Contents

Chapter 1 / Introduction to SAS Decision Manager ................................................................. 1
  Enterprise Decision Management Systems ................................................................. 1
  About Business Rules ................................................................................................ 1
  SAS Decision Manager Features ............................................................................. 1
  Workflow for Creating and Publishing Decisions ..................................................... 2
  Sign in to SAS Decision Manager .............................................................................. 2

Chapter 2 / Working with Business Rules ............................................................................ 3
  About Rules, Rule Sets, and Expressions .................................................................. 4
  Create a New Rule Set ............................................................................................... 5
  Managing the Variables in a Rule Set ....................................................................... 5
  Defining New Rules in a Rule Set .............................................................................. 7
  Add a Stand-alone Assignment Statement ............................................................. 16
  Duplicate a Rule ........................................................................................................ 16
  Delete a Rule ............................................................................................................ 16
  Rename a Rule .......................................................................................................... 16
  Reorder Rules ........................................................................................................... 17
  Duplicate Rule Sets .................................................................................................. 17
  Move Rule Sets ......................................................................................................... 18
  Delete Rule Sets ....................................................................................................... 18
  Rename Rule Sets ..................................................................................................... 18
  Managing Versions of Rule Sets ............................................................................ 18
  Testing a Rule Set ..................................................................................................... 19
  Working with Test Output Data ............................................................................... 22
  Publish a Rule Set ..................................................................................................... 22

Chapter 3 / Working with Decisions .................................................................................... 25
  About Decisions ......................................................................................................... 26
  Create a Decision ....................................................................................................... 26
  Views for Editing a Decision ..................................................................................... 26
  Managing the Variables in a Decision .................................................................... 27
  Adding Objects to a Decision .................................................................................. 30
  Reorder Objects in a Decision ................................................................................ 32
  Delete an Object from a Decision ........................................................................... 32
  Edit the Properties of a Decision Node .................................................................. 32
  Open an Object from within a Decision .................................................................. 32
  Duplicate Decisions ................................................................................................. 33
  Move Decisions to a Different Folder ..................................................................... 33
  Delete Decisions ....................................................................................................... 33
  Rename a Decision ................................................................................................... 33
  Managing Versions of Decisions .......................................................................... 34
  Testing a Decision .................................................................................................... 34
  Working with Test Output Data .............................................................................. 38
  Publish a Decision .................................................................................................... 39

Chapter 4 / Using Lookup Tables and Functions .............................................................. 41
  About Lookup Tables and Functions ...................................................................... 41
  Create a New Lookup Table .................................................................................... 42
  Import or Refresh Lookup Table Entries ............................................................... 42
  Export a Lookup Table ............................................................................................. 43
Add Lookup Table Entries ................................................................. 43
Edit Lookup Table Entries ................................................................. 43
Delete Lookup Table Entries ................................................................. 43
Duplicate Lookup Tables ................................................................. 43
Move Lookup Tables ........................................................................... 44
Rename a Lookup Table ..................................................................... 44
Delete Lookup Tables ......................................................................... 44
Managing Versions of Lookup Tables .................................................. 44
Activate a Lookup Table ..................................................................... 45
LOOKUP Function .............................................................................. 45
LOOKUPVALUE Function ................................................................. 46
Introduction to SAS Decision Manager

Enterprise Decision Management Systems

Enterprise decision management systems can transform how businesses make decisions. They enable businesses to use the information they already have to make better decisions—decisions that are based on predictive analytics rather than on past history. Decision management systems automate the process of making decisions, particularly day-to-day operational decisions. They improve the speed, efficiency, and accuracy of routine business processes, in part by reducing the need for human intervention. By automating decisions, organizations in every industry can improve interactions with customers, partners, suppliers, and employees. In addition, organizations that are highly regulated, such as financial services, health care, and insurance, can more easily achieve compliance as a result of repeatable, traceable decisions.

SAS Decision Manager helps organizations manage data, business rules, analytical models, and optimization techniques. Rule management, model management, and data management are integrated into a consistent interface for easier accessibility.

About Business Rules

Business rules capture the logic of business decisions and are a core component of decision management systems. Business rules enable you to codify the decision-making process used by your organization. Business rules make the decision-making process transparent and adaptable, enabling organizations to respond quickly to new information about customers and markets. They enable organizations to identify and deal with fraud, avoid unnecessary risk, and find opportunities hidden in customer data.

SAS Decision Manager Features

You can use SAS Decision Manager to create a database of business rules, combine those rules together into decisions, and publish the decisions for use by other applications. SAS Decision Manager provides the following capabilities:
business rule authoring
A business rule specifies conditions to be evaluated and action to be taken if those conditions are satisfied. For example, you can create a rule that determines whether a particular customer has a mortgage. That same rule can then add the outstanding balance of the mortgage to a running total of the customer’s debt. With SAS Decision Manager, you define the conditions and actions for each rule.

rule set management and publishing
A rule set is a logical collection of rules. A single rule set can have many rules, but it generally corresponds to a single step in a decision. For example, you can have a rule set that determines a customer’s asset balance and another rule set that determines a customer’s debt level. You can use SAS Decision Manager to easily create new rule sets, reorder the rules in a rule set, add new rules to existing rule sets, and so on. When a rule set is published, the versioning features of SAS Decision Manager create a static version of the rule set. This static version helps you enforce integrity and governance over the rules that are put into production.

decision authoring and publishing
SAS Decision Manager enables you to combine analytical models, rule sets, and conditional logic into decisions. You can investigate various scenarios, test and refine the decision logic, and then publish the decisions for use in batch applications and online transactions. Automating decisions with SAS Decision Manager provides a streamlined mechanism for controlling and monitoring the rules and processes used by your organization. After a decision has been published, it is available for use by other applications.

Workflow for Creating and Publishing Decisions

To create and publish decisions by using SAS Decision Manager:
1 Create rule sets.
2 (Optional) Test rule sets.
3 (Optional) Publish rule sets.
4 Create decisions.
5 (Optional) Test decisions.
6 Publish decisions.

After you publish a rule set or decision, it is available for use by other applications. In a production environment, these applications map variables in the rule set or decision to columns in the input data. The output that is generated when a decision is executed is written to an output table. The location of the input and output data is specified by the application.

Sign in to SAS Decision Manager

Note: If you are already signed in to SAS Home, you can access SAS Decision Manager by clicking Manage Decisions.

To sign in to SAS Decision Manager:
1 In the address bar of your web browser, enter the URL for SAS Decision Manager and press Enter. The Sign In page appears.
   Note: Contact your system administrator if you need the URL for SAS Decision Manager. The default URL is http://host_name/SASDecisionManager.
2 Enter a user ID and password.
3 Click Sign In.
A rule specifies conditions to be evaluated and actions to be taken if those conditions are satisfied. Rules are grouped together into rule sets. Rule sets are logical collections of rules that are grouped together because of interactions or dependencies between the rules or because they are processed together after they are published.

Most rules correspond to this form:

if condition_expressions then action_expressions

For example, suppose you have the following rule:

if customer_debt > customer_assets then app_status = 'Decline'

In this case, customer_debt > customer_assets is a condition expression, and app_status = "Decline" is an action expression.

For example, the following figure shows rule above as it appears in the rule set editor:

A single rule can contain multiple condition expressions and action expressions. Multiple condition expressions within the same rule are joined together with the AND operand. For example, suppose you define the following rule in SAS Decision Manager:

SAS Decision Manager generates the following rule:

if ((customer_debts > customer assets) AND (credit_score < 750) AND (isHomeowner = false)) then approval_status = 'Decline'

Rule sets can also contain stand-alone assignment statements, which do not have conditions. Stand-alone assignment statements always execute unless a RETURN action stops the execution of the rule set before execution reaches the assignment statement. See Step 7 of “Add a New Rule” on page 7 for information about the RETURN action.
**Create a New Rule Set**

1. Click the navigation bar.
2. Click **New Rule Set**. The New Rule Set window appears.
3. Enter a name for the rule set if you do not want to use the default name. Rule set names are limited to 100 characters and must be unique within a folder.
4. (Optional) Enter a description for the new rule set. Descriptions are limited to 1000 characters.

   **TIP** You can edit the description at any time on the **Properties** tab.

5. Click , and select the folder where you want to save the rule set.
6. Click **Save**. SAS Decision Manager opens the new rule set and displays the **Rule set** tab.

**Managing the Variables in a Rule Set**

To use a variable in an expression, you must either import the variable from another source or you must create it as a custom variable. You can import variables from data sources, rule sets, and decisions.

**About Variables and Mapping**

The **Properties of a Variable**

Table 2.1 on page 5 describes the properties of a variable.

