

INTERNAL

Project Piper and Docker Setup

Table of ContentS

[PRE-REQUISITES 3](#_Toc83024357)

[dOCKER Setup 3](#_Toc83024358)

[Docker Download 3](#_Toc83024359)

[Verify if Docker is running 5](#_Toc83024360)

[WINDOWS SUBSYSTEM FOR linux iNSTALLATION 8](#_Toc83024361)

[Check Requirements For Running WSL 2 8](#_Toc83024362)

[Enable the Windows Subsystem for Linux 8](#_Toc83024363)

[Enable Virtual Machine Feature 9](#_Toc83024364)

[Verification 9](#_Toc83024365)

[Download and Install the Linux Kernel Update Package 10](#_Toc83024366)

[Set WSL 2 as your default version 11](#_Toc83024367)

[Install and Run Docker Images for Project Piper 12](#_Toc83024368)

[Install Jenkins-Master using Docker Image 12](#_Toc83024369)

[Run Jenkins-Master Docker Image 14](#_Toc83024370)

[Login to Jenkins 15](#_Toc83024371)

[Change the password of the user. 16](#_Toc83024372)

[Configure Github Credentials 17](#_Toc83024373)

[Enter credentials for SAP BTP account. 20](#_Toc83024374)

[Create a sample Fiori Application 21](#_Toc83024375)

[Setup SAP BTP Trial Account 21](#_Toc83024376)

[Setup SAP Business Application Studio 21](#_Toc83024377)

[Create a Dev space in your SAP Business Application Studio 22](#_Toc83024378)

[Create a sample Fiori App 22](#_Toc83024379)

[Create a Job 27](#_Toc83024380)

[Headline 3 31](#_Toc83024381)

[Headline 4 32](#_Toc83024382)

[Headline 2 32](#_Toc83024383)

This document contains setup and validation steps for Project Piper. Project Piper is a shared library that contains reusable steps that enables customers to re-use and customize pre-configured pipelines and even build their own pipelines. These pipelines can be used in the CI/CD processes to help customers develop software with short feedback cycles.

# PRE-REQUISITES

We will be setting up Docker to run Jenkins on our laptops. Docker requires certain pre-requisites to be met before it can be used to run Jenkins. Please make sure the following pre-requisites are met.

* Your computer CPU must support virtualization.
* Dedicate 4GB of memory for Docker.
  + Recommendation: Use a 16GB memory computer.

# dOCKER Setup

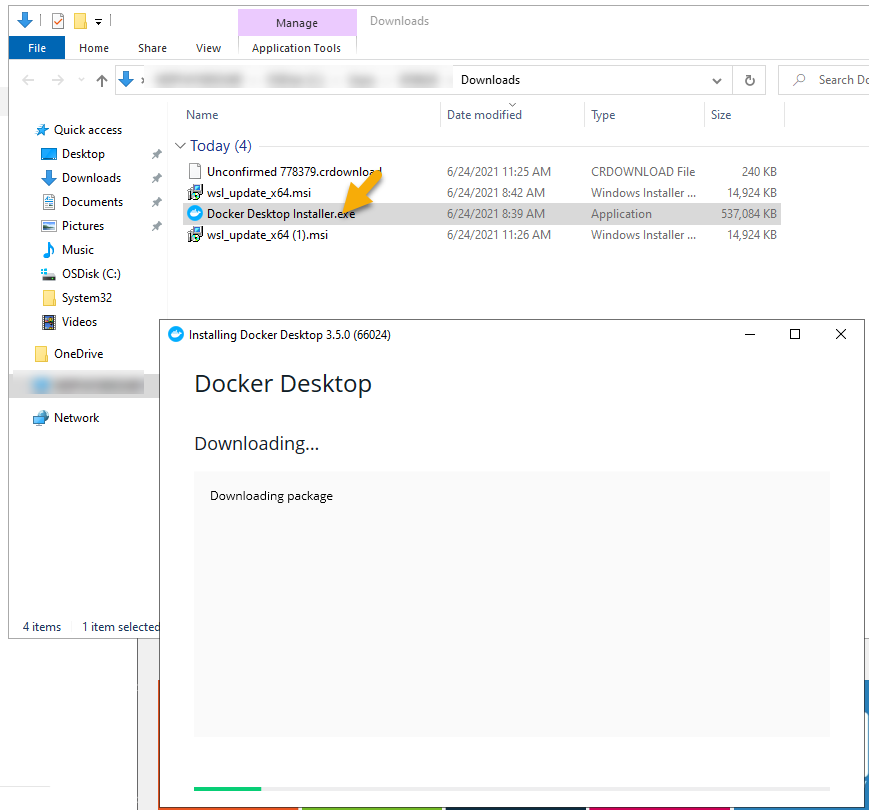
We will be using a containerized version of Jenkins pre-configured with necessary tools and environment to utilize Project Piper libraries. This can be easily achieved by running a Docker Image prepared by the Piper team. Steps below shows how to setup Docker.

