



Visual Workflow Troubleshooting Guide

Version: 2022.1.0

Copyright AppViewX, Inc.

Copyright © 2022 AppViewX, Inc. All Rights Reserved.

This document may not be copied, disclosed, transferred, or modified without the prior written consent of AppViewX, Inc. While all content is believed to be correct at the time of publication, it is provided as general-purpose information. The content is subject to change without notice and is provided “as is” and with no expressed or implied warranties whatsoever, including, but not limited to, a warranty for accuracy made by AppViewX. The software described in this document is provided under written license only, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. Unauthorized use of software or its documentation can result in civil damages and criminal prosecution.

Trademarks

The trademarks, logos, and service marks displayed in this manual are the property of AppViewX or other third parties. Users are not permitted to use these marks without the prior written consent of AppViewX or such third party which may own the mark.

External Reference Links

This product includes software developed by the CentOS Project (www.centos.org).

This product includes software developed by Red Hat, Inc. (www.redhat.com).

This product includes software developed by VMware, Inc. (www.vmware.com).

All other trademarks mentioned in this document are the property of their respective owners.

Contact Information

AppViewX, Inc.

222 Broadway, FL 19

New York, NY 10038

Email: info@appviewx.com

Web: www.appviewx.com

Contents

Preface.....	v
Revision History.....	v
About this Guide.....	v
Audience.....	v
Text Conventions.....	v
Chapter 1. Prerequisites.....	6
Web Browser Requirement.....	6
Chapter 2. Issues in Importing a Workflow.....	7
Importing a Workflow.....	7
Workflow version already exists.....	8
Invalid workflow format.....	8
Version limit exceeded.....	9
Chapter 3. Issues in Exporting a workflow.....	10
Chapter 4. Issues in Creating a Workflow.....	11
Overview.....	11
Steps to Create a Workflow.....	11
Adding Versions in a Workflow.....	12
Adding a task	13
Steps to add a task.....	14
Updating a task.....	14
Chapter 5. Issues in User Interface Task.....	16
Overview.....	16
Access Denied Issue in Form Task.....	16
Global Variable Issue in Form Task.....	17
Access Denied Issue in User Interface Tasks.....	18
Chapter 6. Issues in Workflow Validation.....	19
Workflow Validation.....	19

Illegal flow.....	20
Broken flow.....	21
Subflow validation.....	21
Chapter 7. Helper Script.....	22
Overview.....	22
Creating a helper script.....	22
Importing a helper script.....	24
Updating a helper script.....	25
Chapter 8. Hooks.....	28
Overview.....	28
Query explorer Hook.....	28
Creating a Query explorer Hook.....	28
Script Hook.....	30
Creating a Script Hook.....	30
REST Hook.....	31
Creating a REST Hook.....	32
Exporting a Hook.....	32
Importing a Hook.....	33
Chapter 9. Issues in Creating a Workflow Request.....	35
Overview.....	35
Steps to create a request.....	35
Submitting a user task.....	35
Submitting an executed request.....	36
Reflection of workflow data updations in a request.....	36
Form task in a request.....	36
Data manipulation in a request.....	37
Command task in a request	37

Preface

Revision History

Revision	Description	Date
1.0	Initial release of document for Release 2022.1.0	June 2022

About this Guide

This document helps you troubleshoot the common problems that you might encounter when using Visual Workflow functionalities.

Audience

This guide is useful for experienced users with reasonable programming experience (Python), trying to enhance their skills with automating network tasks. You can use AppViewX pre-built workflow tasks, commands, API, SDK to automate across the L2-L7 layer, such as automate load balancers, DNS, WAF, PKI, switches, routers, and other network elements.

Text Conventions

The following text conventions are used in this document:

Convention	Description
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in the text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>codeblock</code>	Indicates commands with a paragraph, URLs, codes in examples, text that appears on the screen, or text that you enter.

Chapter 1: Prerequisites


Web Browser Requirement

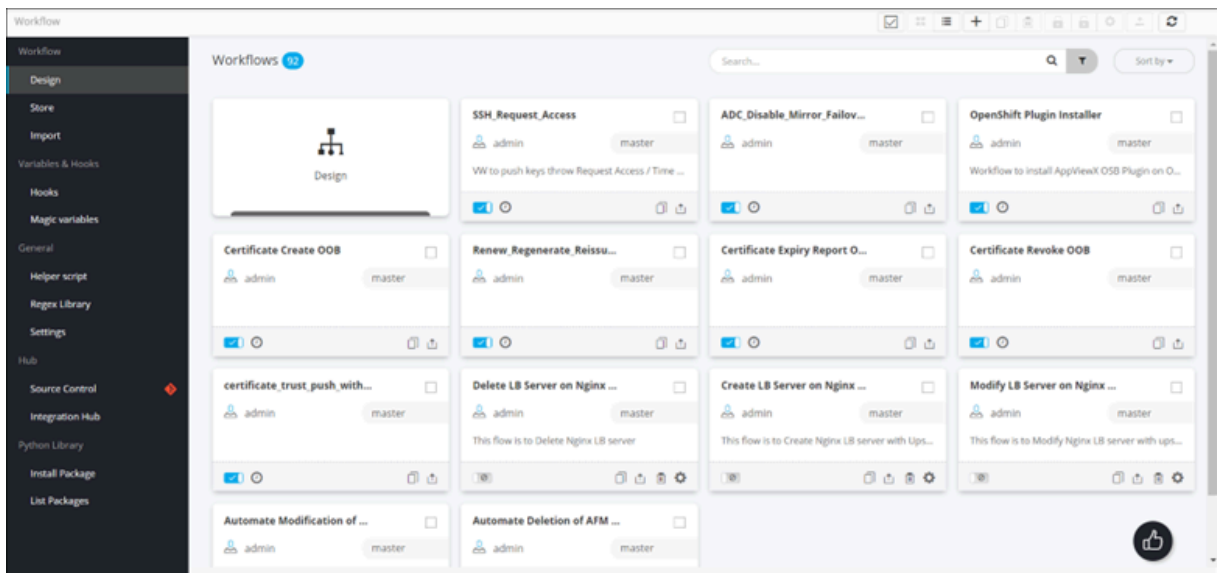
Browser	Version	Notes
Firefox	Till latest (Version 84.0.4147.135)	
Chrome	Till latest (Version 80.0)	
IE	Limited support in 9, Full support from 10+	No support for IE9 post AppViewX Version 11.0
Safari	Till latest (Windows - Version 5.1.7, macOS - Version 13.1.2)	From AppViewX Version 11.1
Opera	Till latest (Version 70)	From AppViewX Version 11.1

Chapter 2: Issues in Importing a Workflow

- Importing a Workflow
- Workflow version already exists
- Invalid workflow format
- Version limit exceeded

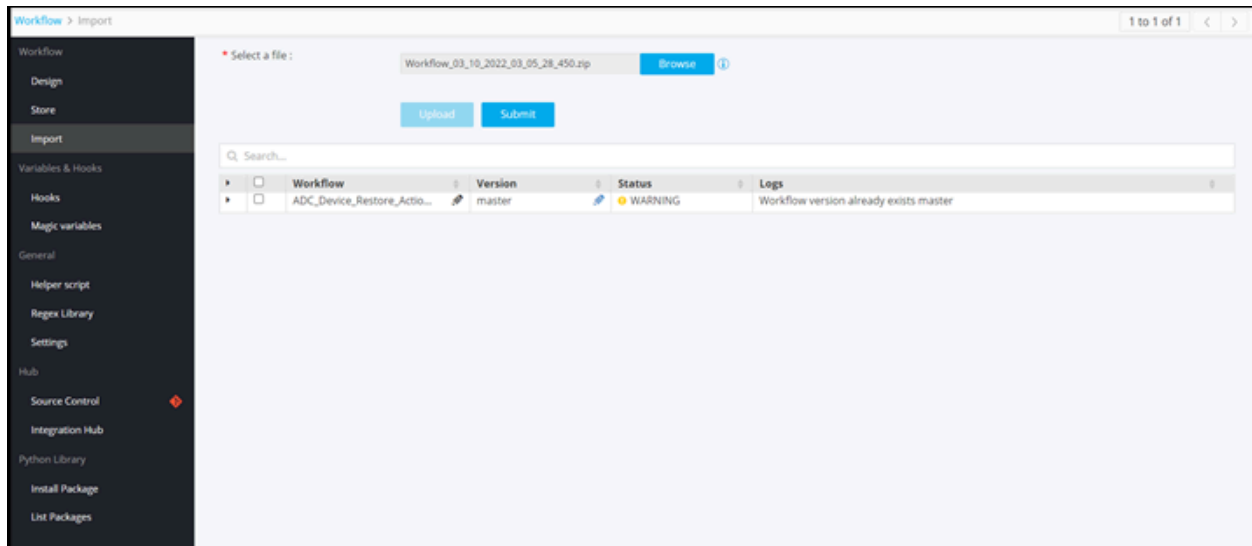
Importing a Workflow

1. To access the navigation pane, hover the mouse over the  icon.
2. From the menu displayed, select **Studio > Workflow**.



