Script documentation: create\_rack\_structures

[Introduction 2](#_Toc103879039)

[Prerequisites 2](#_Toc103879040)

[Running the script 3](#_Toc103879041)

[Scheduled execution 3](#_Toc103879042)

[On demand execution 4](#_Toc103879043)

# Introduction

This document describes the functionality of the create\_rack\_structures.py script. Its purpose is to identify and organize specific resources from the Velocity inventory and place them under nested rack structures, which will help the users browse the inventory more easily and identify more clearly resources grouped under certain racks.

# Prerequisites

1. Resources intended for rack organizing should NOT be placed under the following folders:
   1. Handsets
   2. Lab IP Addresses
   3. Laptops

A screenshot of a computer

Description automatically generated

All of the resources placed under the other folders (or not placed under a specific folder) will be taken into consideration by the script.

To check if a resource is placed into a specific folder:

1. Go to the Inventory -> Resources page;
2. Search for that specific resource into the search box;

Graphical user interface, text, application

Description automatically generated

1. Open it;
2. Check the right side of the screen for a property named ‘Folder’

Graphical user interface, text, application, email

Description automatically generated

1. If the resource is not placed under the proper folder, then use the ‘Move’ button shown in the screenshot above and select the proper folder to move the resource to.
2. Rack related properties should be valid

* In order for the script to be able to identify the resources which need to be organized and to know in which rack it should be placed under, it will scan the entire inventory and filter out the devices which do not have a valid set of rack positioning properties.
* The properties which are being used to organize the resources are:
  + Lab Row
  + Rack Number

Graphical user interface, text, application, email

Description automatically generated

* The values for these two properties need to be non-empty and different from the following values:
  + “None”
  + “To be filled(A-I)”
  + “To be filled(1-16)”
  + “N/A”

# Running the script

### Scheduled execution

The script is pre-configured to run nightly, and update the results automatically. No manual intervention or action is required.

### On demand execution

For the script to run successfully, it will need to have an available agent with Python execution capabilities. Verify this by going to Reports -> Velocity Agents and checking that there is an Online agent (green status) with the proper capability:

A screenshot of a computer

Description automatically generated

The following steps need to be followed in order to manually trigger a script execution:

1. Go to the Library -> Automation Assets page;
2. Search for create\_rack\_structures in the Search box;
3. Click on the Run Automation Asset button;

Graphical user interface, text, application, email

Description automatically generated

1. Click on Run if the script should run at this moment, or modify the values in the Schedule section to schedule an execution at a specific hour or date;

Graphical user interface, text, application, email

Description automatically generated

1. Go to the Reports -> Executions page to check the status of the script execution;
2. The execution will take approximately up to 3 or 4 minutes to finish.

After the execution is finished successfully, the Racks folder can be checked in the Resources page. Each rack and their rows can be expanded to see their respective structure, together with the devices which were nested under them.

A screenshot of a computer

Description automatically generated

Note:

* if one or more devices are added to the inventory and are matching the criteria listed into the Prerequisites section, then the user should run this script in order to have them nested under their respective racks;
* if a device was previously nested under a certain rack but its properties were invalidated/deleted, then script will remove that certain resource from under the rack;
* if a new device will be created which will contain new Lab Row or Rack Number property values ((for example Row J, RACK-01), then a new rack will be created in the inventory with the proper name, and the resource will be nested under it;