**Table 2.1  Variable Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Name**       | Variable names can contain only alphanumeric characters and the underscore (_). They can be up to 32 characters long and must be unique within a rule set.  
                 | **Note:** Do not use any of these operators or keywords as variable names: AND, OR, IN, NOT, LIKE, TRUE, or FALSE. Do not use _N_ or any DS2 reserved word as a variable name. See **SAS DS2 Programmer's Guide** for information about reserved words in the DS2 language. |
| **Data type**  | SAS Decision Manager supports the following data types: Boolean, Character, Date, Datetime, Decimal, and Integer.  
                 | For Boolean values, enter **True** and **False**. When SAS Decision Manager generates code, it represents Boolean values using the numbers one and zero. |
| **Variable type** | A variable can be an input variable, an output variable, both, or neither (a temporary variable). See **“Input Variables, Output Variables, and Temporary Variables” on page 6** for more information. |
| **Length**     | The length of character variables that are input variables is derived from the length of the column in the input table. You can specify a length for output-only and temporary character variables. The length for all other variable types is set automatically. |
| **Description** | Descriptions are limited to 256 characters. |
Input Variables, Output Variables, and Temporary Variables

For each variable used in a rule set, you must specify whether the variable is an input variable, an output variable, both an input and an output variable, or a temporary variable.

- Input variables are variables that are present in the input table for a rule set. When a rule set is deployed in a production system, all input variables must be mapped to table columns in input data. When you test a rule set in SAS Decision Manager, for each input variable, you must either map it to a table column or specify a constant as its input value. When you create or edit a variable, clear the Input check box for any variable that you do not want to be mapped to a column in an input table or for which you do not want to specify a value.

- Output variables are variables that are written to the output table that is created when a rule set is run. When you create or edit a variable, clear the Output check box for any variable that you want to exclude from the output data.

- Temporary variables are variables that are not present in the input data and they are not written to the output table. To create a temporary variable for use only while a rule set is executing, clear both the Input and Output check boxes.

When you create a new variable, it is created as both an input and output variable by default.

Import Variables from a Data Source

1. On the Variables tab, click Add Variable, and select Data Source. The Choose a Data Source window appears.
2. Select the table from which you want to import variables, and click OK. The Add Variables window appears.
3. Select the variables that you want to import and click . To import all of the variables in the table, click .
4. Click OK.

Import Variables from a Rule Set or Decision

1. On the Variables tab, click Add Variable, and select Rule set or Decision. The Add Variables window appears.
2. Select the rule set or decision from which you want to import variables.
3. Select the variables that you want to import and click . To import all of the variables in the table, click .
4. Click OK.

Create Custom Variables

Create a Variable Dynamically

To create a variable of type Decimal, you can enter the new variable name in any field of an expression in which you can enter a variable name. SAS Decision Manager creates the variable as both an input and output variable. For example, you can create a Decimal variable named daysLate by entering it in variable field:

Create Variables on the Variables Tab

To create custom variables on the Variables tab:
1. Click Add Variable, and select Custom variable. The Add Variables window appears.
2 Enter the name of the new variable, and select the data type of the variable. See Table 2.1 on page 5 for additional information.

3 (Optional) Click \( \text{ } \) to display the Length and Description fields.

4 (Optional) Enter a length and description for the new variable. See Table 2.1 on page 5 for additional information.

5 Click Add. SAS Decision Manager adds the new variable to the table of variables. By default, variables are added to the table as both input and output variables.

6 (Optional) Clear the check boxes in the Input or Output columns.

\[ \begin{align*}
\text{Clear the Input check box for any variable that you do not want to be mapped to a column in an input table or for which you do not want to specify a value.} \\
\text{Clear the Output check box for any variable that you want to exclude from the output data.} \\
\text{Clear both the Input and Output check boxes to create a temporary variable.}
\end{align*} \]

See Table 2.1 on page 5 for additional information.

Repeat Steps 2 through 6 for each variable that you want to add.

7 Click OK to add the variables.

Delete Variables

On the Variables tab, select the check box for the variables that you want to delete, click \( \text{ } \) and select Delete.

Note: You cannot delete a variable if it is used in a rule set or decision.

Edit Variable Properties

On the Variables tab, click on the variable name of the variable that you want to edit. The Edit Variable window appears. Edit the properties as needed. See Table 2.1 on page 5 for additional information.

Defining New Rules in a Rule Set

Add a New Rule

1 Create or open the rule set.

2 Click Add rule if the rule set is empty or, if the rule set contains at least one statement, select one of the following options:

\[ \begin{align*}
\text{Add } \rightarrow \text{ Add rule} \\
\text{Add } \rightarrow \text{ Else rule}
\end{align*} \]

Add \( \rightarrow \) Add rule

Adds a new IF-THEN rule to the end of the rule set.

Add \( \rightarrow \) Else rule

Adds an ELSE clause to the currently selected rule. The ELSE clause does not have a condition, but you can add one by selecting Add \( \rightarrow \) Condition.

3 Import or create any variables that are required for the new rule that have not already been added to the rule set. You can add or create the variables on the Variables tab, or you can define variables dynamically as you author the rule. See “Managing the Variables in a Rule Set” on page 5 for more information.

4 Define the condition expression for the rule. See “Define Expressions for a Rule” on page 9 for additional information.

To add additional condition expressions to the selected rule, select Add \( \rightarrow \) Condition.

5 Define the action expressions for the rule. See “Define Expressions for a Rule” on page 9 for additional information.
To add additional action expressions to the selected rule, select Add \(\Rightarrow\) Action.

**TIP** To move condition or action expressions up or down within an IF or ELSE clause, select the expression and click ↑ or ↓.

6 (Optional) Change the rule operator to Else. If the rule is the first rule in a rule set, the rule operator must be IF.

   When you change the operator on a rule from IF to ELSE, the condition expression is preserved, and the rule becomes an ELSE clause with an IF condition. For more information, see “Controlling Which Conditions Are Evaluated” on page 8.

7 (Optional) Change the operator on the THEN clause from Assign to Return. The RETURN action stops the execution of any additional statements in the rule set. See “Controlling Which Conditions Are Evaluated” on page 8 for more information.

8 (Optional) Change the order of the new rule. Rules are evaluated sequentially. To move a rule up or down within a rule set, select the rule and click ↑ or ↓.

9 (Optional) Change the name of the rule. Rule names are limited to 100 characters and must be unique within a rule set. For instructions, see “Rename a Rule” on page 16.

**TIP** Assigning logical names to the rules makes it easier to determine which rules fired when you review rule-fired data.

10 (Optional) Clear the Record rule-fired data check box if you do not want a rule-fired record to be written each time this rule fires. See “How Rules Are Evaluated and When Rule-Fired Records Are Generated” on page 9 for more information.

11 Click \(\bigcirc\) to save the rule set. SAS Decision Manager validates the syntax of the expressions. If it does not detect any problems, it saves the rule set.

### Controlling Which Conditions Are Evaluated

By default, rules are assigned the IF rule operator, which means that the rule’s conditions are evaluated regardless of the results of previous rules. You can control whether condition expressions are evaluated by using the RETURN action and the ELSE operator.

The RETURN action stops the execution of any remaining rules in a rule set. If you are executing a single rule set, execution ends. If you are executing a decision, control moves to the next object in the decision. For example, the rule in the following figure stops the execution of any remaining rules in the rule set if the value of the Order_Quantity variable is missing.

If you set a clause’s operator to ELSE, then the clause’s conditions are evaluated only if the previous clause’s conditions evaluated to false. For example, given the rule set shown the following figure, if Order_Quantity is 9, the condition for the IF clause evaluates to false, and the condition for the first ELSE clause evaluates to true. Therefore, the action for first ELSE clause is executed, and the condition for the last ELSE clause is not evaluated. The value of Offer_Percent is set to 5.
How Rules Are Evaluated and When Rule-Fired Records Are Generated

By default, the condition expressions for all rules in a rule set are evaluated sequentially regardless of the results of previous rules. However, you can use the ELSE operator and the RETURN action to control whether condition expressions are evaluated. See “Controlling Which Conditions Are Evaluated” on page 8 for more information.

If a rule’s condition expressions evaluate to True, SAS Decision Manager executes the rule’s action expressions. The rule is said to have fired.

By default, every time a rule fires, it generates a rule-fired record. You can control when rule-fired records are generated by using the Record rule-fired data check boxes. See Step 10 in “Add a New Rule” on page 7.

Note: Stand-alone assignment statements always execute unless a RETURN action stops the execution of the rule set before execution reaches the assignment statement. Rule-fired data is not generated for standalone assignment statements.