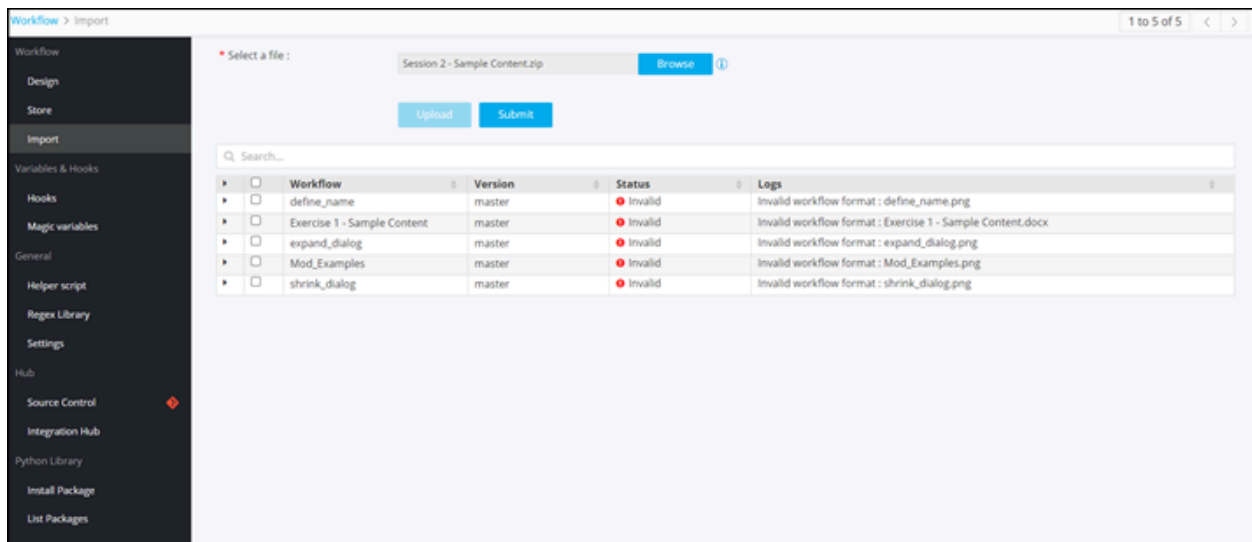
3. From the navigation pane on the left, click **Import**.
4. Click **Browse**.
5. Select a file with .zip format
6. Click **Upload**.

Workflow version already exists



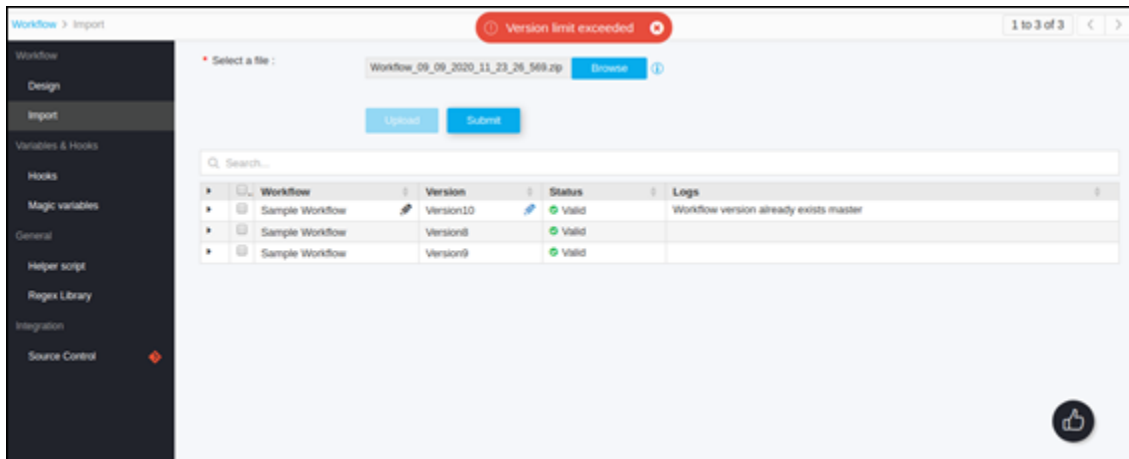
Error Message	Possible Cause	Possible Solution
Workflow version already exists.	Workflow imported with the version which is already present.	Provide some other name for the version.

Invalid workflow format






Error Message	Possible Cause	Possible Solution
Invalid workflow format	The file contains invalid entries and it cannot be parsed.	<p>Choose an import file that has valid workflow details.</p> <ol style="list-style-type: none"> 1. Navigate to Studio > Workflow. 2. Click Import. 3. To select a valid import file, click Browse. 4. Click Upload. <p>The workflow details parsed should display the status as valid, indicating valid import entries.</p> <ol style="list-style-type: none"> 5. Click Submit.

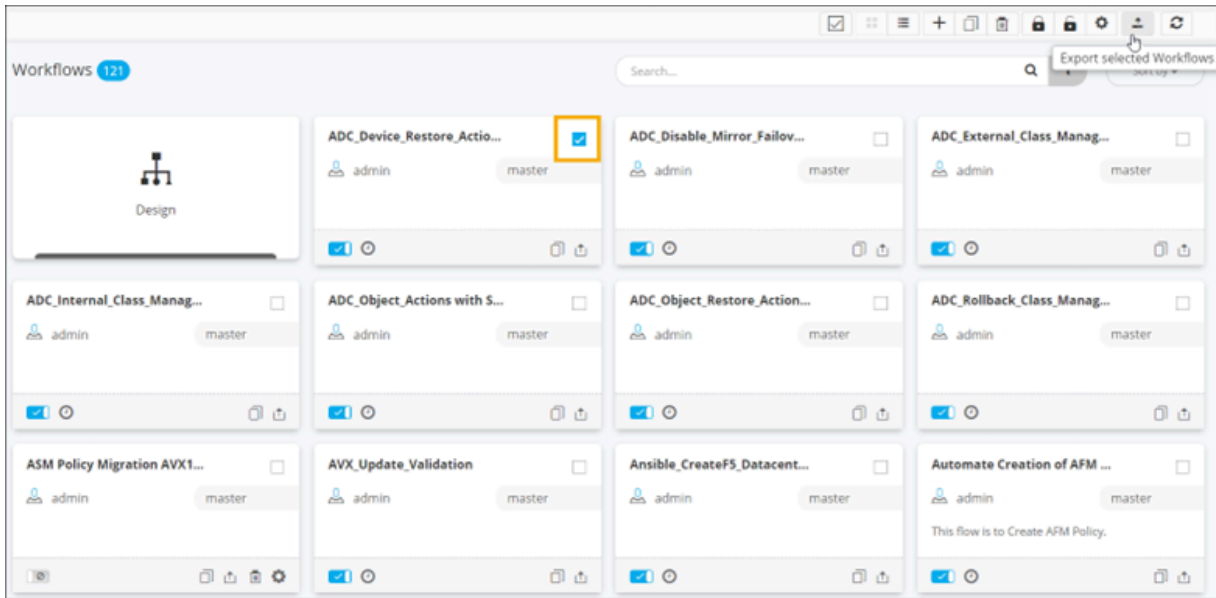
Version limit exceeded



Error Message	Possible Cause	Possible Solution
Version limit exceeded	A workflow can have only 10 versions.	Check if the workflow has more than 10 versions and delete the unrequired versions.

Chapter 3: Issues in Exporting a workflow

1. To access the navigation pane, hover the mouse over the  icon.
2. From the menu displayed, select **Studio > Workflow**.
3. To select all workflows visible on the page, from the top right corner of the screen, click .
4. To export the selected workflows, from the top right corner of the screen, click .



Error Message	Possible Cause	Possible Solution
Cannot export more than 25 workflows at once	Only 25 workflows can be exported at a time	Select only 25 workflows to export


Chapter 4: Issues in Creating a Workflow

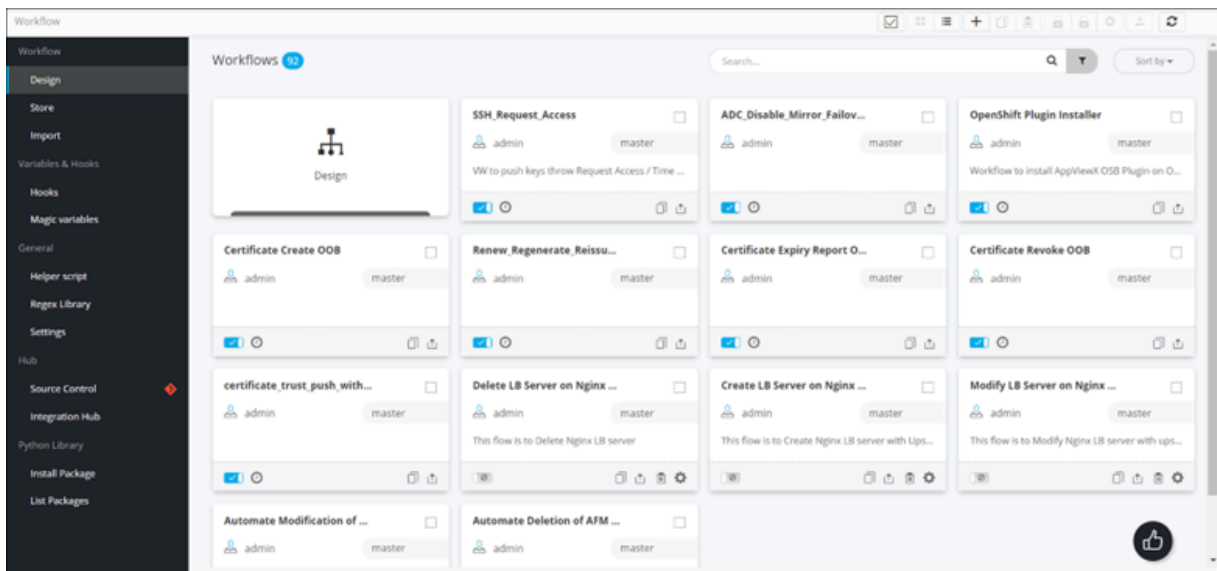
- Overview
- Steps to Create a Workflow
- Adding Versions in a Workflow
- Adding a task
- Updating a task