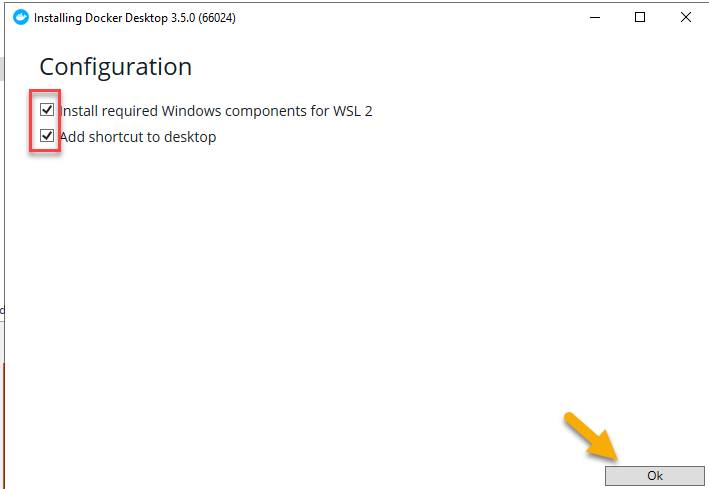
## Docker Download

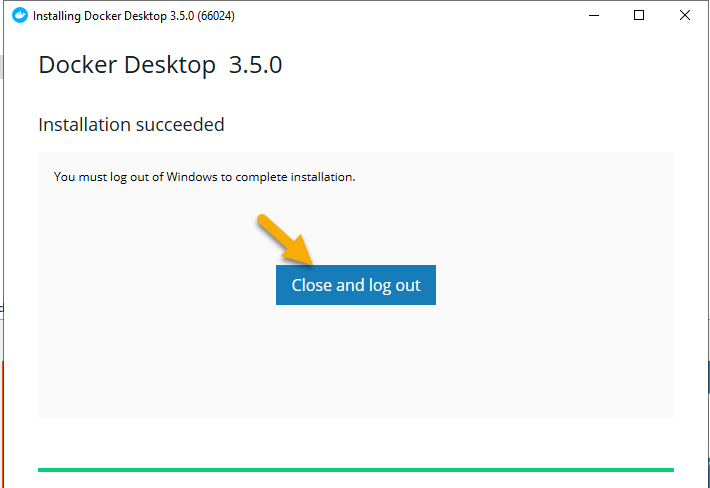
Download Docker Desktop from the link below.

<https://hub.docker.com/editions/community/docker-ce-desktop-windows/>

Once it is downloaded, double click on it to run the installer.





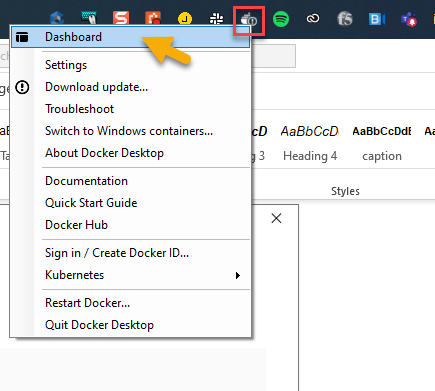


Click on ‘Close and log out’.

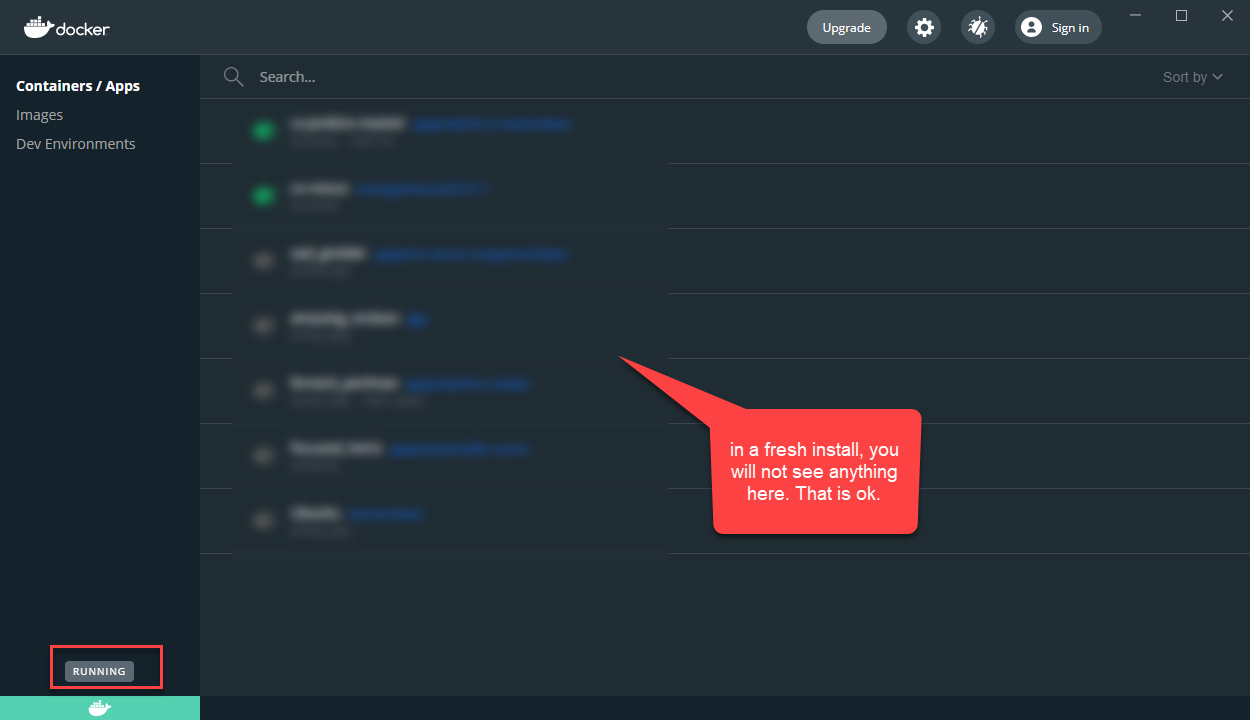
## Verify if Docker is running

Look for a whale icon in the tool bar as shown below and right click and select ‘Dashboard’