Define Expressions for a Rule

Expressions can be up to 1024 characters long. They can contain numeric constants, character strings, variables, operators, SAS DS2 functions, and the SAS Decision Manager LOOKUP and LOOKUPVALUE functions. Action expressions can only be assignment statements. You can enter expressions directly into the expression fields, or you can use the Expression Editor to create and edit expressions.

TIP Use caution when you test for equality by using scientific notation. Two numbers that appear to be the same might evaluate to different numbers because of the precision involved in scientific notation.
For more information about entering expressions, see the following topics:

- “Using the Expression Editor” on page 10
- “Using the Lookup Expression Editor” on page 11
- “Punctuation for Data Values” on page 12
- “Operators for Use in Expressions” on page 13
- “Using the LIKE Operator” on page 14
- “Using Functions in Expressions” on page 15
- “Working with Missing Values” on page 15

Using the Expression Editor

You can use the Expression Editor to enter expressions that do not use the LOOKUP or LOOKUPVALUE functions. You must use the Expression Editor to enter expressions that use the OR operator, the concatenation (||) operator, or the exponent operator (**).

To open the Expression Editor, select an expression, and click 📒.

You can enter expressions directly into the expression field, or you can use the lists of operators, function names, and variable names to add them to the expression.

- To add an operator, single-click the operator in the rows above the expression field.
- To add a variable, click the Variables tab, and double-click the variable name.
- To add a function call, click the Functions tab, select a function name, and click 📒.
You can click **Validate** at any time to check the syntax of the expression that you are building. Click **Clear** to clear the expression field.

When you are finished building the expression, click **Save**. The Expression Editor validates the syntax of the expression. If the expression validates successfully, SAS Decision Manager adds the expression to the rule set and closes the Expression Editor.

**Note:** When you use the Expression Editor to enter an expression, the expression field is disabled and can be edited by using the Expression Editor only. To return to the default view of the field, click 

However, if you revert to the default view, any changes that you made in the Expression Editor are discarded.

### Using the Lookup Expression Editor

The Lookup Expression Editor enables you to enter expressions that use either the LOOKUP or LOOKUPVALUE functions.

**Note:** You can enter the LOOKUP function only in condition expressions, and you can enter the LOOKUPVALUE function only in action expressions.

To open the Lookup Expression Editor, select the expression, and click 

The following figure shows the Lookup Expression Editor for a condition expression.

The Lookup Expression Editor for an action expression is the same except that it specifies the LOOKUPVALUE function:

```
LookupValue("Lookup table", inputVariable)
```
You can enter the expression directly into the expression field, or you can use the lists on the Lookup Tables and Variables tabs to enter the expression.

- To add the lookup table name, select the Lookup Tables tab, and double-click the table name.
- To add the input variable name, select the Variables tab, and double-click the variable name.

You can click Validate at any time to check the syntax of the expression that you are building.

**TIP** When you select values on the tabs, the hint text in the expression field is automatically replaced. You do not need to select that text in order to replace it.

When you are finished building the expression, click Save. The Lookup Expression Editor validates the syntax of the expression. If the expression validates successfully, SAS Decision Manager adds the expression to the rule set and closes the Lookup Expression Editor.

**Note:** When you use the Lookup Expression Editor to enter an expression, the expression field is disabled and can be edited by using the Lookup Expression Editor only. To return to the default view of the field, click 15. However, if you revert to the default view, any changes that you made in the Lookup Expression Editor are discarded.

### Punctuation for Data Values

Depending on whether you use the Expression Editor or enter expressions directly into the expression fields, you must enter some values differently.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Punctuation Needed</th>
<th>Example</th>
</tr>
</thead>
</table>
| Character  | Enclose character strings in single quotation marks. For embedded quotation marks, use two single quotation marks. | 'Gold Account'  
'd''oscail' |
<p>| Date       | In the rule set editor, enter Date values by using the format DDMMYYYY. Enclose each value in single quotation marks followed by d. | '01AUG2017'd |
|            | In the Expression Editor, use the DS2 function TO_DOUBLE and specify the DATE data type in order to cast the Date value so that it can be compared correctly to other variables. See SAS DS2 Programmer’s Guide for information about date, time, and timestamp values and SAS DS2 Language Reference for information about the TO_DOUBLE function. | to_double(date '2017-11-04') |
| Datetime   | In the rule set editor, enter Datetime values by using the format DDMMYYYY:HH:MM:SS. Use 24-hour clock notation. Enclose each value in single quotation marks followed by dt. | '31AUG2017:15:00:00'dt |
|            | In the Expression Editor, use the DS2 function TO_DOUBLE and specify the TIMESTAMP data type in order to cast the Datetime value so that it can be compared correctly to other variables. See SAS DS2 Programmer’s Guide for information about date, time, and timestamp values and SAS DS2 Language Reference for information about the TO_DOUBLE function. | to_double(timestamp '2017-11-04 10:54:34.012') |</p>
<table>
<thead>
<tr>
<th>Data Type</th>
<th>Punctuation Needed</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>In the rule set editor, Boolean values are not enclosed in quotation marks. Enter only the values.</td>
<td>True, False</td>
</tr>
<tr>
<td></td>
<td>In the Expression Editor, enclose Boolean values in single quotation marks.</td>
<td>'True', 'False'</td>
</tr>
</tbody>
</table>

### Operators for Use in Expressions

The following table lists the operators that you can use in an expression. Do not enter a space between the elements of the operators <=, >=, or ^=. Some mnemonic equivalents for these operators cannot be used in SAS Decision Manager expressions. See *SAS DS2 Programmer’s Guide* for more information about specifying operators in expressions.

**Table 2.2 Operators**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Multiply</td>
<td>0.085 * sales</td>
</tr>
<tr>
<td>/</td>
<td>Divide</td>
<td>amount / 5</td>
</tr>
<tr>
<td>+</td>
<td>Add</td>
<td>num + 3</td>
</tr>
<tr>
<td>–</td>
<td>Subtract</td>
<td>sale - discount</td>
</tr>
<tr>
<td>**</td>
<td>Raises the first operand to the power of the second operand</td>
<td>num1**num2</td>
</tr>
<tr>
<td>=</td>
<td>Equal to</td>
<td>tries = maxTriesAllowed</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>!=</td>
<td>Not equal to</td>
<td>insufficientFunds != True</td>
</tr>
<tr>
<td>^=</td>
<td>Not equal to</td>
<td>balance ^= 'low'</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
<td>daysLate &gt; 5</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
<td>balance &gt;= 1000</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
<td>balance &lt;= 250</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>The maximum of the left and right operands</td>
<td>num1 &lt;&gt; num2</td>
</tr>
<tr>
<td>IN (value-list)</td>
<td>Equal to an item in value-list</td>
<td>risk in ('high','medium','low')</td>
</tr>
<tr>
<td>Operator</td>
<td>Definition</td>
<td>Example</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>NOT IN (value-list)</td>
<td>Not equal to an item in value-list</td>
<td>offerPercent not in (10,20,30)</td>
</tr>
<tr>
<td>LIKE 'pattern'</td>
<td>If the variable’s value matches the expression pattern in pattern, the result is true.</td>
<td>like 'HS%PP'</td>
</tr>
<tr>
<td>expression AND</td>
<td>If both expressions are true, the result is true.</td>
<td>dateExpired &gt;= '01AUG2015'd AND dateExpired &lt;= '31AUG2015'd</td>
</tr>
<tr>
<td>expression OR</td>
<td>If either expression is true, the result is true.</td>
<td>dateEnrolled &gt;= '01JAN2015' OR member = True</td>
</tr>
</tbody>
</table>

**Using the LIKE Operator**

The LIKE operator determines whether the value of a variable matches a pattern-matching expression. An expression that uses the LIKE operator has the following syntax:

LIKE 'pattern-matching-expression'

If a variable’s value matches the pattern that is specified by pattern-matching-expression, the expression evaluates to true (1). Otherwise, the expression evaluates to false (0).

There are three classes of pattern-matching characters.

*Table 2.3 Pattern-Matching Characters*

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>underscore (_)</td>
<td>Matches any single character</td>
</tr>
<tr>
<td>percent sign (%)</td>
<td>Matches any sequence of zero or more characters</td>
</tr>
<tr>
<td></td>
<td>Note: Be aware of the effect of trailing blanks. To match values, you might have to use the TRIM function to remove trailing blanks.</td>
</tr>
<tr>
<td>any other character</td>
<td>Matches that character</td>
</tr>
</tbody>
</table>

The LIKE expression is case sensitive. To search for mixed-case strings, use the UPCASE function to create an uppercase version of the variable that you want to search. You can use a temporary variable to store the results of the UPCASE function. Use the LIKE operator to search the uppercase version of the variable. For example, you can search the variable Part_Number for mixed-case strings that begin with HS and end with PP by using the two rules shown in the following figure.
The following table shows examples of the matches that result if you search a variable that could have these values: Smith, Smooth, Smothers, Smart, Smuggle.