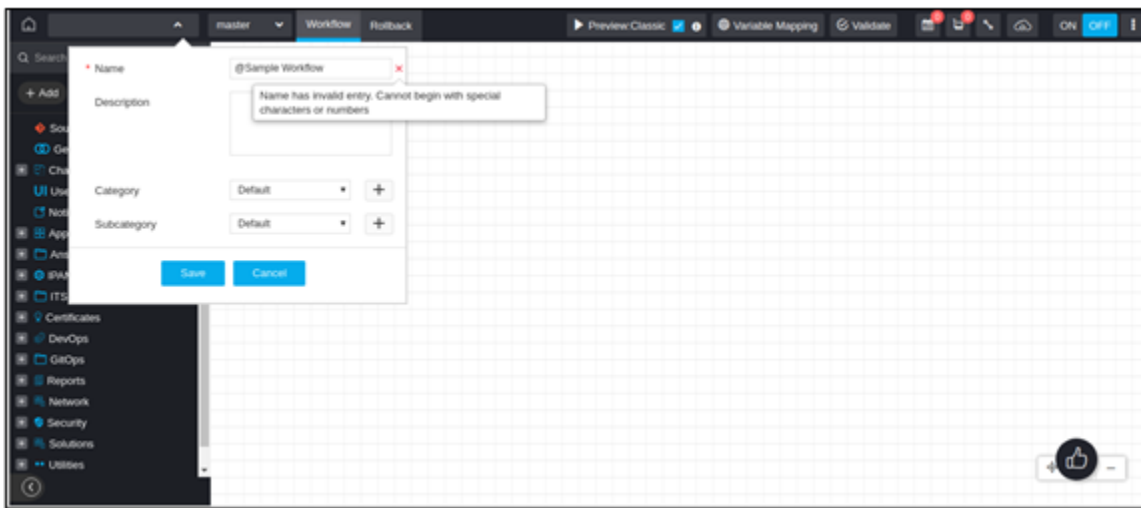
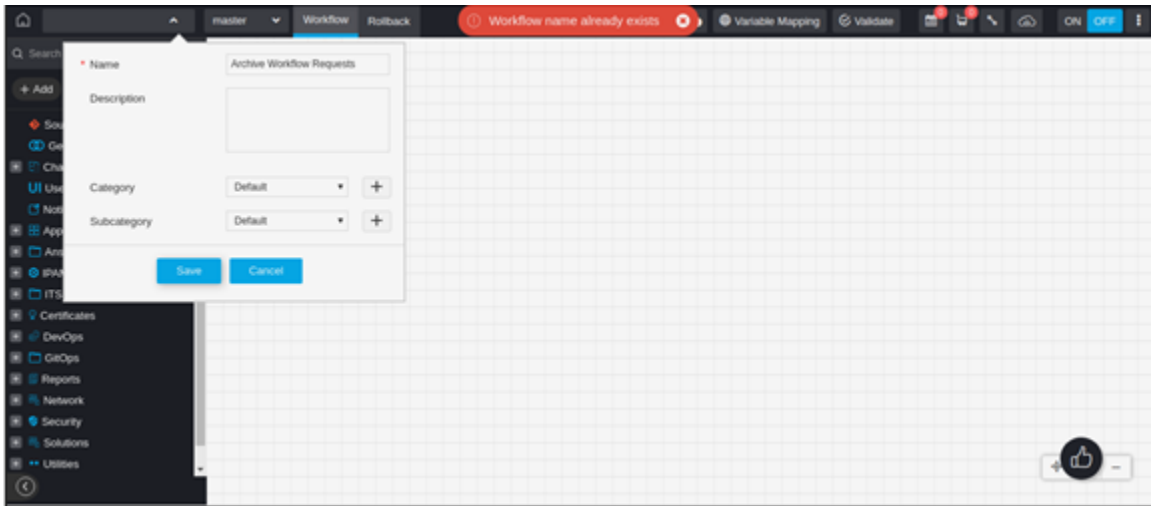
Overview

- A placeholder for users to design simple and complex workflow(s).
- Design simple, complex automation workflows; and view the inventory of all workflows by category.

Steps to Create a Workflow


1. Log into AppViewX.
2. To access the navigation pane, hover the mouse over the  icon.
3. From the menu displayed, select **Studio** > **Workflow**.





Error Message	Possible Cause	Possible Solution
Workflow name already exists	Workflow name should be unique	Provide some other name
Name has invalid entry	Only few special characters are allowed	Allowed characters are '-', '_', '*', ':', ' ' and space

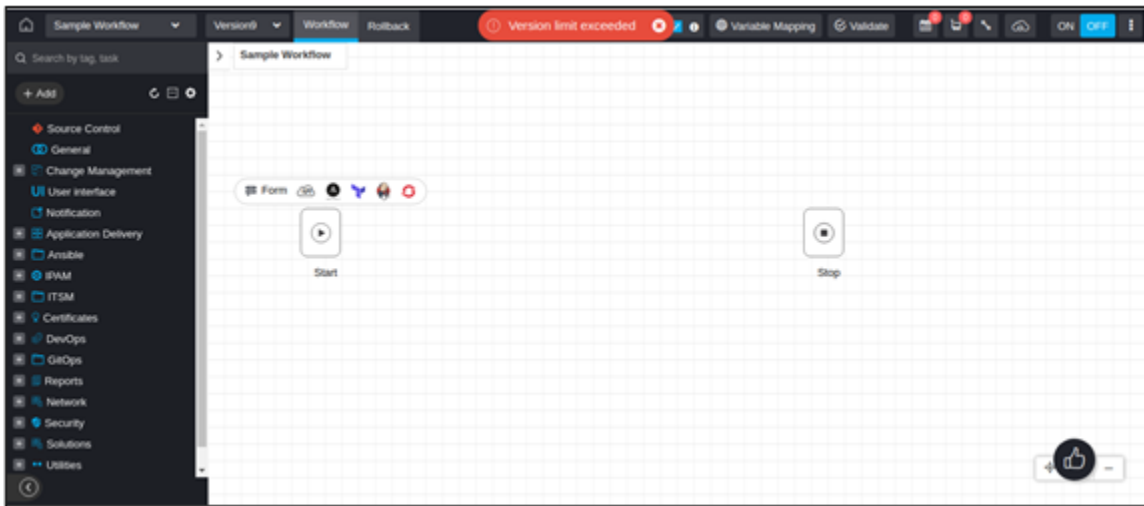
Adding Versions in a Workflow

1. Log into AppViewX.
2. To access the navigation pane, hover the mouse over the  icon.

3. From the menu displayed, select **Studio > Workflow**.
4. Create a workflow.



Note: By default, the created workflow will have a master version. You can create new versions for the workflow.




Error Message	Possible Cause	Possible Solution
Version limit exceeded	A workflow can have only 10 versions.	Create a workflow with 10 versions

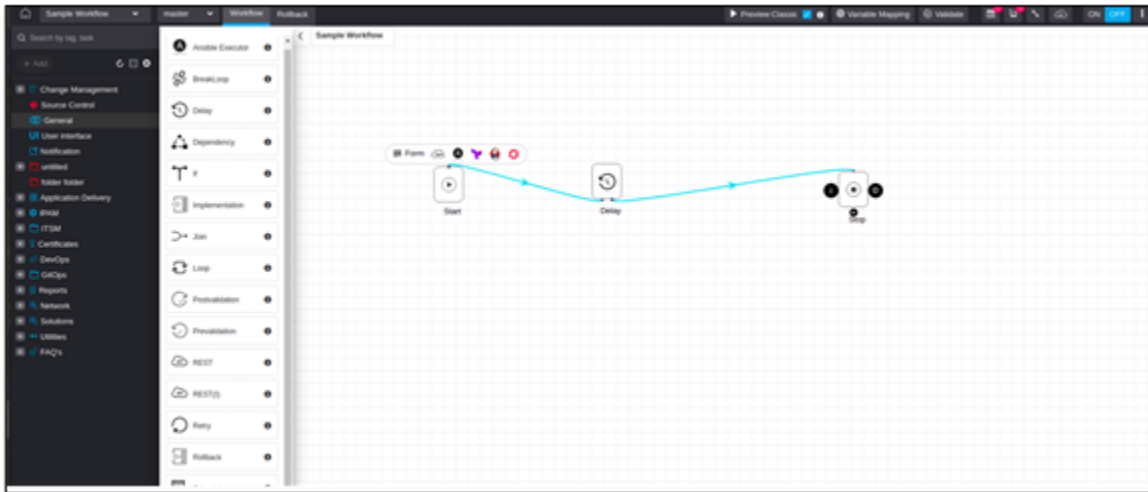
Adding a task

A workflow palette or task is an independent business logic block required to automate and stitch together a workflow.

- Tasks are objective workflow elements required to automate and design a workflow
- Tasks can have custom and/or pre-defined business logic to aid in workflow automation
- Tasks can be defined as either: a) User task b) Automated task
- Tasks can be cloned and reused with the workflow
- Tasks will have an Input/output (I/O operation) to pass data (in the form of variables) from one stage of a workflow to another.
- [Steps to add a task](#)

Steps to add a task

1. To access the navigation pane, hover the mouse over the  icon.
2. From the menu displayed, select **Studio > Workflow**.
3. Design a workflow.
4. From the **General** section, drag and drop any task.



Error Message	Possible Cause	Possible Solution
Invalid Task component details	When an api is used to drag and drop a task,the payload might be wrong.	Drag and drop a task from ui and check for the payload.
Workflow not found	Workflow details might be missing in the database.	Check whether the workflow details exist in the database.
Workflow Components detail not found	Workflow component details might be missing in the database.	Check whether the workflow details exist in the database.
Link Id is mandatory to perform the action	Task id might be missing.	Check whether the task id is available for the task.

Updating a task

1. Design a workflow.
2. Drag and drop the tasks for the workflow.
3. Enter the required field information for a task.
4. Click **Save**.