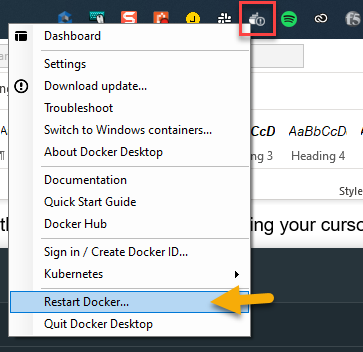




It will open the Docker Dashboard. Check if the docker is running by hovering your cursor at the location shown in the picture below.



If Docker isn’t running, the right click and choose ‘restart docker’ as shown below.



This might take few minutes to docker to start.

If it still doesn’t run, execute the steps in the next section “**WINDOWS SUBSYSTEM FOR LINUX INSTALLATION**“ completely and restart Docker again.

# WINDOWS SUBSYSTEM FOR linux iNSTALLATION

## Check Requirements For Running WSL 2

To update to WSL 2, you must be running Windows 10.

For x64 systems: **Version 1903** or higher, with **Build 18362** or higher.

For ARM64 systems: **Version 2004** or higher, with **Build 19041** or higher.

Builds lower than 18362 do not support WSL 2. Use the [Windows Update Assistant](https://www.microsoft.com/software-download/windows10) to update your version of Windows.

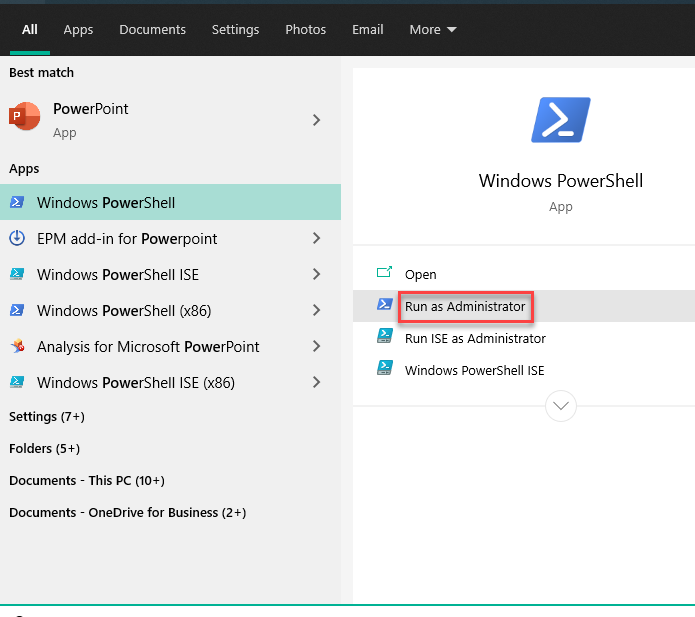
To check your version and build number, select **Windows logo key + R**, type **winver**, select **OK**. [Update to the latest Windows version](ms-settings:windowsupdate) in the Settings menu.

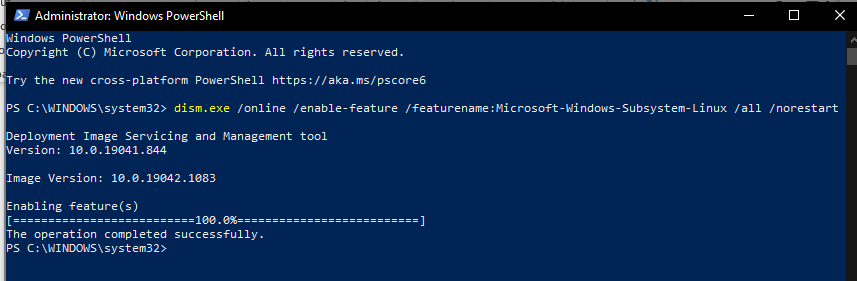
## Enable the Windows Subsystem for Linux

You must first enable the "Windows Subsystem for Linux" optional feature before installing any Linux distributions on Windows.

Open PowerShell as Administrator and run:

*dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart*





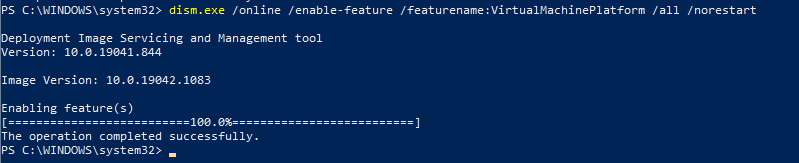
## Enable Virtual Machine Feature

Before installing WSL 2, you must enable the **Virtual Machine Platform** optional feature. Your machine will require [virtualization capabilities](https://docs.microsoft.com/en-us/windows/wsl/troubleshooting#error-0x80370102-the-virtual-machine-could-not-be-started-because-a-required-feature-is-not-installed) to use this feature.

