code-server/Machine/settings.json.	<code>sas-workbench-execute --option fullstimer --option fmterr --autoexec /tmp/autoexec.sas /workspaces/myfolder/sales.sas</code>
Przykład z użyciem wiersza autoexec zamiast pliku. Uwaga: Wartości podane w opcji --autoexec mają pierwszeństwo przed domyślną opcją autoexec.sas w \$HOME/.local/share/code-	<code>sas-workbench-execute --option fullstimer --option fmterr --autoexec 'options pagesize=80;' 'proc options; run;'</code>

Wyjaśnienie	Przykładowy kod
server/Machine/settings.json.	
Przykład użycia opcji <code>--variable</code> przypisywania makrozmiennych SAS-owych.	<pre>sas-workbench-execute --variable myvar='Macro var set from cmd line' --variable myvar2='and another' 'data _null_; put "&myvar &myvar2"; run;'</pre>
Przykład wielowierszowy z użyciem <code>bash</code> i <code>sas-workbench-execute</code> .	<pre>sas-workbench-execute 'proc options; run;' echo 'STARTING JOB1.SAS' sas-workbench-execute /workspaces/myfolder/job1.sas echo 'STARTING JOB2.SAS' sas-workbench-execute /workspaces/myfolder/job2.sas</pre>
Uwaga: Każdy z programów SAS-owych działa w innej sesji serwera SAS-owego, zatem programy te nie współdzielą zasobów tymczasowych, takich jak Praca SAS-a i zmienna. Zasoby te są współdzielone, jeśli programy są uruchamiane interaktywnie w tej samej sesji SAS Viya Workbench.	
Przykład alternatywny umożliwiający uruchamianie obu programów SAS-owych w tej samej sesji serwera SAS-owego.	Uwaga: Jest to alternatywa dla wielowierszowego przykładu, w którym użyto <code>bash</code> i <code>sas-workbench-execute</code>. Użycie pierwszego bloku kodu jest modyfikacją pliku <code>job1.sas</code> w celu wywołania <code>job2.sas</code> na końcu tej samej sesji. <pre>/* Use %include to call job2.sas at the end of job1.sas */ %global job1; %let job1='This is job1'; data _null_; put "&job1"; put "&SYS_COMPUTE_SESSION_ID"; run; %include "&WORKSPACE_PATH/job2.sas";</pre> Oto kod tworzący definicję procesu: <pre>/* Now run job1.sas as part of the job run */ sas-workbench-execute 'proc options; run;' echo 'STARTING JOB1.SAS'</pre>

Wyjaśnienie	Przykładowy kod
	<code>sas-workbench-execute /workspaces/myfolder/job1.sas</code>
Przykład użycia -- ignore-code- server- connection- profile co oznacza, że polecenie ignoruje ustawienia w <code>\$HOME/.local/ share/code- server/Machine/ settings.json</code> .	<code>sas-workbench-execute --option fullstimer --ignore- code-server-connection-profile 'proc options; run;'</code>

Rozwiązywanie problemów: Procesy SAS-owe

Wyświetlanie logów procesów

Aby wyświetlić log procesu SAS-owego, otwórz kartę **Przekazywanie procesów**. Dla określonego procesu kliknij **Wyświetl logi**. Można także zapisać logi jako tekst w miejscu przechowywania używanym przez proces.

Dlaczego po uruchomieniu procesu pojawia się błąd?

Może pojawić się stan błędu, jeżeli proces istnieje z niezerowym kodem zakończenia.

Co oznacza status Oczekujące?

Kiedy przekazujesz proces do wykonania, SAS Viya Workbench wyświetla status procesu (np. W toku, Ukończony lub Błąd). Jeśli zobaczysz status Oczekiwanie,

oznacza to, że środowisko i definicja procesu rywalizują o zasoby. Aby uruchomić proces SAS-owy, najpierw zatrzymaj bieżącą przestrzeń roboczą.