5. To edit/modify the task configurations, double click on the task.

Error Message	Possible Cause	Possible Solution
Workflow Detail not found	Workflow details might be missing in the database	Check whether the workflow details exist in the database

Chapter 5: Issues in User Interface Task

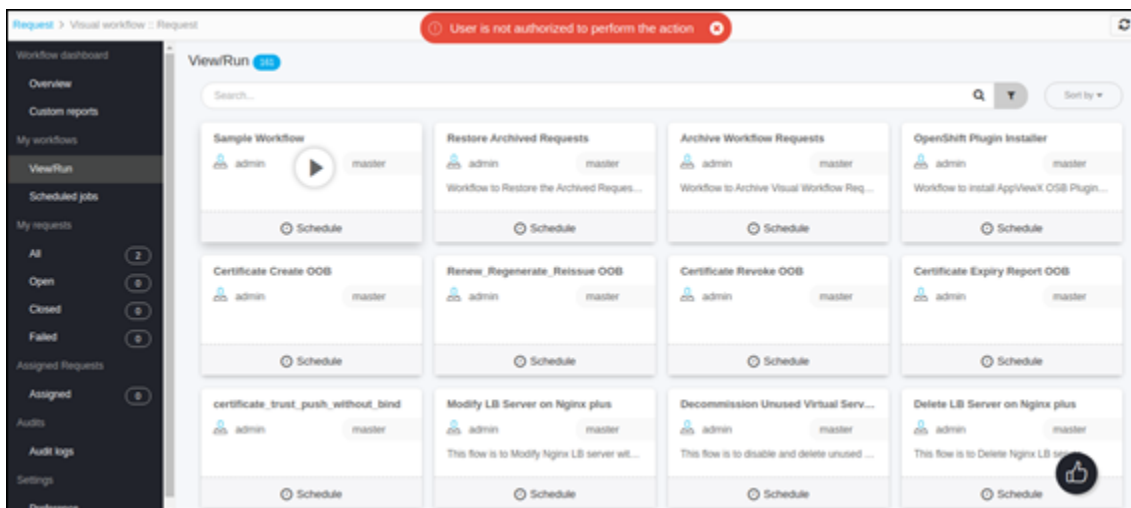
- Overview
- Access Denied Issue in Form Task
- Global Variable Issue in Form Task
- Access Denied Issue in User Interface Tasks

Overview

The User Interface section comprises tasks that will need a certain amount of user intervention.

Access Denied Issue in Form Task

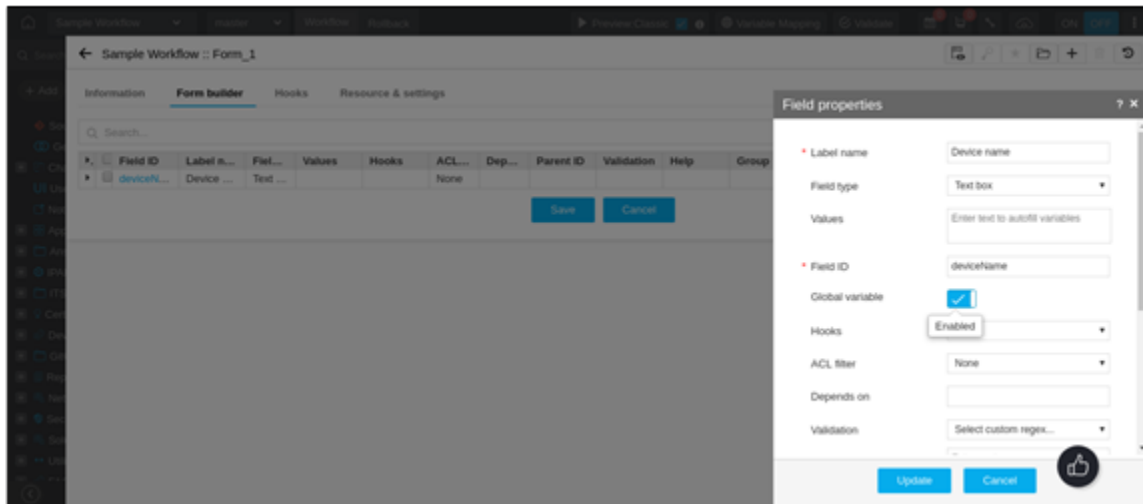
1. Design a workflow.
2. From the **User Interface** section, drag and drop a **Form** task.
3. Define the form fields.
4. Click **Save**.
5. Enable and trigger the workflow.



Error Message	Possible Cause	Possible Solution
User is not authorized to perform the action	The logged in resource might not have access to the Form task.	a. Open the task and click Resource & Settings. b. Under Resource & Settings, click Request Resource. c. Give submitter access to the logged in resource.

Global Variable Issue in Form Task

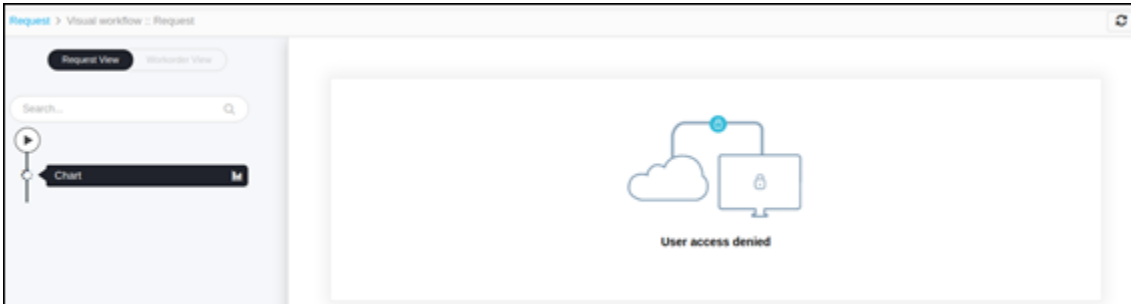
1. Design a workflow.
2. From the **User Interface** section, drag and drop a **Form** task.
3. Define the form fields.
4. Assign/define required values as Global variables. Once declared, reference the global variables across any other stage using the following syntax: `<%variable%>`
5. Enable and trigger the workflow.



Error Message	Possible Cause	Possible Solution
The referenced global variable will be null	The global variable toggle might not be enabled when adding the field in the Form task.	Enable the global variable toggle in the Form task to reference it across the workflow.

Access Denied Issue in User Interface Tasks

1. Design a workflow.
2. From the **User Interface** section, drag and drop any task other than Form task, for example, the **Chart** task.
3. Define the task fields.
4. Click **Save**.
5. Trigger the workflow.



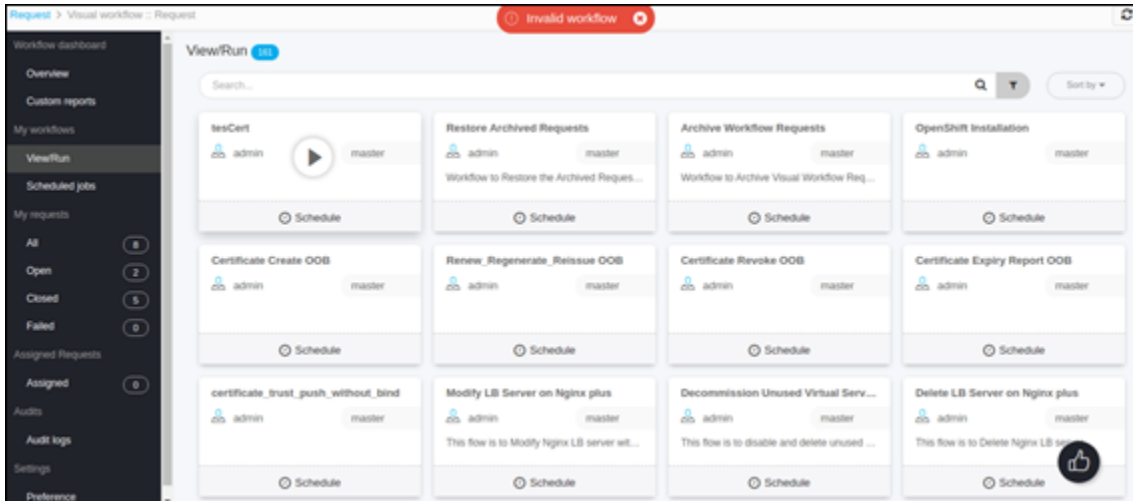
Error Message	Possible Cause	Possible Solution
User access denied	The logged in resource might not have access to the user interface task.	<ol style="list-style-type: none"> a. Open the task and click Resource & Settings. b. Under Resource & Settings, click Request Resource. c. Give submitter access to the logged in resource.