**Table 2.4  Examples of LIKE Expressions**

<table>
<thead>
<tr>
<th>LIKE Expression Example</th>
<th>Matching Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>like 'Sm%'</td>
<td>Smith, Smooth, Smothers, Smart, Smuggle</td>
</tr>
<tr>
<td>like '%th'</td>
<td>Smith, Smooth</td>
</tr>
<tr>
<td>like 'S__gg%'</td>
<td>Smuggle</td>
</tr>
<tr>
<td>like 'S_o'</td>
<td>(no matches)</td>
</tr>
<tr>
<td>like 'S_o%'</td>
<td>Smooth, Smothers</td>
</tr>
<tr>
<td>like 'S%th'</td>
<td>Smith, Smooth</td>
</tr>
</tbody>
</table>

**Using Functions in Expressions**

SAS Decision Manager supports the following functions in expressions:

- **LOOKUP and LOOKUPVALUE functions.** Condition expressions can contain the LOOKUP function, and action expressions can contain the LOOKUPVALUE function. However, if the expression contains the LOOKUP or LOOKUPVALUE function, then the expression cannot contain anything else. See “LOOKUP Function” on page 45 and “LOOKUPVALUE Function” on page 46 for more information.
- **SAS DS2 functions.** Syntax information for the most commonly used functions is available by clicking on a function name in the Expression Editor. For additional information about these functions and additional DS2 functions, see *SAS DS2 Language Reference*.

**Working with Missing Values**

You can use the **MISSING** function to check for missing values. This function returns a 0 (false) or 1 (true). Missing values have a value of **false** when you use them with logical operators such as AND or OR. For more information about how DS2 processes nulls and SAS missing values, see *SAS DS2 Programmer’s Guide*.

You can also use the period (.) to denote missing numeric values, and the null string (‘_’) to denote missing character values.
Delete Condition or Action Expressions

To delete a condition or action expression, click ❌ for that expression.

Add a Stand-alone Assignment Statement

Stand-alone assignment statements always execute unless a RETURN action stops the execution of the rule set before execution reaches the assignment statement. Rule-fired data is not generated for standalone assignment statements.

1. Create or open the rule set.
2. Click Add assignment if the rule set is empty or, if the rule set contains at least one statement, select Add. The application adds an assignment statement to the top of the rule set.
3. Import or create any variables that are required for the assignment statement that have not already been added to the rule set. You can add or create the variables on the Variables tab, or you can define variables dynamically as you author the statement. See “Managing the Variables in a Rule Set” on page 5 for more information.
4. Select the variable to which you want to assign a value.
5. Enter the expression for the variable in the expression field. See “Define Expressions for a Rule” on page 9 for additional information.
6. (Optional) Move the assignment statement to a different position in the rule set. To move the statement, click ⬆️ or ⬇️.
7. Click ✒️ to save the rule set. SAS Decision Manager validates the syntax of the expressions. If it does not detect any problems, it saves the rule set.

Duplicate a Rule

To duplicate a rule, click ☐️ for the rule, and select Duplicate rule.

Delete a Rule

To delete a rule, click ☐️ for the rule, and select Delete rule.

Rename a Rule

To rename the IF clause of a rule:
1. Click ☐️ for the rule, and select Rename rule. The Rename Rule window appears.
2. Enter the new name and click Rename.

To rename the ELSE clause of a rule:
1. Right-click on the Else operator and select Rename rule.
2 Enter the new name and click Rename.

**TIP** Rule names for ELSE clauses do not appear in the rule set editor.

---

### Reorder Rules

To move a rule up or down within an IF or ELSE clause, select the rule, and click ↑ or ↓.

---

### Duplicate Rule Sets

**Note:** You cannot duplicate a rule set if it is open.

To duplicate a single rule set:

1 In the Rule Sets view, select the rule set that you want to duplicate.
2 Click ⬇ and select Duplicate. The Duplicate Rule Set window appears.
3 Enter a new name for the duplicate rule set.
4 (Optional) Enter a description for the rule set.
5 Click Duplicate.

To duplicate multiple rule sets:

1 In the Rule Sets view, select the rule sets that you want to duplicate.
2 Click ⬇ and select Duplicate. SAS Decision Manager duplicates the rule sets and appends _Copy to the names of the duplicate copies. If needed, a number is also appended to the names of the duplicate copies.
Move Rule Sets

1. In the Rule Sets view, select the rule sets that you want to move.
2. Click and select Move. The Choose a Location window appears.
3. Select the location to which you want to move the rule sets, and click OK.

Delete Rule Sets

Note: You cannot delete a rule set if it is open.

In the Rule Sets view, select the rule sets that you want to delete, click , and select Delete.

Rename Rule Sets

Note: You cannot rename a rule set if it is open.

1. In the Rule Sets view, select the rule set that you want to rename.
2. Click and select Rename. The Rename window appears.
3. Enter a new name for the rule set, and click Rename.

Managing Versions of Rule Sets

To change the version of a rule set that is specified in a decision node, edit the properties of the node. For more information, see “Edit the Properties of a Decision Node” on page 32.

Set the Displayed Version

The displayed version is the version whose information is displayed on the other tabs, such as the Properties and Rule set tabs. On the Versions tab, a ✓ indicates the displayed version. To change the displayed version, select the version that you want to view, and click Set Version.

Create a New Version

Note: The current version of an object is the version with the highest version number. When you create a new version, SAS Decision Manager locks the current version before it creates the new version.

Note: You cannot save changes to a version that is locked. If you modify a version that is locked and click , SAS Decision Manager asks you if you want to replace the current unlocked version with your edited version.

Note: You cannot unlock a locked version.

To create a new version:
2 Select the version type: **Minor** or **Major**. Version numbers follow the format **Major.Minor**. If you select **Major**, the number to the left of the period is incremented. If you select **Minor**, the number to the right of the period is incremented.

3 (Optional) Enter information about the new version in the **Notes** field.

**TIP** You can edit these notes at any time on the **Versions** tab.

4 Click **Save**.

---

### Testing a Rule Set

Testing a decision is optional, but it enables you to discover any problems with a decision before it is published and incorporated into a production system.

#### Create and Run a New Test

1 On the **Tests** tab, click **New Test**. The New Test window appears.
2 Enter a name for the test if you do not want to use the default name.
3 (Optional) Enter a description for the test. Descriptions are limited to 1000 characters.
4 Click , select the input table for the test, and click **OK**.
5 Map variables.

   SAS Decision Manager automatically maps the input variables in the rule set to columns in the input table when the names and data types of the variables match those of the table columns. If any input variables cannot be mapped automatically, an error message is displayed.

   **Note:** Do not enclose character strings in quotation marks.

6 (Optional) Click **Advanced** to display the advanced options.
7 (Optional) Click and select the library where you want to write the output of the test.
8 (Optional) Select the version of the rule set that you want to test.
9 (Optional) Select **Preserve unmapped columns in the output table** if you want columns that are not mapped to an output variable to be written to the output table.
10 Click **Run** to run the test. Alternatively, click **Save** to save the test definition without running it.

The status of the test is indicated by the icon in the **Status** column.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="icon.png" alt="Icon" /></td>
<td>The test is not ready to run. The test definition is not complete, or it might contain errors.</td>
</tr>
</tbody>
</table>
### Icon | Status
---|---
| ![green](Icon) | The test is defined correctly and is ready to run.
| ![green](Icon) | The test is running.
| ![green](Icon) | The test completed successfully.
| ![green](Icon) | The test completed, but warnings were issued in the SAS log. The URI to the log file is shown on the Test Results page. See Step 11.
| ![red](Icon) | The test did not run successfully. Check the SAS log for information. The URI to the log file is shown on the Test Results page. See Step 11.

11 Click in the **Results** column to view the results of the test. The Test Results page displays information about the test, including the URIs for the test definition and test results. It also includes URIs to the SAS code that was run by SAS Decision Manager, the output data set, and the SAS log that was generated when the code was run.

### Run a Rule-Fired Analysis

If a rule’s conditions evaluate to True, then the rule is said to have fired. Rule-fired data includes summary information about how many times each rule fired and detailed information for each time that a rule evaluates to True. See "How Rules Are Evaluated and When Rule-Fired Records Are Generated" on page 9 for more information.

1 Click **Rule-Fired Analysis** in the navigation pane.
2 Click **Run Rule-Fired Analysis**. SAS Decision Manager analyzes the test results to determine which rules fired for each row in the input table.
3 Click **Analysis** in the navigation pane. The Analysis page displays the output table at the top of the page and the rule-fired table at the bottom of the page. If you select a row in the output table, the rules that fired for that output row are displayed in the bottom table.

