



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Designer User's Guide

Debug Block

Contents

- 1 Debug tab
 - 1.1 Capture Checkpoint
- 2 Data tab
- 3 Advanced tab



- Administrator

Use this block to debug runtime logic that isn't running as intended.

Related documentation:

-
-
-

You can use this block during the application development phase to define a specific checkpoint in a module or application. This is useful when you want to debug runtime logic that isn't running as intended.

For example, you might add some ECMAScript expressions to an Assign Variables block to assign values to certain variables, but discover that the logic isn't producing the desired result. To assist in debugging this, you could add a **Debug** block immediately after the Assign Variables block to capture the values of the variables as they exist at that time. These values can then be reviewed in Designer Analytics, under the debugcheckpoints property of the Session Detail Record (SDR).

Important

The **Debug** block is only processed in the DEV, QA, and UAT application streams. It is ignored in the LIVE stream.

Debug tab

Capture Checkpoint

Select this option to enable debugging for a specified checkpoint.

Checkpoint Name

Specify the name of the checkpoint to be captured. You can also specify a variable that holds the value of the checkpoint name.

Condition

Specify the condition as a script expression. For example:

```
var02 === 1
```

Data tab

This tab displays a list of variables that can be captured by the checkpoint. Select the variables that you want to include.

Advanced tab

Enable the **Write these statements to platform logs** option if you want to write the results of the specified ECMAScript expressions to platform logs that can be reviewed by Genesys support. Click **Add Log** to enter the ECMAScripts.

You can also specify an error message statement to add to the logs if the expression evaluations result in an error.