Open PowerShell as Administrator and run:

PowerShell

*dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart*

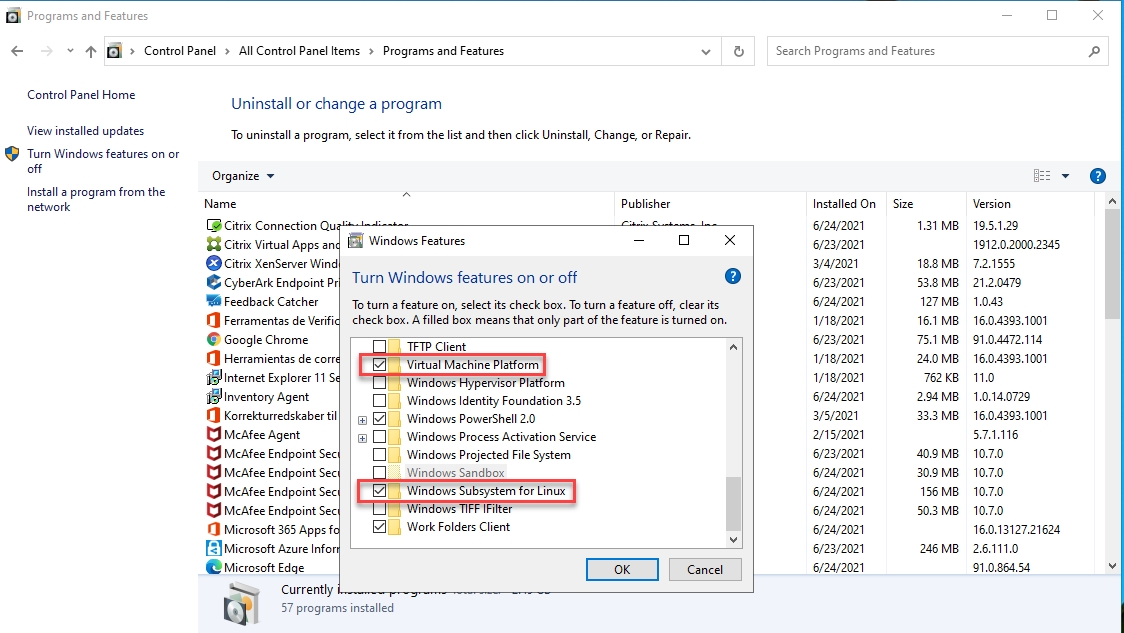


**Restart** your machine to complete the WSL install and update to WSL 2.

## Verification

The WSL and Virtual Machine Platform features enabled in the previous steps can be verified by going to

Control Panel -> Programs and Features -> Turn Windows Features On or Off. This will open a pop-up as shown below and make sure the highlighted features are checked.



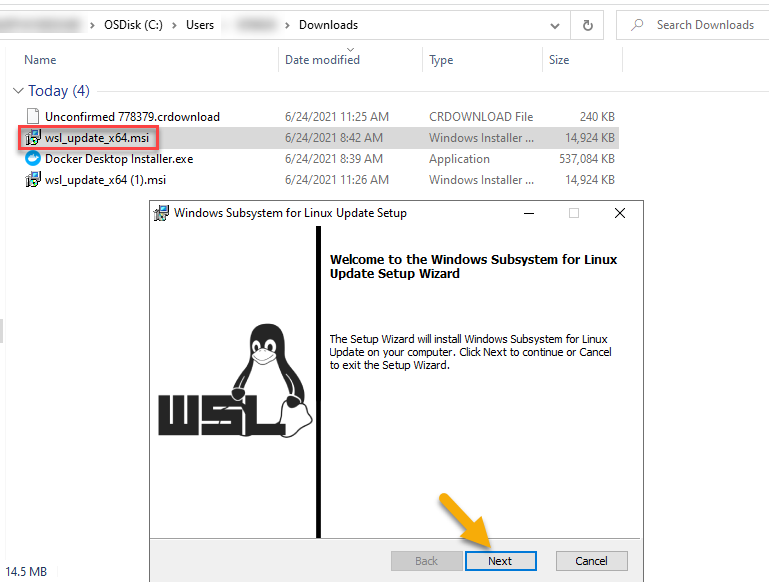
## Download and Install the Linux Kernel Update Package

Download the latest package: [WSL2 Linux kernel update package for x64 machines](https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi)

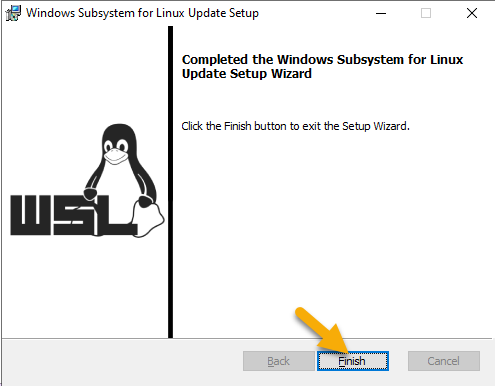
**Note**

If you're using an ARM64 machine, please download the [ARM64 package](https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_arm64.msi) instead. If you're not sure what kind of machine you have, open Command Prompt or PowerShell and enter: systeminfo | find "System Type". **Caveat:** On non-English Windows versions, you might have to modify the search text, for example, in German it would be systeminfo | find "Systemtyp".

Run the update package downloaded in the previous step. (Double-click to run - you will be prompted for elevated permissions, select ‘yes’ to approve this installation.)



