

## Variables

Last Modified on 05.05.26

Here you can set global variables, at zone level, within Application Workspace. These variables can be used within packages, to simplify the use of common paths or parameters across multiple packages. For more information about inheritance and available system parameters, see [Variables](#).

### Variable list



Two types of variables are available:

- Static – used for fixed values, which can still be overridden by contexts or packages.
- Dynamic – allows you to select a credential or license which will be used as the value of the variable.  
For example, with credentials selected as entity type, passwords or usernames are stored encrypted within the Application Workspace System. Application Workspace will ask only once for a password or username and stores this information within the Credential Store for that user. Because of this, the credential variable is reusable within multiple packages.

#### Selecting multiple table rows


- Adjacent rows: Click on the first row, then hold down the Shift key and click on the last row in the range
- Non-adjacent rows: Hold down the Ctrl key (or "Command" on Mac) and click on each row you want to select
- Adjacent rows, using only the mouse: Click and hold the left mouse button on the first row, then drag the cursor up or down across the rows to highlight the desired ones

### Table toolbar

To create a new variable, click on the  **Create** button; to view the details of a variable or edit it, double-click its entry or select the variable and click the  **Edit** button.

 **Views** gives you control on how the table is displayed.

The default view contains all your variables in alphabetic ascending order, and all columns **Name**, **Type**, **Sensitive** and **Value**.

You can create your own personalized view of the table, by filtering the variables or adding/deleting columns and clicking **Save as** in the drop-down menu of  **Views**.

### Create a variable dialog box

The **Name** of the variable, can be used within packages by enclosing its name within `${*name*}`.

The **Sensitive** option lets you decide if the variable should be logged in the log file and displayed on Application Workspace UI.

---