Chapter 6: Issues in Workflow Validation

- Workflow Validation
- Illegal flow
- Broken flow
- Subflow validation

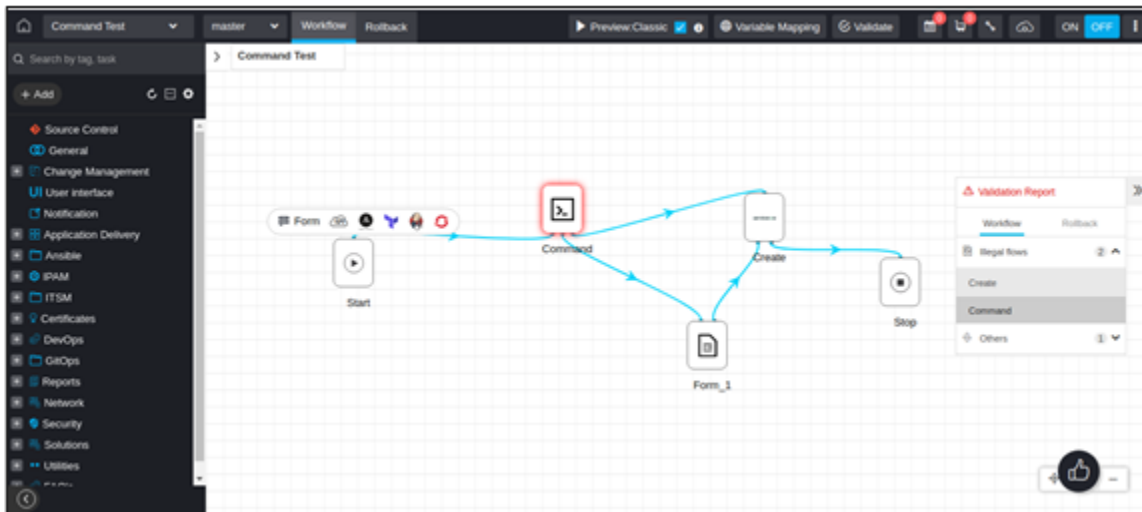
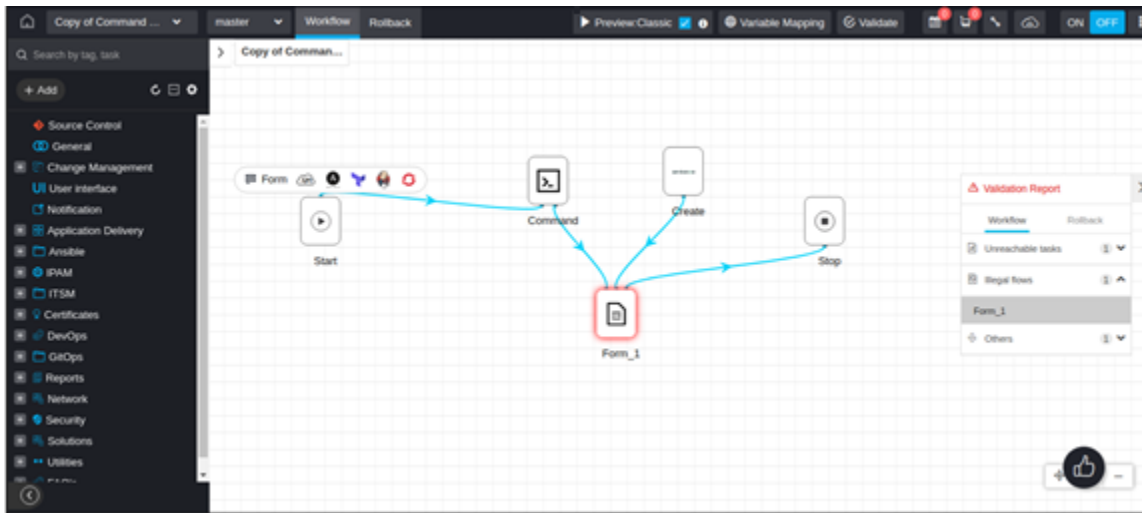
Workflow Validation

1. Design a workflow.
2. Add the workflow tasks.
3. Enable and trigger the workflow.



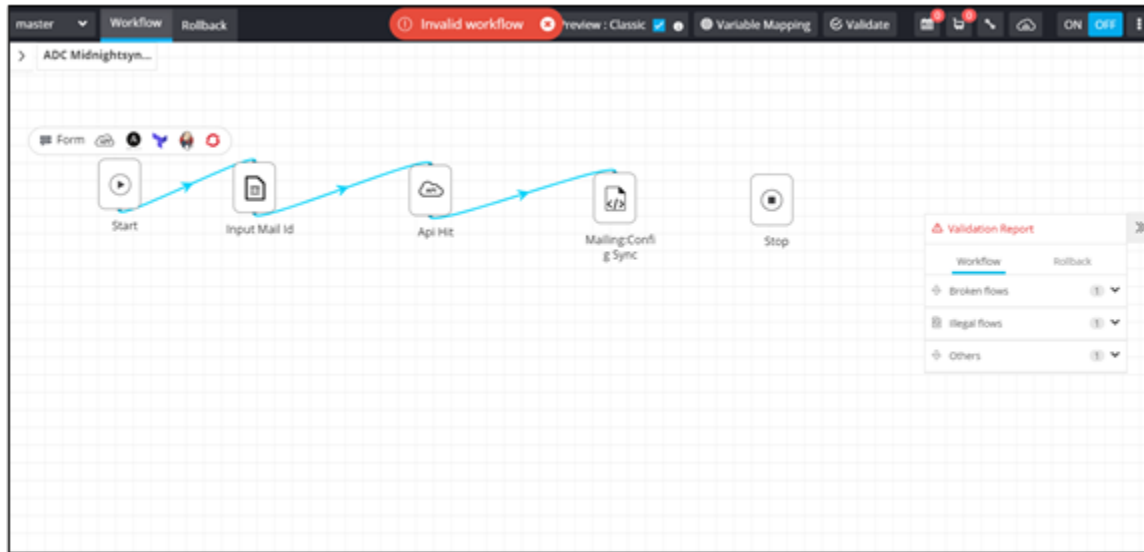
Error Message	Possible Cause	Possible Solution
Invalid Workflow	There might be an issue in the workflow	Avoid run time errors by validating the workflow

Illegal flow



Error Message	Possible Cause	Possible Solution
Invalid workflow - Illegal flow	Multiple task outputs are given as an input to a single task.	Use join task to provide multiple inputs to a task.
Invalid workflow - Illegal flow	When a single task output is connected to multiple tasks.	Use split task to connect single task output to multiple tasks.

Broken flow



Error Message	Possible Cause	Possible Solution
Invalid workflow - broken flow	When a task is not connected to any other tasks.	Check whether all the tasks are connected.

Subflow validation

Nested workflow allows users to reuse existing workflows and automate quickly. Users can design new workflow(s) and/or import workflow(s) under a parent workflow.

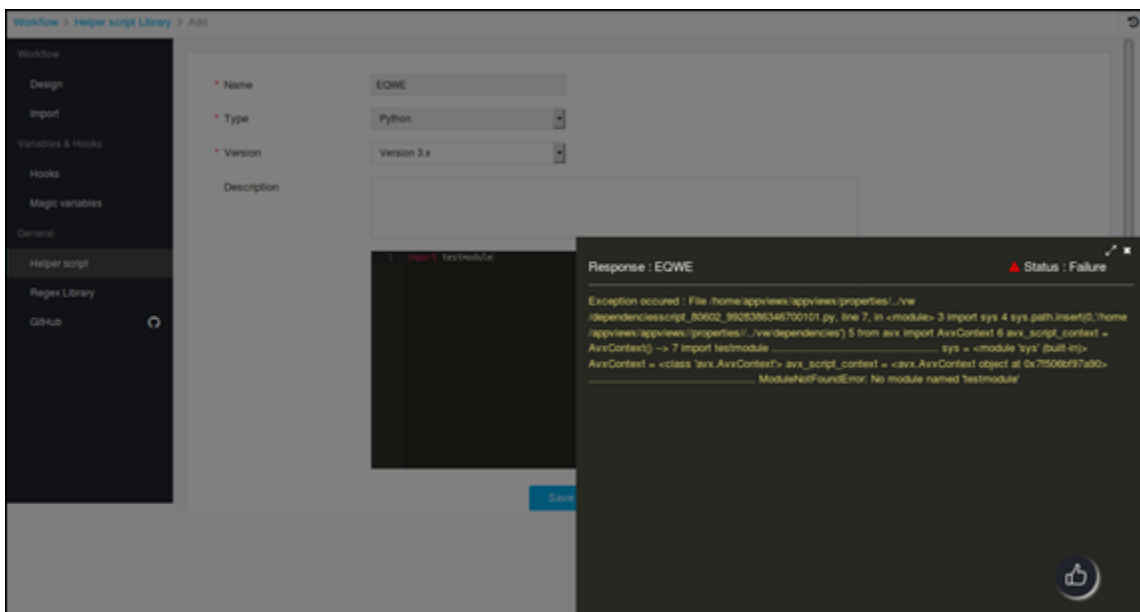
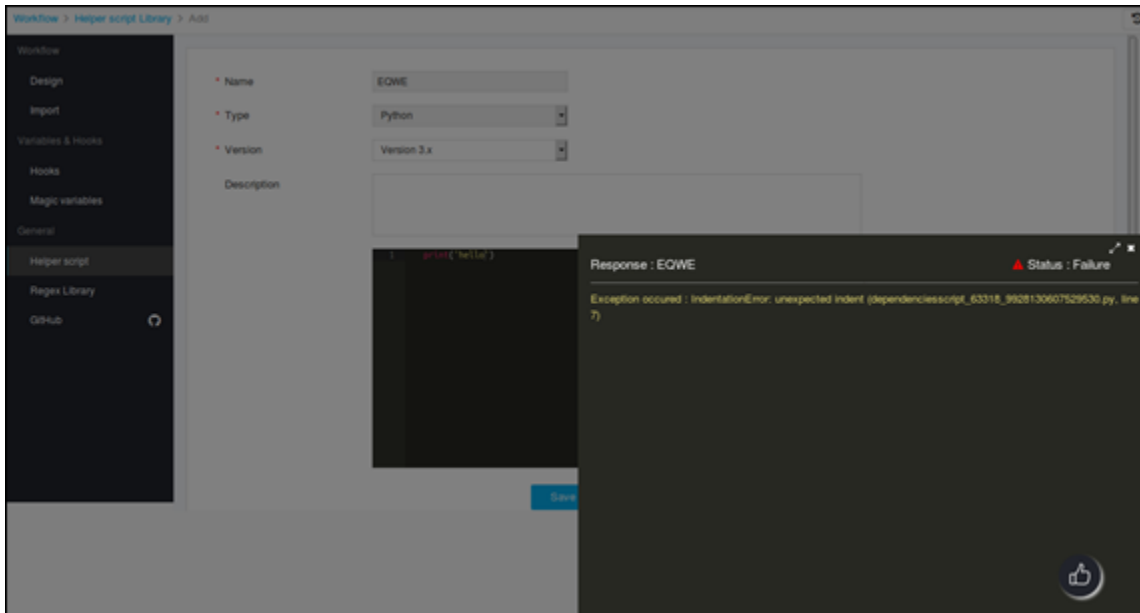
Error Message	Possible Cause	Possible Solution
Invalid workflow	If there is a validation error in a workflow, check for the subflows.	Check whether the subflow contains any illegal flow or broken flow.