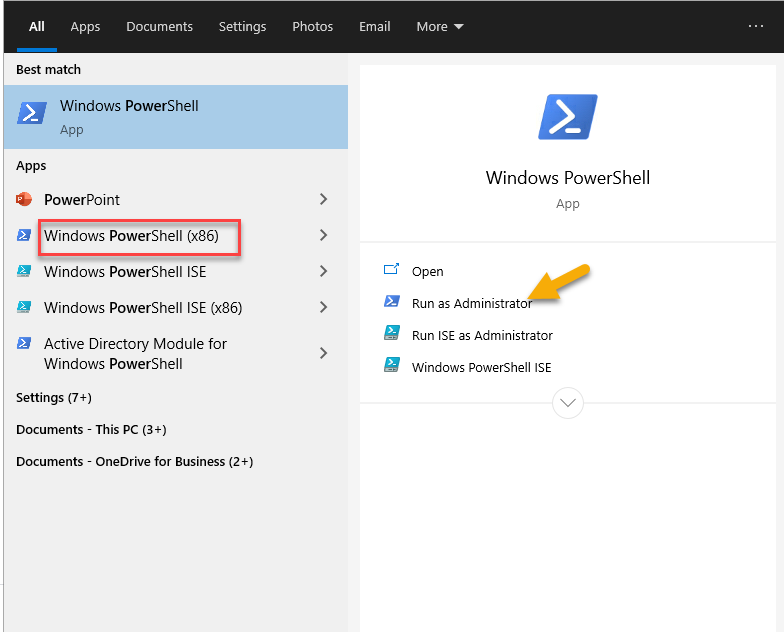
Finish the installation by clicking on Finish button.

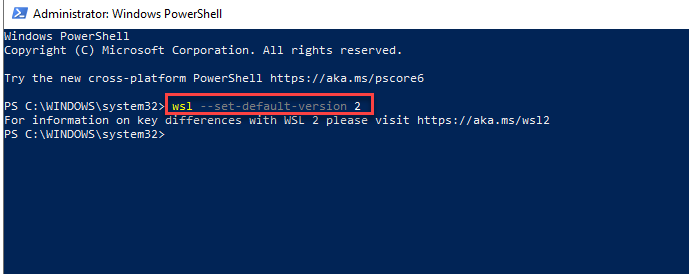


## Set WSL 2 as your default version

Open PowerShell and run this command to set WSL 2 as the default version when installing a new Linux distribution:

*wsl --set-default-version 2*





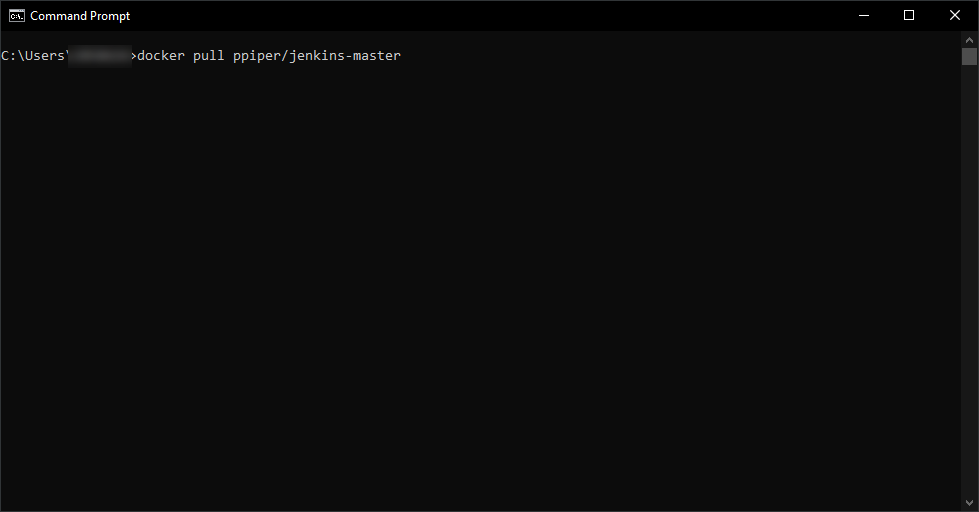
## Install and Run Docker Images for Project Piper

Please make sure Docker is running before you proceed with next steps.

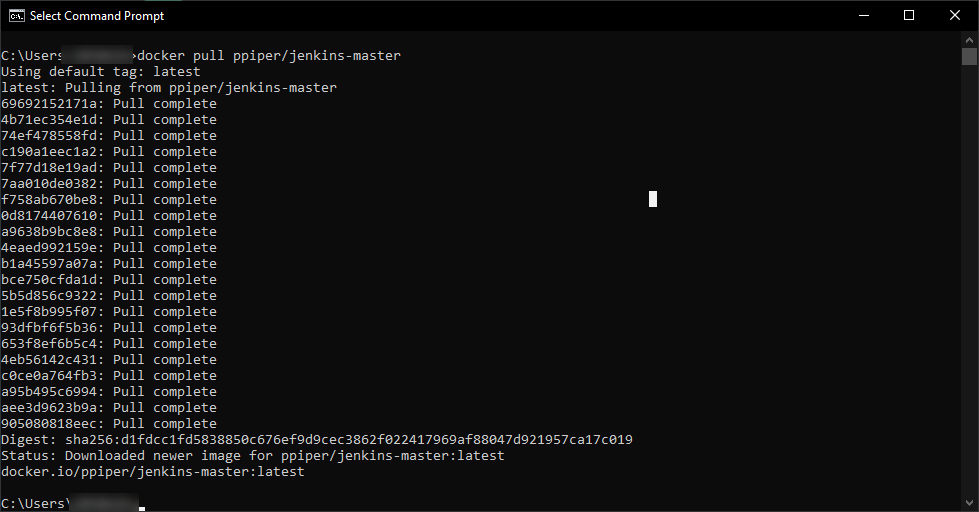
### Install Jenkins-Master using Docker Image

In a command prompt enter the command below and hit enter. This will download the image locally.