For example, in the following analysis results, the selected output record was generated by the first rule in the Low_Ratio rule set.
Click **Plot** in the navigation pane. SAS Decision Manager displays a bar chart that shows how many times each rule fired. Position your cursor over a bar to display the name of the rule and the number of times that the rule fired.
Working with Test Output Data

After you run a test, you can work with the output table in other SAS applications to analyze the data, create and compare models, discover relationships hidden in the data, and generate reports based on the data.

Note: The actions available to you depend on the applications that are available at your site.

On the Test Results page, select the **Output** table in the navigation pane, click **Actions**, and select one of the following options:

- **Prepare Data** opens the output table in SAS Data Studio. SAS Data Studio enables you to perform data transforms such as joining tables, appending data to a table, transposing columns, creating calculated columns, and so on. For more information, see *SAS Data Studio: User’s Guide*.

- **Manage Data** opens SAS Data Explorer. SAS Data Explorer enables you to import data, connect to databases, and load tables into memory. For more information, see *SAS Data Explorer: User’s Guide*.

- **Explore and Visualize Data** opens the output table in SAS Visual Analytics. SAS Visual Analytics enables you to create, test, and compare models based on the patterns discovered during exploration of the data. You can export the model before or after performing model comparison for use with other SAS products or to put the model into production. SAS Visual Analytics supports a range of visualization, discovery, and reporting features. For more information, see *SAS Visual Analytics: Overview*.

- **Build Models** enables you to create a new project in Model Studio using the output table as the data source. SAS Model Studio is an integrated visual environment that provides a suite of analytic data mining tools to facilitate end-to-end data mining analysis. Model Studio is a common interface for SAS Visual Forecasting, SAS Visual Data Mining and Machine Learning, and SAS Visual Text Analytics. For more information, see *Model Studio: Getting Oriented*.

- **Explore Lineage** opens SAS Lineage Viewer. SAS Lineage Viewer enables you to better understand the relationships between objects in your SAS Viya applications. These objects include data, transformation processes, reports, and visualizations. For more information, see *SAS Lineage Viewer: User’s Guide*.

---

Publish a Rule Set

Publishing content makes it available to other applications. Publishing a rule set creates an entity that can be managed and run in another environment. For example, if you publish content to the SAS Micro Analytic Service destination, SAS Decision Manager creates a DS2 package that can be managed and run through the SAS Micro Analytic Service interface.

Note: The publishing destinations that are available are determined by your system administrator.

To publish a rule set:

1. Open the rule set that you want to publish. The version that is published is the displayed version.
2. (Optional) Change the displayed version:
   a. Click the **Versions** tab.
   b. Select the version that you want to publish, and click **Set Version**.
3. Click **Publish**. The Publish Rule Sets window appears.
(Optional) If you have previously published the rule set, expand the **Items to Publish** section, and select the check box in the **Replace** column.

5 Select the destination to which you want to publish, and click **Publish**. The Publishing Results window appears. It displays the published name of the object, its status, and information about any issues encountered while publishing.

**Note:** The publishing destinations that are available to you depend what is configured at your site. See “Configuring Publishing Destinations” in SAS Decision Manager: Administrator’s Guide for more information.

6 After the status changes to **Published successfully**, click **Close** to close the Publishing Results window.

7 Click **Close** to close the rule set.
Working with Decisions

About Decisions .......................................................... 26
Create a Decision ......................................................... 26
Views for Editing a Decision ................................. 26
The Decision Flow Tab versus the Decision Tab .... 26
Condition Nodes in Each Tab ...................................... 26
Controlling the Tab Display ......................................... 27
Managing the Variables in a Decision .................. 27
About Variables and Mapping ............................... 27
Import Variables from a Data Source .............. 28
Import Variables from a Rule Set or Decision .... 29
Create Custom Variables ......................................... 29
Delete Variables ...................................................... 30
Edit Variable Properties .............................................. 30
Adding Objects to a Decision ............................... 30
Add a Rule Set or Model ............................................ 30
Add a Condition ..................................................... 30
Add an Object to a Condition Path ............................ 31
Reorder Objects in a Decision ............................... 32
Delete an Object from a Decision ....................... 32
Edit the Properties of a Decision Node .............. 32
Open an Object from within a Decision ............ 32
Duplicate Decisions ................................................... 33
Move Decisions to a Different Folder ............... 33
Delete Decisions ....................................................... 33
Rename a Decision .................................................... 33
Managing Versions of Decisions ......................... 34
Set the Displayed Version ........................................ 34
Create a New Version .............................................. 34
Testing a Decision ..................................................... 34
Create and Run a New Test ..................................... 34
Run a Rule-Fired Analysis ....................................... 35
Run a Path Tracking Analysis ............................... 37
Working with Test Output Data ......................... 38
Publish a Decision ..................................................... 39
About Decisions

A decision enables you to combine rule sets, analytical models, and conditional logic into a single process. Explicitly defining a decision makes your organization’s decision-making process transparent, and enables you to monitor the process for accuracy.

Create a Decision

1. Click 🏛️ on the navigation bar.
2. Click New Decision. The New Decision window appears.
3. Enter a name for the decision if you do not want to use the default name. Decision names are limited to 100 characters and must be unique within a folder.
4. (Optional) Enter a description for the new decision. Descriptions are limited to 1000 characters. [TIP] You can edit the description at any time on the Properties tab.
5. Click 🏛️, and select the folder where you want to save the decision.
6. Click Save. SAS Decision Manager opens the new decision and displays the Decision Flow tab.

Views for Editing a Decision

The Decision Flow Tab versus the Decision Tab

There are two tabs on which you can view and edit decisions.

- The Decision Flow tab enables you to edit a decision using a graphical editor. You add and rearrange nodes in the decision diagram by dragging them.
- The Decision tab enables you to edit the decision using a tabular view similar to the rule set editor. You add and rearrange objects in the decision by using menu options and clicking icons.

You can switch between the tabs according to your personal preference. Save your work before switching tabs. Click 🔄 to refresh the view in a tab.

Condition Nodes in Each Tab


For example, suppose you have the following nodes on the Decision Flow tab:
The same nodes appear on the **Decision** tab as an IF-THEN-ELSE statement:

---

### Controlling the Tab Display

On the **Decision Flow** tab:
- Click « or » to show or hide the bars on either side of the tab.
- Click ☐ to open the diagram overview. The diagram overview is a scaled-down version of the entire diagram. The section that is currently visible on the screen is outlined. The overview is useful when a decision diagram is too large to display all of the nodes on one screen.
- Click ☐ to display the list of objects that you can add to a decision.
- Click ☐ and ☐ to enlarge or shrink the diagram.
- Click ☐ to resize the diagram to fill the display.

On the **Decision** tab:
- Click ▼ or ▶ to collapse or expand a single node in the decision.
- Click ▲ or ▼ to collapse or expand all of the nodes in the decision.

---

### Managing the Variables in a Decision

To use a variable in an expression, you must either import the variable from another source or you must create it as a custom variable. You can import variables from data sources, rule sets, and decisions.

#### About Variables and Mapping

**The Properties of a Variable**

Table 3.1 on page 28 describes the properties of a variable.
Table 3.1  Variable Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Variable names can contain only alphanumeric characters and the underscore (_). They can be up to 32 characters long and must be unique within a rule set.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Do not use any of these operators or keywords as variable names: AND, OR, IN, NOT, LIKE, TRUE, or FALSE. Do not use <em>N</em> or any DS2 reserved word as a variable name. See SAS DS2 Programmer’s Guide for information about reserved words in the DS2 language.</td>
</tr>
<tr>
<td>Data type</td>
<td>SAS Decision Manager supports the following data types: Boolean, Character, Date, Datetime, Decimal, and Integer.</td>
</tr>
<tr>
<td></td>
<td>For Boolean values, enter True and False. When SAS Decision Manager generates code, it represents Boolean values using the numbers one and zero.</td>
</tr>
<tr>
<td>Variable type</td>
<td>A variable can be an input variable, an output variable, both, or neither (a temporary variable). See “Input Variables, Output Variables, and Temporary Variables” on page 28 for more information.</td>
</tr>
<tr>
<td>Length</td>
<td>The length of character variables that are input variables is derived from the length of the column in the input table. You can specify a length for output-only and temporary character variables. The length for all other variable types is set automatically.</td>
</tr>
</tbody>
</table>
| Description | Descriptions are limited to 256 characters.  

**Input Variables, Output Variables, and Temporary Variables**

For each variable used in a decision, you must specify whether the variable is an input variable, an output variable, both an input and an output variable, or a temporary variable.