Error Message	Possible Cause	Possible Solution
Helper script name already exists	Helper Script name should be unique.	Provide some other name.
Name has invalid entry	Only few special characters are allowed.	Allowed characters: '-', '_', '(', ')'"


5. To validate the script, click **Validate**.

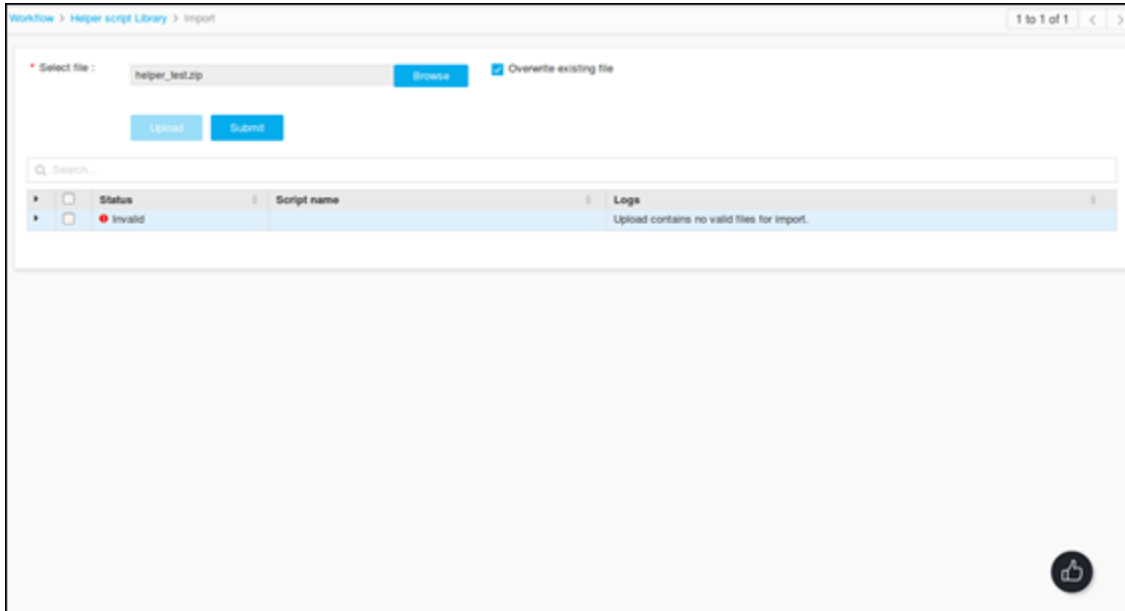
6. Click **Save**.

Error Message	Possible Cause	Possible Solution
Syntax error/ Indentation Error	Validated python script may contain syntax errors / indentation errors.	Fix the syntax/indentation errors in the python script.
Module not found	Validated python script may contain modules that are not available in python/AppViewX or imported module name is wrong.	Import the module with the right name or add the required modules to AppViewX.



Importing a helper script

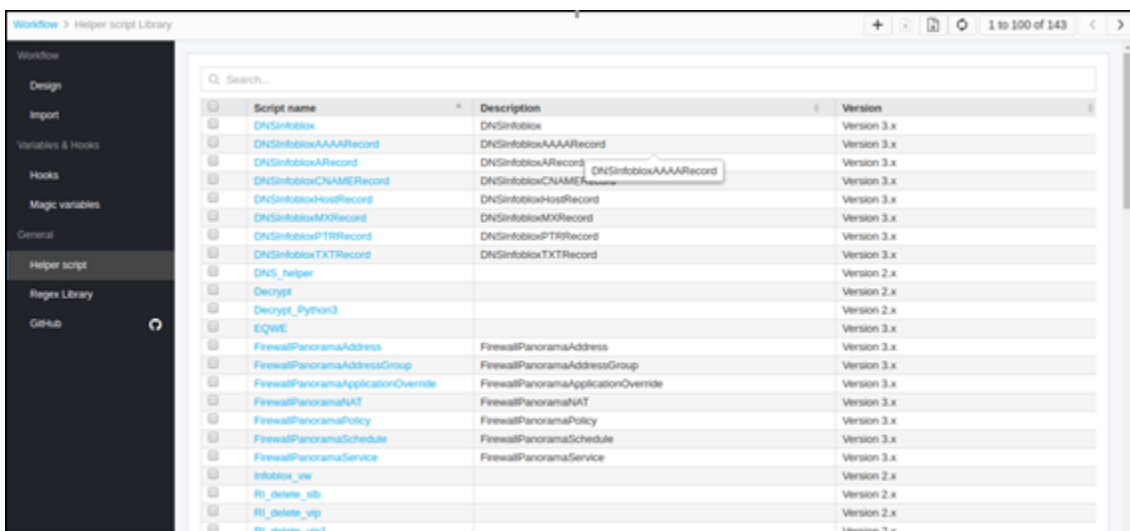
1. On the **Helper Script Library** page, click .
2. To select a file, click **Browse**.
3. Click **Upload**.



Error Message	Possible Cause	Possible Solution
Invalid helper script format	The file contains invalid entries and it cannot be parsed.	Choose an import file that has valid helper script details.

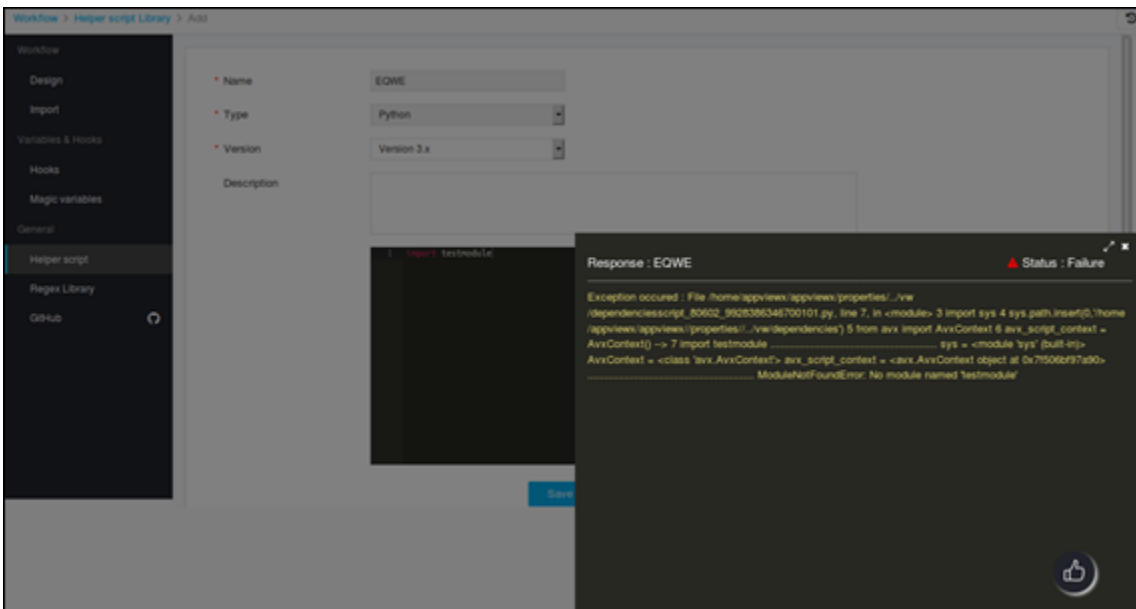
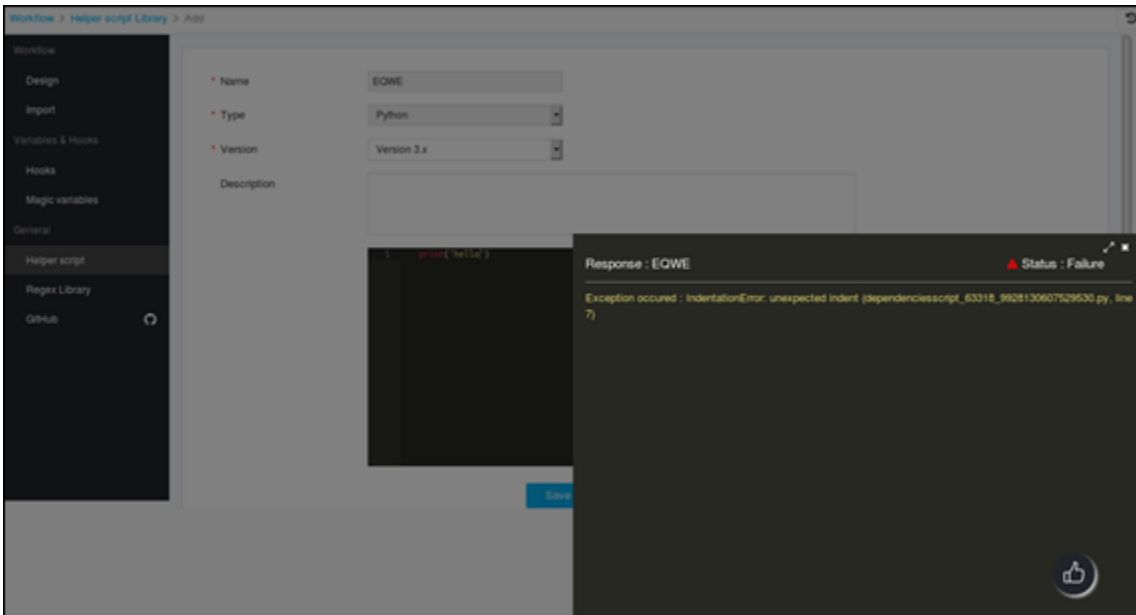
Updating a helper script

1. On the **Helper Script Library** page, click on an existing **Helper Script**.
2. On the **Modify** page, edit the Python script.