Załącznik 1

Względy bezpieczeństwa

<i>Ładowanie i pobieranie danych</i>	59
<i>Przechowywanie danych uwierzytelnienia użytkownika</i>	59

Ładowanie i pobieranie danych

Podczas pracy z SAS Viya Workbench, funkcjonalność przesyłania i pobierania plików jest zapewniana przez zintegrowane środowisko programistyczne. SAS Viya Workbench nie ma żadnych ograniczeń co do typu i rozmiaru pliku. SAS Viya Workbench nie egzekwuje żadnych ograniczeń prędkości. Pamiętaj, aby postępować zgodnie z wytycznymi bezpieczeństwa opracowanymi przez Twoją firmę.

Przechowywanie danych uwierzytelnienia użytkownika

W SAS Viya Workbench, użytkownicy mogą przechowywać swoje poświadczenia bazy danych w pliku.

- Dostęp użytkownika do zewnętrznej relacyjnej bazy danych wymaga, aby poświadczenia użytkownika były przechowywane w pliku. Opcja AUTHDOMAIN= nie jest obsługiwana.
- Poświadczenia mogą być przechowywane w katalogu domowym użytkownika (plik autoexec, plik INI) lub w instrukcji LIBNAME w programie. Dalsze

informacje znajdziesz w [“LIBNAME Statement Examples for PostgreSQL”](#) w [Accessing External Data Sources with SAS®Viya Workbench](#).

- SAS Viya Workbench nie wyświetla ostrzeżenia ani błędu, jeśli Twoje dane uwierzytelniające nie są zaszyfrowane.

Podczas pracy z SAS Viya Workbench, pamiętaj o przestrzeganiu wytycznych bezpieczeństwa obowiązujących w Twojej firmie.

Załącznik 2

Wskazówki dotyczące rozwiązywania problemów i często zadawane pytania

<i>Jakie są różnice pomiędzy SAS Viya Workbench i platformą SAS Viya?</i>	61
<i>Czy SAS Viya Workbench obsługuje SASPy?</i>	62
<i>Co się stanie, jeśli przestrzeń robocza przekroczy przydzielony limit miejsca?</i>	62
<i>Jak przywrócić połączenie po przekroczeniu limitu czasu w SAS Viya Workbench</i>	63
Rozwiązywanie problemów: Włączanie plików cookie podmiotów zewnętrznych dla SAS Viya Workbench	64
Błąd ładowania widoku internetowego	64
Chrome: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych	65
Mozilla Firefox: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych	65
Microsoft Edge: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych	66
Rozwiązywanie problemów: Możliwy błąd podczas korzystania z pakietów SWAT i SASVIYA	66

Jakie są różnice pomiędzy SAS Viya Workbench i platformą SAS Viya?

Oto kilka znaczących różnic między SAS Viya Workbench a platformą SAS Viya:

- Libref umożliwia połączenia ze źródłem danych zamiast biblioteki CAS-owej.

- Opcja PRESERVE_COL_NAMES=YES jest wymagana do włączenia zapisywania procedur SAS Visual Analytics i SAS Machine Learning w Oracle. Tę opcję można określić jako opcję LIBNAME lub opcję zbioru danych. PRESERVE_COL_NAMES=YES jest opcjonalne w przypadku zapisywania do Oracle'a w ramach innych procedur.
- [“Amazon Redshift: Functional Differences between SAS Viya Workbench and SAS Viya” w Accessing External Data Sources with SAS®Viya Workbench](#)
- [“Oracle: Functional Differences between SAS Viya Workbench and SAS Viya Platform” w Accessing External Data Sources with SAS®Viya Workbench](#)
- [“PostgreSQL: Functional Differences between SAS Viya Workbench and SAS Viya Platform” w Accessing External Data Sources with SAS®Viya Workbench](#)
- [“Snowflake: Functional Differences between SAS Viya Workbench and SAS Viya Platform” w Accessing External Data Sources with SAS®Viya Workbench](#)
- [“Microsoft SQL Server: Functional Differences between SAS Viya Workbench and SAS Viya Platform” w Accessing External Data Sources with SAS®Viya Workbench](#)

Czy SAS Viya Workbench obsługuje SASPy?