- **Input variables** are variables that are present in the input table for a decision. When a decision is deployed in a production system, all input variables must be mapped to table columns in input data. When you test a decision in SAS Decision Manager, for each input variable, you must either map it to a table column or specify a constant as its input value. When you create or edit a variable, clear the Input check box for any variable that you do not want to be mapped to a column in an input table or for which you do not want to specify a value.

- **Output variables** are variables that are written to the output table that is created when a decision is run. When you create or edit a variable, clear the Output check box for any variable that you want to exclude from the output data.

- **Temporary variables** are variables that are not present in the input data and they are not written to the output table. To create a temporary variable for use only while a decision is executing, clear both the Input and Output check boxes.

When you create a new variable, it is created as both an input and output variable by default.

**Import Variables from a Data Source**

1.  On the Variables tab, click Add Variable, and select Data Source. The Choose a Data Source window appears.
2.  Select the table from which you want to import variables, and click OK. The Add Variables window appears.
3.  Select the variables that you want to import and click OK. To import all of the variables in the table, click OK.
4.  Click OK.
Import Variables from a Rule Set or Decision
1. On the Variables tab, click Add Variable, and select Rule set or Decision. The Add Variables window appears.
2. Select the rule set or decision from which you want to import variables.
3. Select the variables that you want to import and click ➔. To import all of the variables in the table, click ➔.
4. Click OK.

Create Custom Variables

Create a Variable Dynamically
On the Decision Flow tab, you can enter the new variable by entering the variable name in the Variable or Value property fields of a condition node. SAS Decision Manager creates a variable of type Decimal that is both an input variable and an output variable. For example, you can enter a Decimal variable named daysLate by entering it in the Variable field:

Properties

Variable: dayLate

Operator: >

Comparison mode: Value

Variable or Value:

Create Variables on the Variables Tab
To create custom variables on the Variables tab:
1. Click Add Variable, and select Custom variable. The Add Variables window appears.
2. Enter the name of the new variable, and select the data type of the variable. See Table 3.1 on page 28 for additional information.
3. (Optional) Click ➔ to display the Length and Description fields.
4. (Optional) Enter a length and description for the new variable. See Table 3.1 on page 28 for additional information.
5. Click Add. SAS Decision Manager adds the new variable to the table of variables. By default, variables are added to the table as both input and output variables.
6. (Optional) Clear the check boxes in the Input or Output columns.
   - Clear the Input check box for any variable that you do not want to be mapped to a column in an input table or for which you do not want to specify a value.
   - Clear the Output check box for any variable that you want to exclude from the output data.
   - Clear both the Input and Output check boxes to create a temporary variable.
See Table 3.1 on page 28 for additional information.

Repeat Steps 2 through 6 for each variable that you want to add.
7 Click **OK** to add the variables.

### Delete Variables

On the **Variables** tab, select the check box for the variables that you want to delete, click **:*** and select **Delete**.

**Note:** You cannot delete a variable if it is used in a rule set or decision.

### Edit Variable Properties

On the **Variables** tab, click on the variable name of the variable that you want to edit. The **Edit Variable** window appears. Edit the properties as needed. See Table 3.1 on page 28 for additional information.

### Adding Objects to a Decision

By default, new objects are added immediately after the currently selected object. If no object is currently selected, the decision editor adds the new object at the beginning of the decision. If the currently selected object is a condition and the condition is expanded, the decision editor adds the new object to the THEN clause. See “Reorder Objects in a Decision” on page 32 for information about reordering the objects in a decision.

On the **Decision** tab, you cannot add a new object to a decision when the selected object is a condition and the condition is collapsed.

### Add a Rule Set or Model

1 On the **Decision Flow** tab, drag the **Rule Set** or **Model** object from the list of objects onto the diagram where you want to add it. The Select a Rule Set or Select a Model window appears.

   Alternatively, on the **Decision** tab, click **Add** and select **Add rule set** or **Add model**. The Select a Rule Set or Select a Model window appears.

2 Select the rule set or model that you want to add, and click **OK**. By default, the current unlocked version of the rule set or model is added to the decision.

   **Note:** If a new version of a model is created and the decision is not locked, the decision is updated to use the new version of the model.

3 (Optional) Select a different version of the rule set. Edit the properties of the node and change the version. See “Edit the Properties of a Decision Node” on page 32 for more information.

   **Note:** You cannot change the version of a model that is in a decision.

### Add a Condition

**Note:** When you add a condition, any objects that follow the currently selected object become part of the ELSE clause in the condition.

1 On the **Decision Flow** tab, drag the **Condition** object from the list of objects onto the diagram where you want to add it. The condition is added to the decision flow, and the **Properties** pane for the condition opens.

   Alternatively, on the **Decision** tab, click **Add** and select **Add condition**. SAS Decision Manager adds an IF-THEN-ELSE statement to the decision.

2 On the **Decision Flow** tab, in the **Variable** field on the **Properties** pane, select the first variable in the condition expression.
Alternatively, on the Decision tab, in the first field of the IF condition, select the first variable in the condition expression.

**Note:** When you specify a variable in a condition expression, SAS Decision Manager updates the properties of that variable so that it is an input variable, regardless of how the variable is defined. If it is not an input variable, update the variable properties to specify the correct variable type. For more information, see “Edit Variable Properties” on page 30 and “Input Variables, Output Variables, and Temporary Variables” on page 28.

3 Select the expression operator.
4 Select **Value** or **Variable** for the comparison mode.

**Value**
- compares the first variable to a literal value. The condition becomes `first-variable operator value`.

**Variable**
- compares the first variable to the value of a second variable. The condition becomes `first-variable operator second-variable`.

5 Select a variable or enter a literal value in the **Variable or Value** field, depending on which choice you made in Step 4.

For example, the condition expression `DEBTINC > 35.5` appears on the Decision tab as follows:

```
If DEBTINC > Value 35.5 Then
```

The following figure shows the same condition as it appears on the Properties pane on the Decision Flow tab:

**Properties**

- **Variable:**
  - DEBTINC
- **Operator:**
  - >
- **Comparison mode:**
  - Value
- **Variable or Value:**
  - 35.5

---

**Add an Object to a Condition Path**

On the Decision Flow tab, right-click on the **Yes** or **No** side of the condition node, and select one of the following options:

- **Add rule set**
  - The Select a Rule Set window appears. Select the rule set and click OK.

- **Add model**
  - The Select a Model window appears. Select the model and click OK.
Add condition

A condition is added to the path, and the Properties pane for the condition opens. See “Add a Condition” on page 30 for more information.

On the Decision tab, the condition paths are THEN and ELSE clauses. See “Condition Nodes in Each Tab” on page 26 for more information. For more information about adding objects to THEN and ELSE clauses, see “Add a Rule Set or Model” on page 30 and “Add a Condition” on page 30.

---

Reorder Objects in a Decision

On the Decision Flow tab, you can drag rule sets and models from one position to another. To move a condition, you must delete and re-add the condition in the new location.

On the Decision tab, to move an object up or down, including into and out of conditions, select the object and click ☞ or ◄. You can also move the selected object using Shift + ⇧ and Shift + ⬇.

---

Delete an Object from a Decision

On the Decision Flow tab, click ✎ on the object that you want to delete, and select Delete.

On the Decision tab, click ☯ on the object that you want to delete.

---

Edit the Properties of a Decision Node

On the Decision Flow tab:
1. Select the node and click ☛. The Properties pane appears.
2. (Optional) Select a different model, rule set, or rule set version.
3. (Optional) Click ☜. The Input Variables pane appears.
4. (Optional) Modify the variable mappings.
5. (Optional) Click ☞. The Output Variables pane appears.
6. (Optional) Modify the variable mappings.

On the Decision tab:
1. Click ☛ for the node. The properties window for the node appears.
2. (Optional) On the Variables tab, modify the variable mappings.
3. (Optional) On the Properties tab, select a different model, rule set, or rule set version.
4. Click Close to save your changes and close the properties window.

---

Open an Object from within a Decision

On a node in the Decision Flow diagram, click ☛ for the object that you want to open, and select Open.

On the Decision tab, click ☛ for the object that you want to open.

Rule sets open in the rule set editor. Models open in SAS Model Manager if you have access to that application. For condition nodes, SAS Decision Manager displays the property pane for the node.
Duplicate Decisions

Note: You cannot duplicate a decision if it is open.

To duplicate a single decision:
1 In the Decisions view, select the decision that you want to duplicate.
2 Click \( 	ext{Duplicate} \) and select Duplicate. The Duplicate Decision window appears.
3 Enter a new name for the duplicate decision
4 (Optional) Enter a description for the decision.
5 Click Duplicate.

To duplicate multiple decisions:
1 In the Decisions view, select the decisions that you want to duplicate.
2 Click \( 	ext{Duplicate} \) and select Duplicate. SAS Decision Manager duplicates the decisions and appends _Copy to the names of the duplicate copies. If needed, a number is also appended to the names of the duplicate copies.