3. To validate the script, click **Validate**.

4. Click **Save**.



Error Message	Possible Cause	Possible Solution
Syntax error/	Validated python script may contain syntax errors / indentation errors.	Fix the syntax/indentation errors in the python script.

Error Message	Possible Cause	Possible Solution
Indentation Error		
Module not found	Validated python script may contain modules that are not available in python/AppViewX or imported module name is wrong.	Import the module with the right name or add the required modules to AppViewX.

Chapter 8: Hooks

- [Overview](#)
- [Query explorer Hook](#)
- [Script Hook](#)
- [REST Hook](#)
- [Exporting a Hook](#)
- [Importing a Hook](#)

Overview


- Hooks is a mechanism to query data from different sources and leverage them as part of the workflow automation process.
- The mode of query could be either through – Query Explorer, Script, REST API.
- The data sources could be either via the database, or the device.

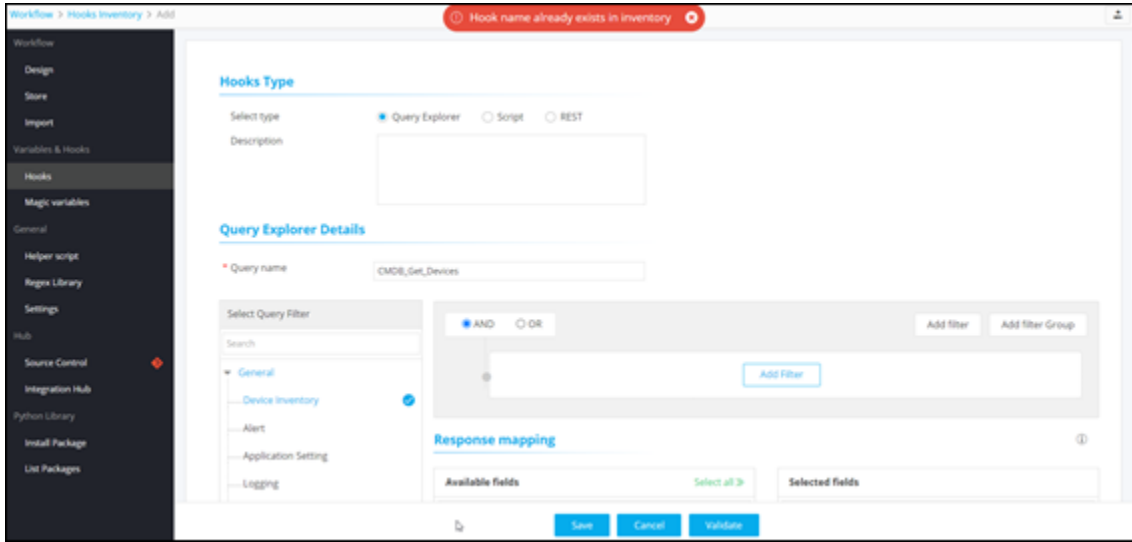
Query explorer Hook

Query explorer allows for users to design custom queries, business logic; and build self-service forms quickly.

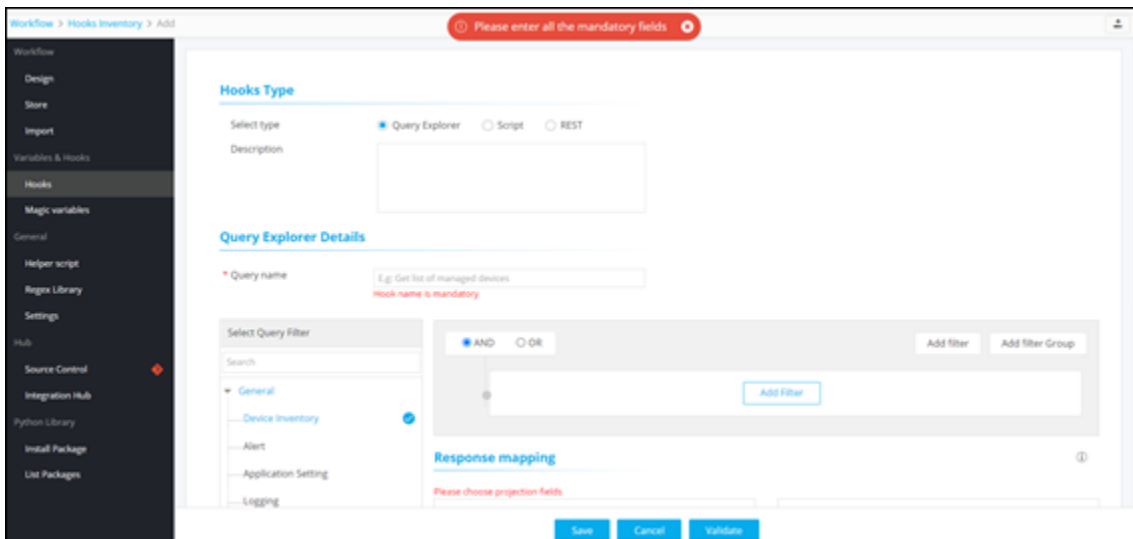
- [Creating a Query explorer Hook](#)

Creating a Query explorer Hook

1. On the Workflow Inventory page, from the navigation pane on the left, click Hooks.
2. To create a hook, click  in the command bar.
3. Select **Hooks Type** as **Query Explorer**.
4. Enter a valid description for the hook.
5. Under **Query Explorer Details**, enter or select the field information.
6. Click **Save**.



Error Message	Possible Cause	Possible Solution
Hook name already exists inventory	Hook name should be unique.	Provide some other name




Error Message	Possible Cause	Possible Solution
Please enter mandatory fields	Hook name/Projection fields maybe missing.	Provide hook name/ select projection fields.

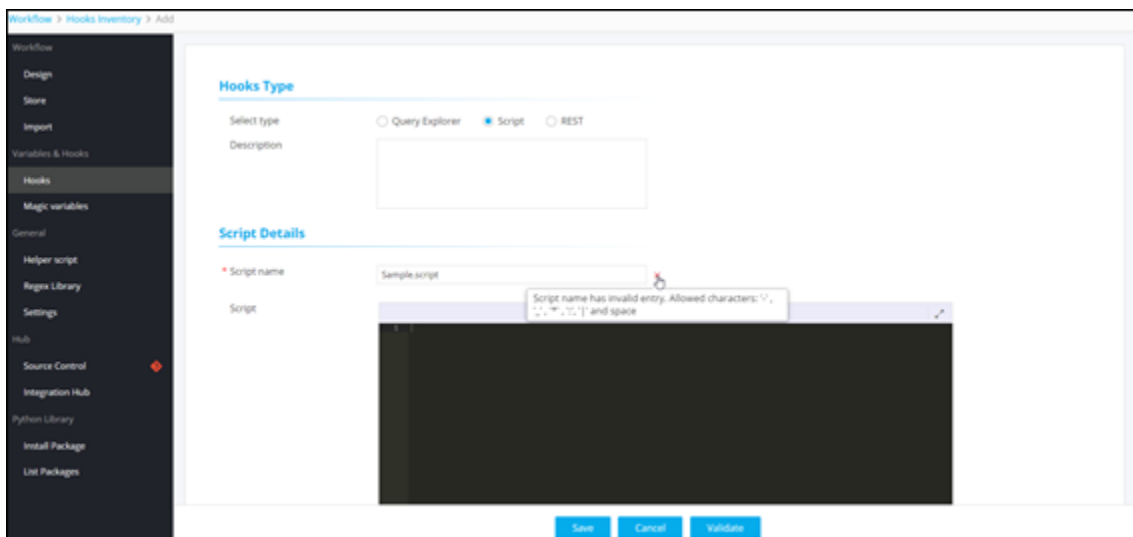
Script Hook

- Scripts are used to dynamically query the database, device, or with any external systems.
- The response from the query will in turn be mapped to the appropriate field(s) on the form.
- The need for an associate script arises owing to the dynamic nature of using a self-servicing form, with custom form fields; and queries.
- The form builder provides the user with a provision to define one or more 'associate scripts'; and map them against the relevant form field.
- [Creating a Script Hook](#)

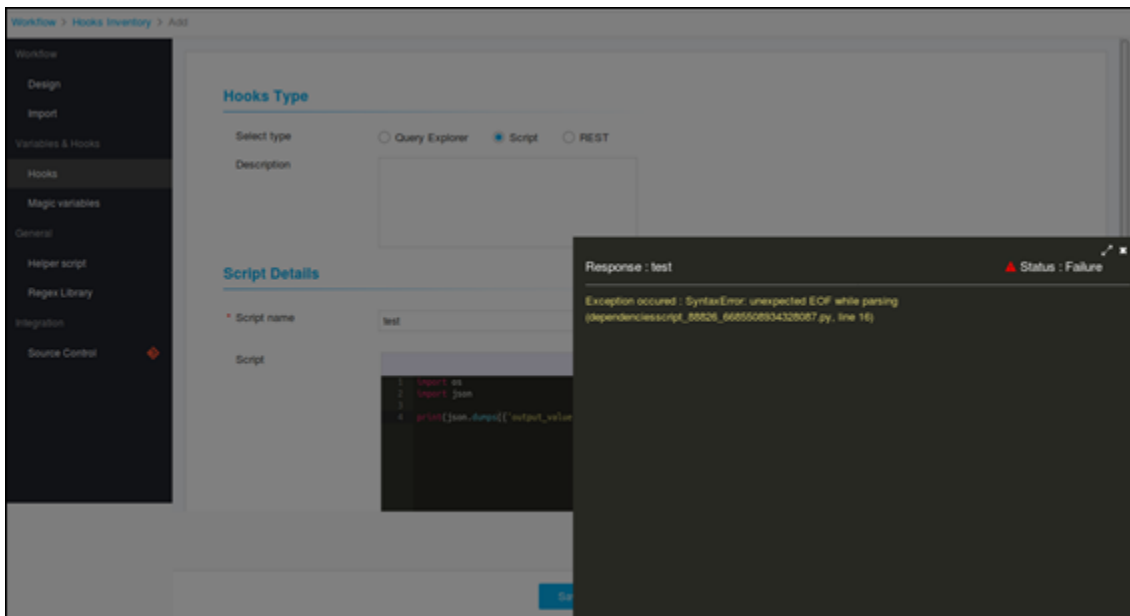
Creating a Script Hook

1. On the **Workflow Inventory** page, from the navigation pane on the left, click **Hooks**.
2. To create a hook, click  in the command bar.
3. Select **Hooks Type** as **Script**.
4. Enter a valid description for the hook.
5. Under **Script Details**, provide a **Script name**.
6. Define the script logic. To expose the output variables from the script, use the following format:


```
print(json.dumps({'field1':'value1','field2':'value2'}))
```
7. To validate the script, click **Validate**.
8. Click **Save**.