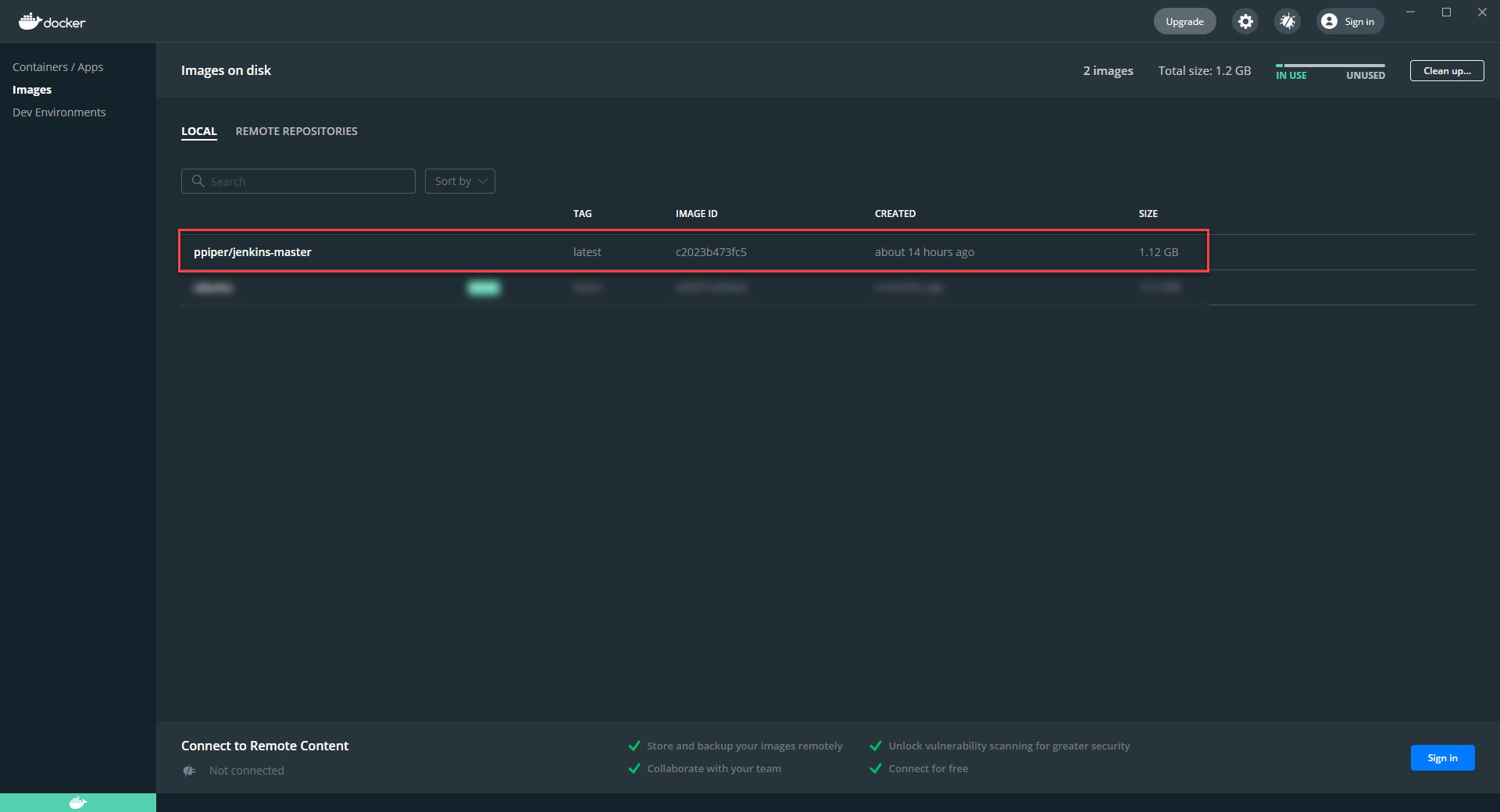
*docker pull ppiper/jenkins-master*



You should see an ouput as shown in the picture below.



Go to the Docker Dashboard and check if you can see this image.