SASPy nie jest zainstalowane w SAS Viya Workbench. Możesz zainstalować SASPy; jednakże ta funkcjonalność nie została przetestowana, oraz SAS Viya Workbench nie konfiguruje niczego do użytku z SASPy.

Co się stanie, jeśli przestrzeń robocza przekroczy przydzielony limit miejsca?

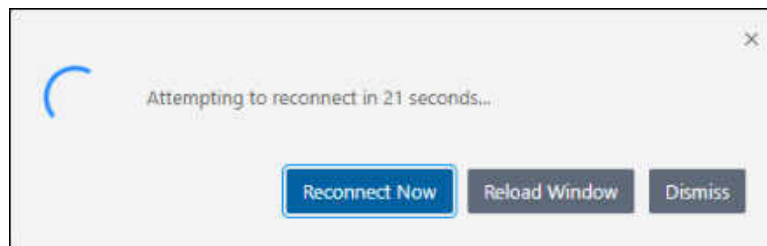
Domyślnie każde miejsce przechowywania (magazyn) ma do dyspozycji 5 GB. Możesz zmienić tę wartość podczas tworzenia magazynu. Zawsze możesz zwiększyć ilość miejsca na dysku, ale nie możesz go zmniejszyć.

Jeśli przekroczysz tę ilość miejsca, przestrzeń robocza automatycznie ponownie się uruchomi. Jeśli przestrzeń robocza uruchomi się ponownie, odśwież stronę startową SAS Viya Workbench.

Jak przywrócić połączenie po przekroczeniu limitu czasu w SAS Viya Workbench

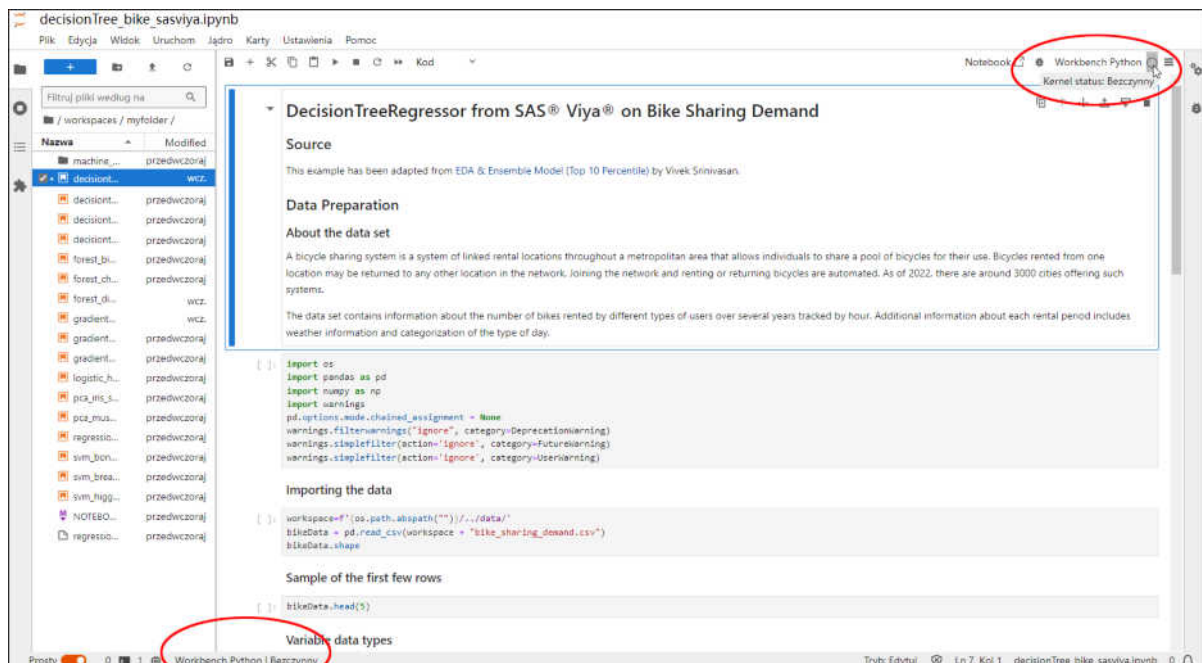
SAS Viya Workbench wygasa po 30 minutach bezczynności.