Error Message	Possible Cause	Possible Solution
Hook name already exists	Hook name should be unique.	Provide some other name.
Name has invalid entry	Only few special characters are allowed.	Allowed characters: '-', '_', '(', ')'"




Error Message	Possible Cause	Possible Solution
Syntax error/ Indentation Error	Validated python script may contain syntax errors / indentation errors.	Fix the syntax/indentation errors in the python script.
Module not found	Validated python script may contain modules that are not available in python/AppViewX or imported module name is wrong.	Import the module with the right name or add the required modules to AppViewX.

REST Hook

REST API allows querying the database, device or external systems.



- [Creating a REST Hook](#)

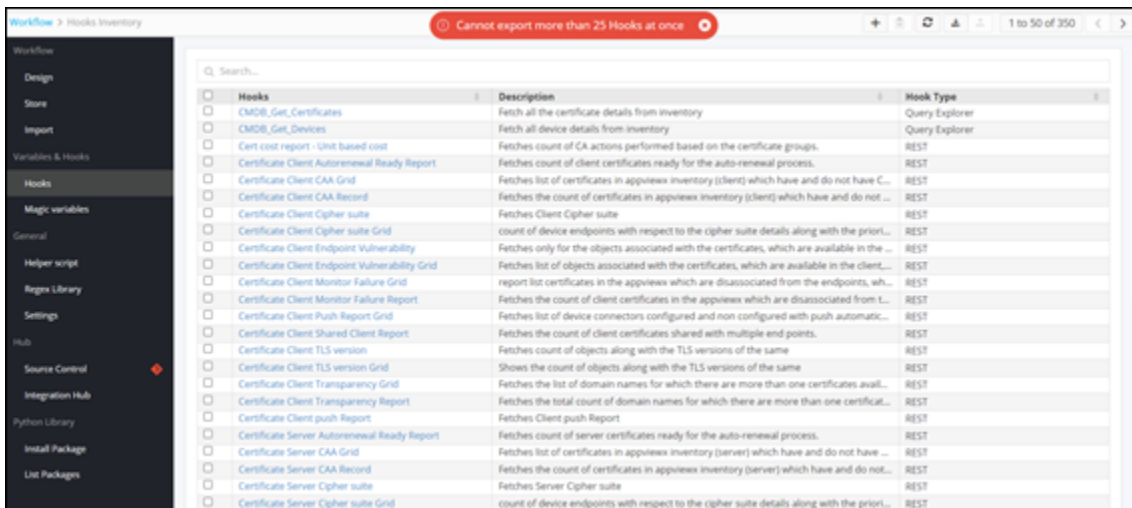
Creating a REST Hook

1. On the **Workflow Inventory** page, from the navigation pane on the left, click **Hooks**.
2. To create a hook, click  in the command bar.
3. Select **Hooks Type** as **REST**.
4. Enter a valid description for the hook.
5. Under **API Details**, provide an **API name**.
6. To enter API URL, under Select type, select the **REST** option.
7. To select from a list of available URLs, select the **Internal** option.
8. Define the script logic.
9. Provide values for **Header**, **Payload**, and use **Response mapping** to expose variables from the response.
10. To validate the hook, click **Validate**.
11. Click **Save**.

Error Message	Possible Cause	Possible Solution
Hook name already exists	Hook name should be unique.	Provide some other name.
Name has invalid entry	Only few special characters are allowed.	Allowed characters: '-', '_', '(', ')'"


Exporting a Hook

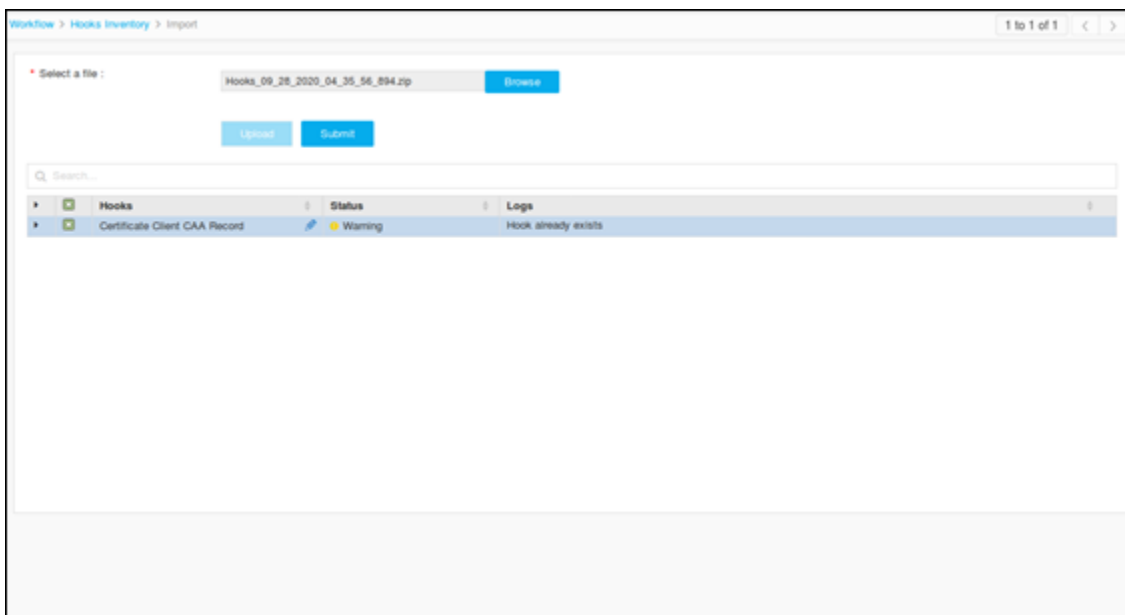
1. On the **Hooks Inventory** page, select the hooks individually or to select all hooks, from the top right corner of the screen, click .
2. To export selected hooks, click .



Error Message	Possible Cause	Possible Solution
Cannot export more than 25 hooks at once	Only 25 hooks can be exported at a time.	Select only 25 hooks to export.

Importing a Hook

1. On the **Hooks Inventory** page, from the top right corner of the screen, click .
2. To select a file with .zip format, click **Browse**.
3. Click **Upload**.



Error Message	Possible Cause	Possible Solution
Hook already exists	Hook imported which is already present.	Provide some other name for the hook.
Invalid hook format	The file contains invalid entries and it cannot be parsed.	Choose an import file that has valid hook details.


Chapter 9: Issues in Creating a Workflow Request

- [Overview](#)
- [Steps to create a request](#)
- [Submitting a user task](#)
- [Submitting an executed request](#)
- [Reflection of workflow data updations in a request](#)
- [Form task in a request](#)
- [Data manipulation in a request](#)
- [Command task in a request](#)

Overview

The Request console provides users with a single point view of all the workflows assigned, workflow requests triggered, and workflow(s) that are scheduled.

Steps to create a request

1. From the top left corner of the screen, click .
2. From the menu displayed, click **Request**.
3. On the Request page, from the navigation pane on the left, click **View/Run**.
4. To trigger a workflow, click .

Submitting a user task

Tasks that display the data in a specific format and require user's approval for further action are called user tasks. Following are the possible limitations that users might encounter during task execution:

- When an **unauthorised user** tries to submit a user task, unauthorized exception will be thrown
- When a user task is the first task of a workflow and when an unauthorised user tries to trigger that workflow, unauthorized exception will be thrown
- Triggering a request with following tasks as first task of the workflow:

- If
- Switch and
- Split task

These tasks cannot be seen in the Request stage view. However, you can view them in the Activity log on the Request page under All requests.

- Clean up flow cannot contain work order tasks in it

Submitting an executed request

- A rolled back request cannot be rolled back again or cloned
- A request which is in “in progress” status cannot be cloned or rolled back. Only requests with status (completed ,failed ,partial) can be cloned / rolled back.

Reflection of workflow data updations in a request

- Any changes made in the workflow after triggering a request will not be reflected in that particular request
- Cloning a request/cloned request will only comprise the request’s data from which it has been cloned and will not reflect any changes made to that workflow later

Form task in a request

- **Auto fetch** of associate script output will not occur if the Form task is saved as “draft” and opened again. It will load the data only in case of a newly triggered Form
- When **RGF flow** is enabled in form task, the request execution will take place in prescribed order by respective users who are allowed to create/review or submit the task in order
- **Pseudo form** is supported in following tasks alone
 - Slack
 - Rest
 - Rest (I)
 - Schedule
 - Script and Email tasks

Data manipulation in a request

- Any workflow containing **global key(s)** with dot (.) will not be allowed to create a request [Mongo behaviour]
- When the MongoDB document containing the request data exceeds **16mb in size** due to any uploaded file (or) any huge global variable data injection (or) heavy logs and so on, the request will hang

Command task in a request

- Command Task **execution time** is based on device's response time in which the commands are being executed.
- If multiple commands are configured in Command Task, there is no possibility of selecting a particular command for execution, it will execute all the commands present.

Command Task **execution order** cannot be scheduled within the task for each command.