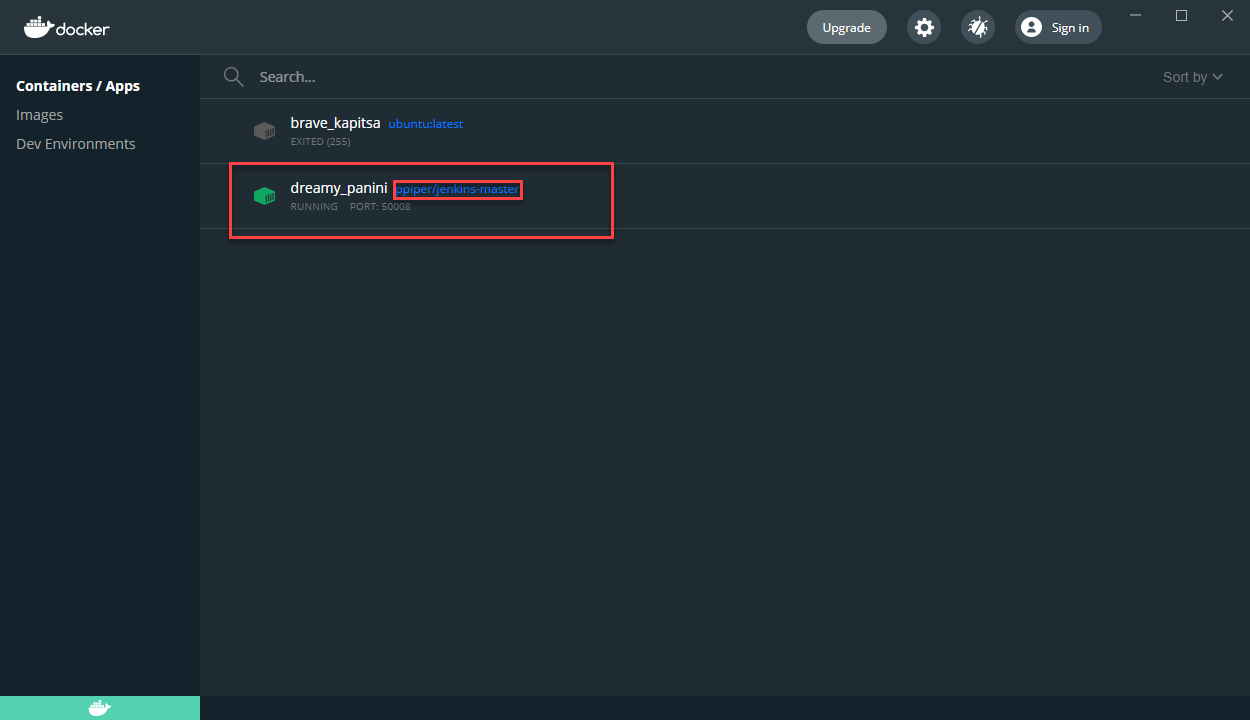
#### Run Jenkins-Master Docker Image

Open a command prompt and run the command below to run the docker image at a specific port.

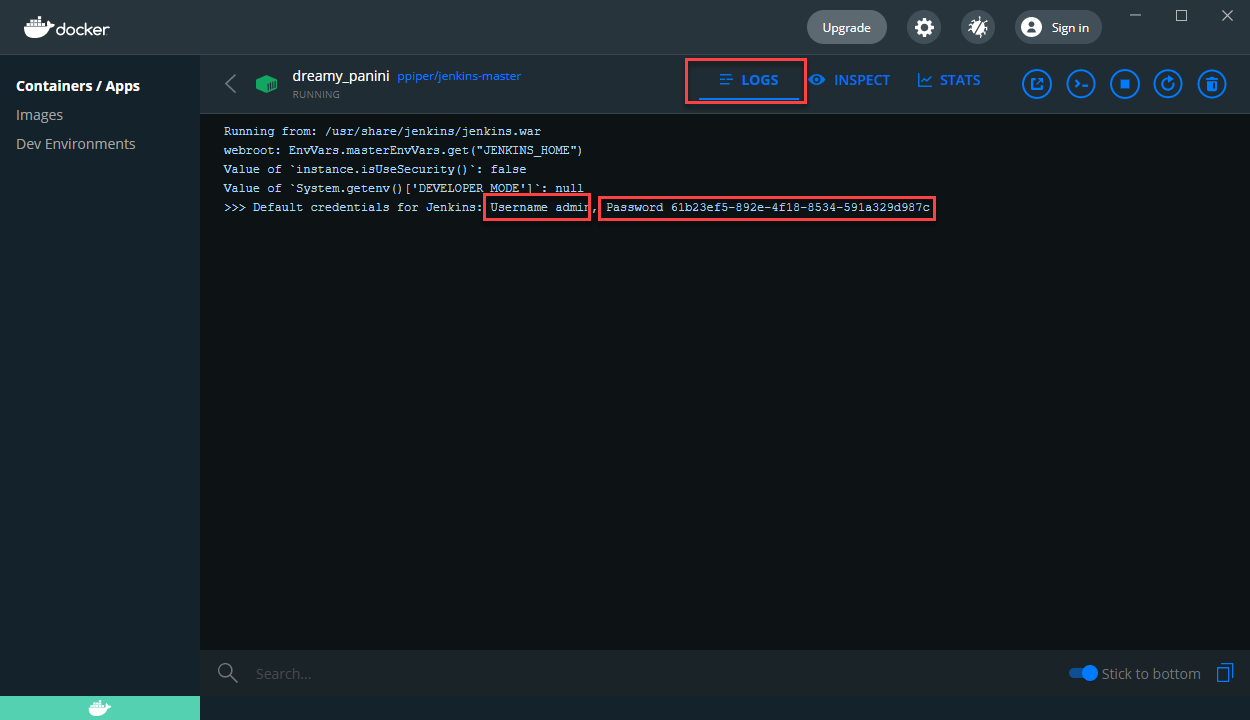
*docker run -d -p 8088:8080 -p 50008:50000 --user root -v /var/run/docker.sock:/var/run/docker.sock ppiper/jenkins-master*



Go to Docker Dashboard and switch to the containers tab. You should see a container running as shown below.