Jeśli używasz SAS Viya Workbench w kodzie VS, wówczas okno ponownego połączenia pojawi się po 30 minutach.

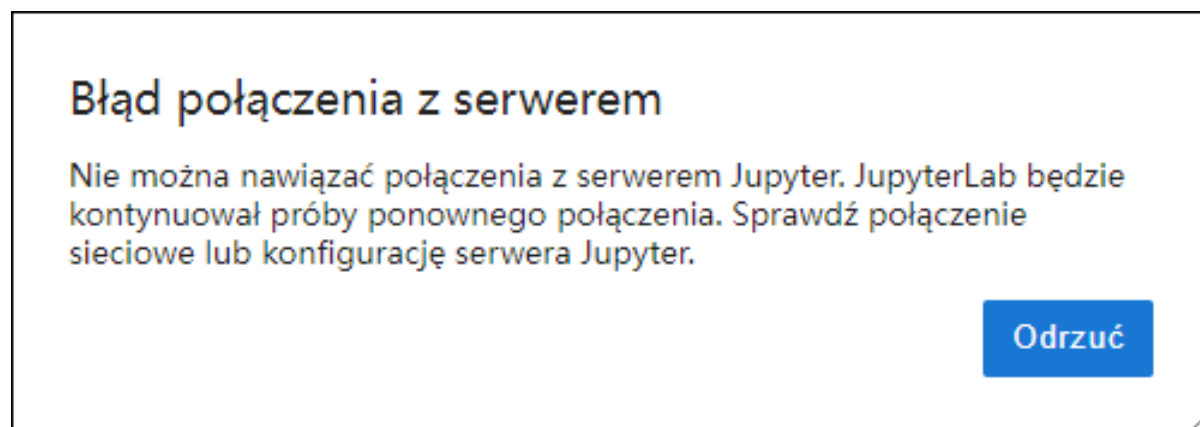


Nie klikaj **Przywróć połączenie**. Zamiast tego zamknij kartę przeglądarki zawierającą SAS Viya Workbench i użyj adresu URL podanego przez SAS-a, aby ponownie otworzyć SAS Viya Workbench.

Jeśli używasz SAS Viya Workbench w Jupyter Notebook lub w Jupyter Lab, w obszarze roboczym może pojawić się status bezczynności.



Możesz także zobaczyć błąd połączenia z serwerem po 30 minutach w Jupyter.