Move Decisions to a Different Folder

1 In the Decisions category view, select the decisions that you want to move.
2 Click \( 	ext{Move} \) and select Move. The Choose a Location window appears.
3 Select the location to which you want to move the decisions, and click OK.

Delete Decisions

Note: You cannot delete a decision if it is open.

In the Decisions category view, select the decisions that you want to delete, click \( 	ext{Delete} \), and select Delete.

Rename a Decision

Note: You cannot rename a decision if it is open.

1 In the Decisions category view, select the decision that you want to rename.
2 Click \( 	ext{Rename} \) and select Rename. The Rename window appears.
3 Enter a new name for the decision, and click Rename.
Managing Versions of Decisions

Set the Displayed Version

The displayed version is the version whose information is displayed on the other tabs, such as the **Properties** and **Decision** tabs. On the **Versions** tab, a ✓ indicates the displayed version. To change the displayed version, select the version that you want to view, and click **Set Version**.

Create a New Version

**Note**: The current version of an object is the version with the highest version number. When you create a new version, SAS Decision Manager locks the current version before it creates the new version.

**Note**: You cannot save changes to a version that is locked. If you modify a version that is locked and click ✎, SAS Decision Manager asks you if you want to replace the current unlocked version with your edited version.

**Note**: You cannot unlock a locked version.

To create a new version:

1. On the **Versions** tab, click **New Version**. The **New Version** window appears.
2. Select the version type: **Minor** or **Major**. Version numbers follow the format **Major.Minor**. If you select **Major**, the number to the left of the period is incremented. If you select **Minor**, the number to the right of the period is incremented.
3. (Optional) Enter information about the new version in the **Notes** field.
4. Click **Save**.

Testing a Decision

Testing a decision is optional, but it enables you to discover any problems with a decision before it is published and incorporated into a production system.

Create and Run a New Test

1. On the **Tests** tab, click **New Test**. The **New Test** window appears.
2. Enter a name for the test if you do not want to use the default name.
3. (Optional) Enter a description for the test. Descriptions are limited to 1000 characters.
4. Click ✎, select the input table for the test, and click **OK**.
5. Map variables.

SAS Decision Manager automatically maps the input variables in the decision to columns in the input table when the names and data types of the variables match those of the table columns. If any input variables cannot be mapped automatically, an error message is displayed.

**Data source**: *

| HMEQ_TEST | Variables |

**Input variables must be mapped to table columns.** ✗
You can change the automatic variable mappings in the Variable Mappings window.

To change variable mappings:
- Click Variables. The Variable Mappings window appears.
- For each input variable, select the table column to which the variable should be mapped. Alternatively, for Decimal, Integer, and Character variables, you can select Use value for the table column, and specify a literal value in the Value column.
  
  Note: Do not enclose character strings in quotation marks.
- Click OK to close the Variable Mappings window.

6 (Optional) Click Advanced to display the advanced options.
7 (Optional) Click and select the library where you want to write the output of the test.
8 (Optional) Select the version of the decision that you want test.
9 (Optional) Select Preserve unmapped columns in the output table if you want columns that are not mapped to an output variable to be written to the output table.
10 Click Run to run the test. Alternatively, click Save to save the test definition without running it.

The status of the test is indicated by the icon in the Status column.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>🐠</td>
<td>The test is not ready to run. The test definition is not complete, or it might contain errors.</td>
</tr>
<tr>
<td>☑️</td>
<td>The test is defined correctly and is ready to run.</td>
</tr>
<tr>
<td>🎨</td>
<td>The test is running.</td>
</tr>
<tr>
<td>☑️</td>
<td>The test completed successfully.</td>
</tr>
<tr>
<td>🔄</td>
<td>The test completed, but warnings were issued in the SAS log. The URI to the log file is shown on the Test Results page. See Step 11.</td>
</tr>
<tr>
<td>🟥</td>
<td>The test did not run successfully. Check the SAS log for information. The URI to the log file is shown on the Test Results page. See Step 11.</td>
</tr>
</tbody>
</table>

11 Click 📊 in the Results column to view the results of the test. The Test Results page displays information about the test, including the URIs for the test definition and test results. It also includes URIs to the SAS code that was run by SAS Decision Manager, the output data set, and the SAS log that was generated when the code was run.

### Run a Rule-Fired Analysis

If a rule's conditions evaluate to True, then the rule is said to have fired. Rule-fired data includes summary information about how many times each rule fired and detailed information for each time that a rule evaluates to True. See “How Rules Are Evaluated and When Rule-Fired Records Are Generated” on page 9 for more information.

1 Click Rule-Fired Analysis in the navigation pane.
2 Click Run Rule-Fired Analysis. SAS Decision Manager analyzes the test results to determine which rules fired for each row in the input table.

  Note: If a decision contains only models, the rule-fired analysis tables and rule-fired plot are not generated.

3 Click Analysis in the navigation pane. The Analysis page displays the output table at the top of the page and the rule-fired table at the bottom of the page. If you select a row in the output table, the rules that fired for that output row are displayed in the bottom table.

  For example, in the following display there is one output record for which two rules fired.
4 Click **Plot** in the navigation pane. SAS Decision Manager displays a bar chart that shows how many times each rule fired. Position your cursor over a bar to display the name of the rule and the number of times that the rule fired.
Run a Path Tracking Analysis

Decision path tracking shows you the route that input records take through the rule sets, models, and conditions in your decision.

1. Click Decision Path Tracking in the navigation pane.
2. Click Run Path Tracking to run a decision path analysis.
3. Click Analysis and Plot to display a Sankey diagram that shows the flow of the input records through the rule sets, models, and conditions in the decision. The numbers in the diagram are the number of rows in the input table that followed each path.
Note: Nodes that are not executed are shown to the right of the Sankey diagram.

4 Click **Node Count** in the navigation pane to display a table showing the number of input records evaluated at each rule set and model node in the decision.

5 Click **Close** to close the Test Results window.

---

**Working with Test Output Data**

After you run a test, you can work with the output table in other SAS applications to analyze the data, create and compare models, discover relationships hidden in the data, and generate reports based on the data.

Note: The actions available to you depend on the applications that are available at your site.

On the Test Results page, select the **Output table** in the navigation pane, click **Actions**, and select one of the following options:
Prepare Data
opens the output table in SAS Data Studio. SAS Data Studio enables you to perform data transforms such as joining tables, appending data to a table, transposing columns, creating calculated columns, and so on. For more information, see SAS Data Studio: User’s Guide.

Manage Data
opens SAS Data Explorer. SAS Data Explorer enables you to import data, connect to databases, and load tables into memory. For more information, see SAS Data Explorer: User’s Guide.

Explore and Visualize Data
opens the output table in SAS Visual Analytics. SAS Visual Analytics enables you to create, test, and compare models based on the patterns discovered during exploration of the data. You can export the model before or after performing model comparison for use with other SAS products or to put the model into production. SAS Visual Analytics supports a range of visualization, discovery, and reporting features. For more information, see SAS Visual Analytics: Overview.

Build Models
enables you to create a new project in Model Studio using the output table as the data source. SAS Model Studio is an integrated visual environment that provides a suite of analytic data mining tools to facilitate end-to-end data mining analysis. Model Studio is a common interface for SAS Visual Forecasting, SAS Visual Data Mining and Machine Learning, and SAS Visual Text Analytics. For more information, see Model Studio: Getting Oriented.

Explore Lineage
opens SAS Lineage Viewer. SAS Lineage Viewer enables you to better understand the relationships between objects in your SAS Viya applications. These objects include data, transformation processes, reports, and visualizations. For more information, see SAS Lineage Viewer: User’s Guide.

---

Publish a Decision
Publishing content makes it available to other applications. Publishing a decision creates an entity that can be managed and run in another environment. For example, if you publish content to the SAS Micro Analytic Service destination, SAS Decision Manager creates a DS2 package that can be managed and run through the SAS Micro Analytic Service interface.

Note: The publishing destinations that are available are determined by your system administrator.

To publish a decision:
1. Open the decision that you want to publish. The version that is published is the displayed version.
2. (Optional) Change the displayed version:
   a. Click the Versions tab.
   b. Select the version that you want to publish, and click Set Version.
3. Click Publish. The Publish Decisions window appears.
4. (Optional) If you have previously published the decision, expand the Items to Publish section, and select the check box in the Replace column.
5. Select the destination to which you want to publish, and click Publish. The Publishing Results window appears. It displays the published name of the object, its status, and information about any issues encountered while publishing.