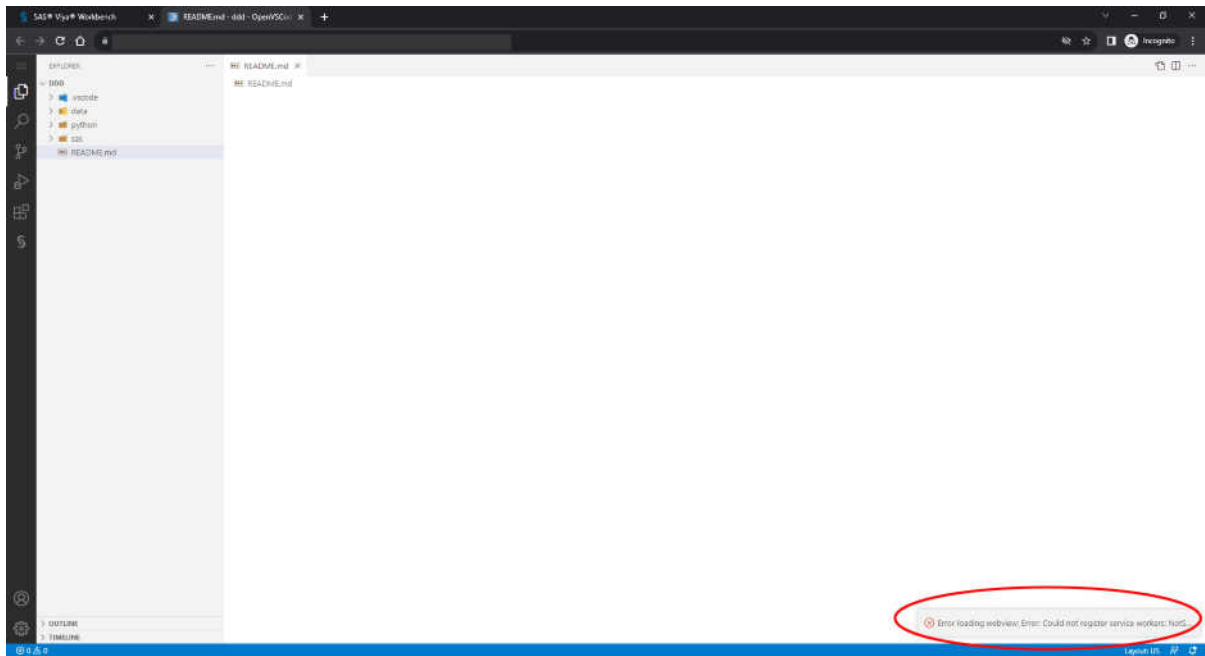


Aby ponownie otworzyć SAS Viya Workbench, zamknij ten komunikat i odśwież okno przeglądarki.

Rozwiązywanie problemów: Włączanie plików cookie podmiotów zewnętrznych dla SAS Viya Workbench

Błąd ładowania widoku internetowego

Jeśli w przeglądarce zablokowałeś pliki cookie podmiotów zewnętrznych lub korzystasz z okna incognito, podczas otwierania przestrzeni roboczej w VS Code może pojawić się błąd.



Aby rozwiązać ten problem, musisz włączyć obsługę plików cookie podmiotów zewnętrznych w SAS Viya Workbench.

Chrome: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych

Uwaga: Nazwy opcji mogą się różnić w zależności od wersji przeglądarki.

- 1 W przeglądarce Chrome otwórz swoje ustawienia.
- 2 W lewym okienku nawigacji kliknij **Prywatność i ochrona**.
- 3 Rozwiń sekcję oznaczoną etykietą **Pliki cookie podmiotów zewnętrznych**.
- 4 Kliknij **Dodaj** obok **Strony, które zawsze korzystają z plików cookie**.
- 5 W polu **Strona**, dodaj adres URL SAS Viya Workbench. Kliknij **Dodaj**.

Mozilla Firefox: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych

Uwaga: Nazwy opcji mogą się różnić w zależności od wersji przeglądarki.

- 1 W przeglądarce Firefox otwórz swoje ustawienia.
- 2 W lewym okienku nawigacji kliknij **Prywatność i ochrona**.
- 3 Obok nagłówka **Usprawniona ochrona przed śledzeniem**, kliknij **Zarządzaj wyjątkami**.
- 4 W polu **Adres strony internetowej** wpisz adres URL SAS Viya Workbench. Kliknij **Dodaj wyjątek**.
- 5 Kliknij **Zapisz zmiany**.

Microsoft Edge: Aktualizowanie listy zaufanych plików cookie podmiotów zewnętrznych

- 1 W przeglądarce Edge otwórz swoje ustawienia.
- 2 W lewym okienku nawigacji kliknij **Prywatność, wyszukiwanie i usługi**.
- 3 Pod nagłówkiem **Zapobieganie śledzeniu** nagłówek, kliknij **Wyjątki**.
- 4 Na stronie **Wyjątki zapobiegające śledzeniu**, kliknij **Dodaj witrynę**.
- 5 W polu **Strona**, dodaj adres URL SAS Viya Workbench. Kliknij **Dodaj**.

Rozwiązywanie problemów: Możliwy błąd podczas korzystania z pakietów SWAT i SASVIYA

Podczas korzystania z interfejsów SAS API do kodowania w Pythonie, może pojawić się błąd podczas korzystania zarówno z pakietu SAS Scripting Wrapper for Analytics Transfer (SWAT), jak i pakietu SASVIYA dla języka Python. (Pakiet SWAT służy do komunikacji z SAS-owym serwerem CAS-a). Ten błąd występuje, ponieważ SWAT i SASVIYA używają nakładającego się zestawu ustawień wewnętrznych, a SWAT musi uzyskać dostęp do tych ustawień przed inicjalizacją SASVIYA.

Aby naprawić ten błąd, zaimportuj pakiet SWAT przed zaimportowaniem pakietu SASVIYA.

Załącznik 3

Licencje i informacje podmiotów zewnętrznych

<i>Wiadomości ogólne</i>	67
<i>Jupyter</i>	68
<i>Jupyter Lab</i>	68
<i>Serwer OpenVSCode</i>	69
<i>Matplotlib</i>	70
<i>scikit-learn</i>	70
<i>seaborn</i>	71
<i>Teradata</i>	72

Wiadomości ogólne

Używanie SAS Viya Workbench wymaga dostępu do oprogramowania podmiotów zewnętrznych.

Ta sekcja zawiera informacje na temat wszelkiego kodu podmiotów zewnętrznych zawartego w oprogramowaniu SAS-owym, w tym uwagi dotyczące oprogramowania podmiotów zewnętrznych i/lub dodatkowe warunki. Dodatkowe informacje dotyczące podmiotów zewnętrznych są dostępne pod adresem [Uwagi dotyczące oprogramowania podmiotów zewnętrznych](#).

Jupyter

Na stronie: <https://github.com/jupyter/notebook/blob/main/LICENSE>

BSD 3-Clause License

- Copyright (c) 2001-2015, IPython Development Team
- Copyright (c) 2015-, Jupyter Development Team

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- 1 Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
- 2 Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
- 3 Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Jupyter Lab

Na stronie: <https://github.com/jupyterlab/jupyterlab/blob/main/LICENSE>

Copyright (c) 2015-2022 Project Jupyter Contributors

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- 1 Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
- 2 Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
- 3 Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Licencja na plik Semver

=====

Plik semver.py pochodzi z <https://github.com/podhmo/python-semver>, który jest objęty licencją „MIT”. Szczegóły znajdziesz w pliku semver.py.

Serwer OpenVSCode

Na stronie: <https://github.com/gitpod-io/openvscode-server/blob/main/LICENSE.txt>

MIT License

Copyright (c) 2015 - present Microsoft Corporation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Matplotlib

Na stronie: <https://matplotlib.org/stable/users/project/license.html>

Matplotlib używa wyłącznie kodu kompatybilnego z BSD, a jego licencja opiera się na licencji PSF.

scikit-learn

Na stronie: <https://github.com/scikit-learn/scikit-learn/blob/main/COPYING>

BSD 3-Clause License

Copyright (c) 2007-2023 The scikit-learn developers.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

seaborn

Na stronie: <https://github.com/mwaskom/seaborn/blob/master/LICENSE.md>

Copyright (c) 2012-2021, Michael L. Waskom All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation, and/or other materials provided with the distribution.

Neither the name of the project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Teradata

Informacje licencyjne dla ICU 65.1

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright © 1991-2023 Unicode, Inc. All rights reserved.

Ukazuje się zgodnie z Warunkami korzystania z <https://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.