   Note: The publishing destinations that are available to you depend what is configured at your site. See “Configuring Publishing Destinations” in SAS Decision Manager: Administrator’s Guide for more information.
6. After the status changes to Published successfully, click Close to close the Publishing_Results window.
7. Click Close to close the decision.
About Lookup Tables and Functions

SAS Decision Manager provides the ability to import lookup tables and reference them from rules. Lookup tables are tables of key-value pairs. For example, you can use a lookup table to retrieve a part name based on the code number of the part or to retrieve the full name for a country based on its abbreviation.

You can import lookup data from comma-separated-values (CSV) files, such as those created by spreadsheet applications, into lookup tables in SAS Decision Manager. You can re-import updated CSV files as needed to refresh the lookup tables.

Note: SAS Decision Manager does not support CSV files that contain signature lines.
In a lookup table, each lookup key is associated with a lookup value. Lookup keys must be unique within each lookup table.

SAS Decision Manager provides two functions, LOOKUP and LOOKUPVALUE, that enable you to determine whether a lookup key exists in a lookup table and to retrieve a lookup value from a lookup table.

Create a New Lookup Table

1. Click on the navigation bar.
2. Click New Lookup Table. The New Lookup Table window appears.
3. Enter a name for the new lookup table. Lookup table names are limited to 250 characters. Lookup table names are case insensitive and must be unique within the database.
4. (Optional) Enter a description for the new lookup table. Descriptions are limited to 1000 characters.

   **TIP** To modify this description at any time, click next to the Description field on the Properties tab.

5. Click , select the folder where you want to create the new lookup table, and click Save. The application opens the new lookup table and displays the Lookup Table tab.
6. Add entries to the new table either by importing a CSV file or by adding entries manually. See "Import or Refresh Lookup Table Entries" on page 42 and "Add Lookup Table Entries" on page 43 for more information.

Import or Refresh Lookup Table Entries

**Note:** It is recommended that a single lookup table contains no more than 5000 entries.

You can import entries into an empty table, and you can refresh an existing lookup table by re-importing the same table.

1. Open the lookup table to which you want to import entries.
2. Click Import. The Import Lookup Table window appears.
3. Click , and select the CSV file that contains the lookup table entries.
4. Select the encoding for the lookup table, and click Import.
Export a Lookup Table

To export a lookup table, open the lookup table and click Export. The table is exported into a CSV file, and a notification appears in the download bar at the bottom of the browser window.

Add Lookup Table Entries

Note: It is recommended that a single lookup table contains no more than 5000 entries.

1. Open the lookup table to which you want to add entries.
2. Click New Entries if the lookup table is empty, or click + if the lookup table already contains entries. The Add Table Entries window appears.
3. Enter the lookup key name and value for the new entry. Key names and lookup values are each limited to 100 characters. Key names must be unique within the same lookup table.
   - To add additional entries, click Add an entry, and enter the new key name and value.
4. Click Save to save the new entries and close the Add Table Entries window.

Edit Lookup Table Entries

1. Open the lookup table.
2. Select the entries that you want to edit, and click . The Edit Table Entries window appears.
3. Edit the exiting entries, and click Save.

Delete Lookup Table Entries

1. Open the lookup table.
2. Select the entries that you want to delete and click .

Duplicate Lookup Tables

Note: If you duplicate a lookup table that has an active version, the duplicate table is automatically activated.

To duplicate a single lookup table:
1. Select the table that you want to duplicate, click , and select Duplicate.
2. Enter a new name for the duplicate lookup table. Names are limited to 250 characters. Lookup table names are case insensitive and must be unique within the database.
3. (Optional) Enter a description for the duplicate table. Descriptions are limited to 1000 characters.
4. Click Duplicate.
To duplicate multiple lookup tables, select the tables that you want to duplicate, click : , and select **Duplicate**. SAS Decision Manager appends an -Copy and a number (if needed) to the duplicate table names.

---

### Move Lookup Tables

1. Select the tables that you want to move, click : , and select **Move**. The Choose a Location window appears.
2. Select the folder where you want to move the tables, and click **OK**.

### Rename a Lookup Table

1. Select the table that you want to rename, click : , and select **Rename**. The Rename window appears.
2. Enter a new name for the table, and click **Rename**.

### Delete Lookup Tables

Select the tables that you want to delete, click : , and select **Delete**.

*Note:* You cannot delete a lookup table that is referenced in a rule set or decision.

---

### Managing Versions of Lookup Tables

Creating a new version of a lookup table does not lock the previous version. The only way to lock a lookup table is to activate it.

### Set the Displayed Version

The displayed version is the version whose information is displayed on the **Lookup Table** tab. On the **Versions** tab, a ✓ indicates the displayed version. To change the displayed version, select the version that you want to view, and click **Set Version**.

### Create a New Version

*Note:* The current version of an object is the version with the highest version number. When you create a new version, SAS Decision Manager locks the current version before it creates the new version.

*Note:* You cannot save changes to a version that is locked. If you modify a version that is locked and click ❌, SAS Decision Manager asks you if you want to replace the current unlocked version with your edited version.

*Note:* You cannot unlock a locked version.

To create a new version:
2 Select the version type: **Minor** or **Major**. Version numbers follow the format *Major.Minor*. If you select **Major**, the number to the left of the period is incremented. If you select **Minor**, the number to the right of the period is incremented.

3 (Optional) Enter information about the new version in the **Notes** field.

**TIP** You can edit these notes at any time on the **Versions** tab.

4 Click **Save**.

---

### Activate a Lookup Table

The active version of a lookup table is the version that is used when the lookup table is referenced by running processes. Activating a version of a lookup table locks that version. Locked tables cannot be unlocked.

You must activate a lookup table before you can use it in a rule.

1 Open the lookup table that you want to activate.
2 On the **Versions** tab, select the version that you want to activate.
3 Click **Activate**.

---

### LOOKUP Function

Determines whether a lookup key exists in a lookup table.

**Restrictions:**
- You can specify the LOOKUP function in condition expressions only.
- If an expression contains the LOOKUP function, then the expression cannot contain anything else.

**Returned data type:**
- Boolean

**Syntax**

```
LOOKUP ('lookup_table_name', variable)
```

- **lookup_table_name** specifies the name of the lookup table that you want to search.
- **variable** specifies the variable that contains a lookup key value.

**Example**

Suppose you have a Country_Codes lookup table that uses two-letter abbreviations for countries as the lookup key and country names as the lookup values.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU</td>
<td>Australia</td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
</tr>
<tr>
<td>CR</td>
<td>Costa Rica</td>
</tr>
</tbody>
</table>
To verify that the value of the variable Cntry_Key exists as a lookup key in the table Country_Codes, you can use the following expression:

```
LOOKUP('Country_Codes', Cntry_Key)
```

If the value of Cntry_Key exists as a lookup key, the LOOKUP function returns the value `True`.

In the following rule, if the key specified by the variable Cntry_Key exists in the lookup table Country_Codes, then the value that is associated with that key is assigned to the variable Country_Name.

```
If
  LOOKUP('Country_Codes', Cntry_Key)
Then
  Assign
  Country_Name
  LOOKUPValue('Country_Codes', Cntry_Key)
```

---

**LOOKUPVALUE Function**

Retrieves a lookup value from a lookup table.

**Restrictions:**
- You can specify the LOOKUPVALUE function only in action expressions.
- If an expression contains the LOOKUPVALUE function, then the expression cannot contain anything else.

**Returned data type:**
- Lookup tables are stored as character data. However, you can assign the results of the LOOKUPVALUE function to the following types of variables: Character, Integer, Decimal, Date, Datetime, or Boolean. The LOOKUPVALUE function converts the results to match the type of the variable.

**Syntax**

```
LOOKUP ('lookup_table_name', variable)
```

- `lookup_table_name` specifies the name of the lookup table that you want to search.
- `variable` specifies the lookup key for the value that you want to retrieve.

**Example**

Suppose you have a Country_Codes lookup table that uses two-letter abbreviations for countries as the lookup key and country names as the lookup values. The Country_Codes lookup table contains the lookup key `CA`, and the lookup value that corresponds to that key is `Canada`.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AU</td>
<td>Australia</td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
</tr>
<tr>
<td>CR</td>
<td>Costa Rica</td>
</tr>
</tbody>
</table>

If the Cntry_Key variable in the current input record contains the value `CA`, you can use the following expression to retrieve the lookup value that is associated with that key from the table Country_Codes:
LOOKUPVALUE('Country_Codes', Cntry_Key)

In the following rule, if the key specified by the variable Cntry_Key exists in the lookup table Country_Codes, then the value that is associated with that key is assigned to the variable Country_Name.

If Lookup("Country_Codes", Cntry_Key)

Then Assign Country_Name LookupValue("Country_Codes", Cntry_Key)