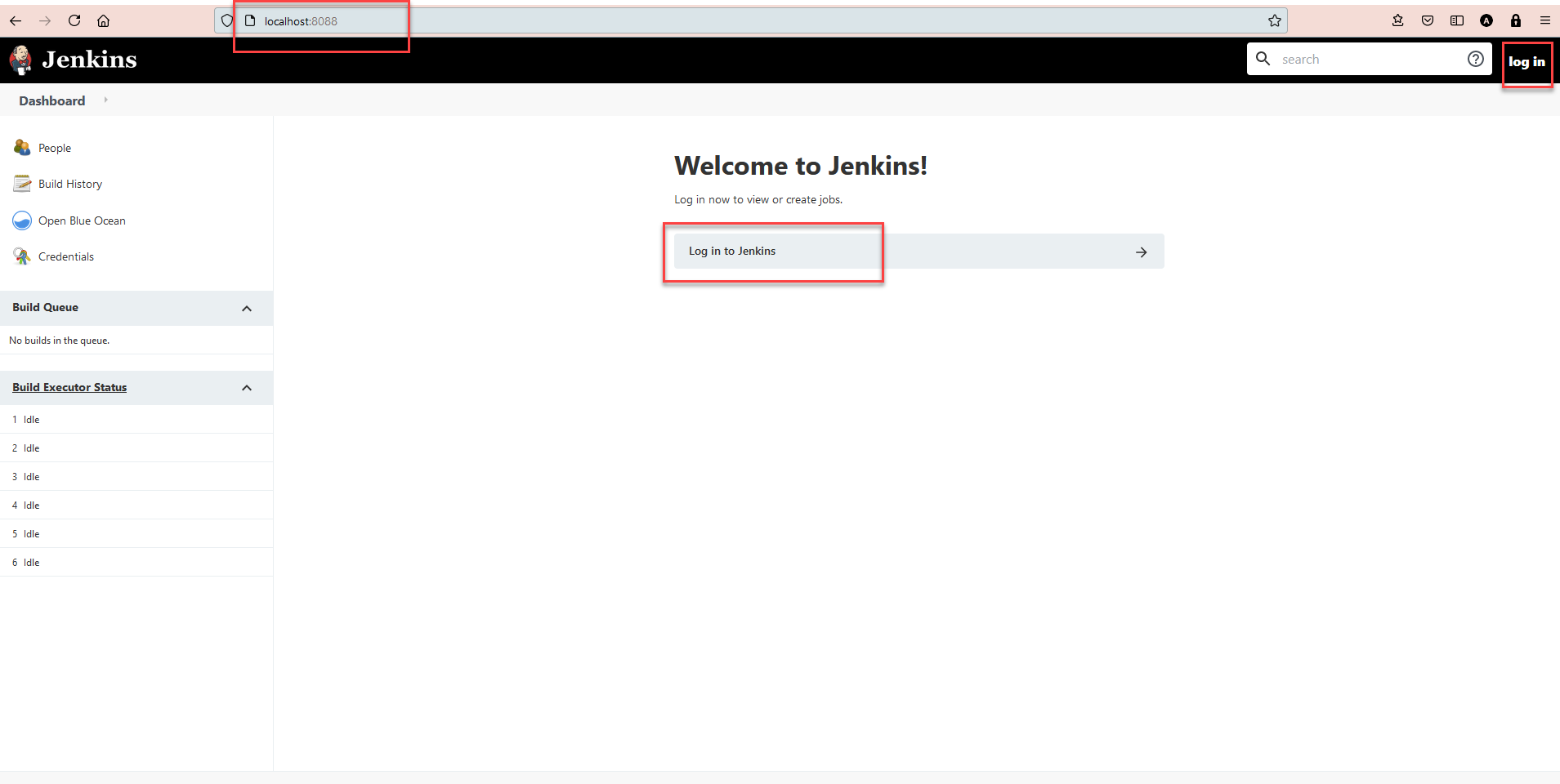


Click on the container and it takes you to a terminal showing logs. These logs contain the initial username and password to login to Jenkins.

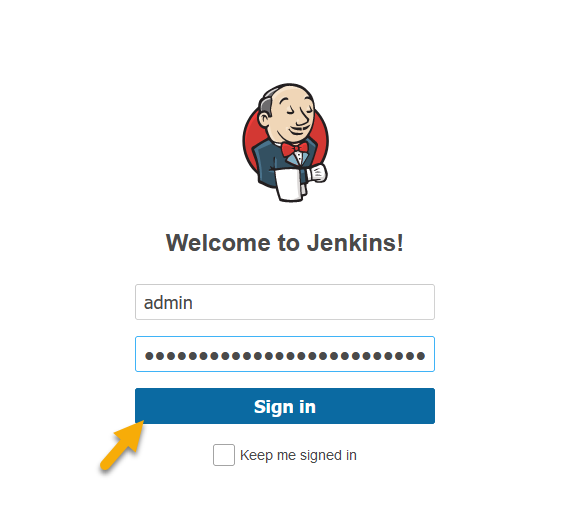


#### Login to Jenkins

Open the url <http://localhost:8088/> on your browser. This should take you to the login page.

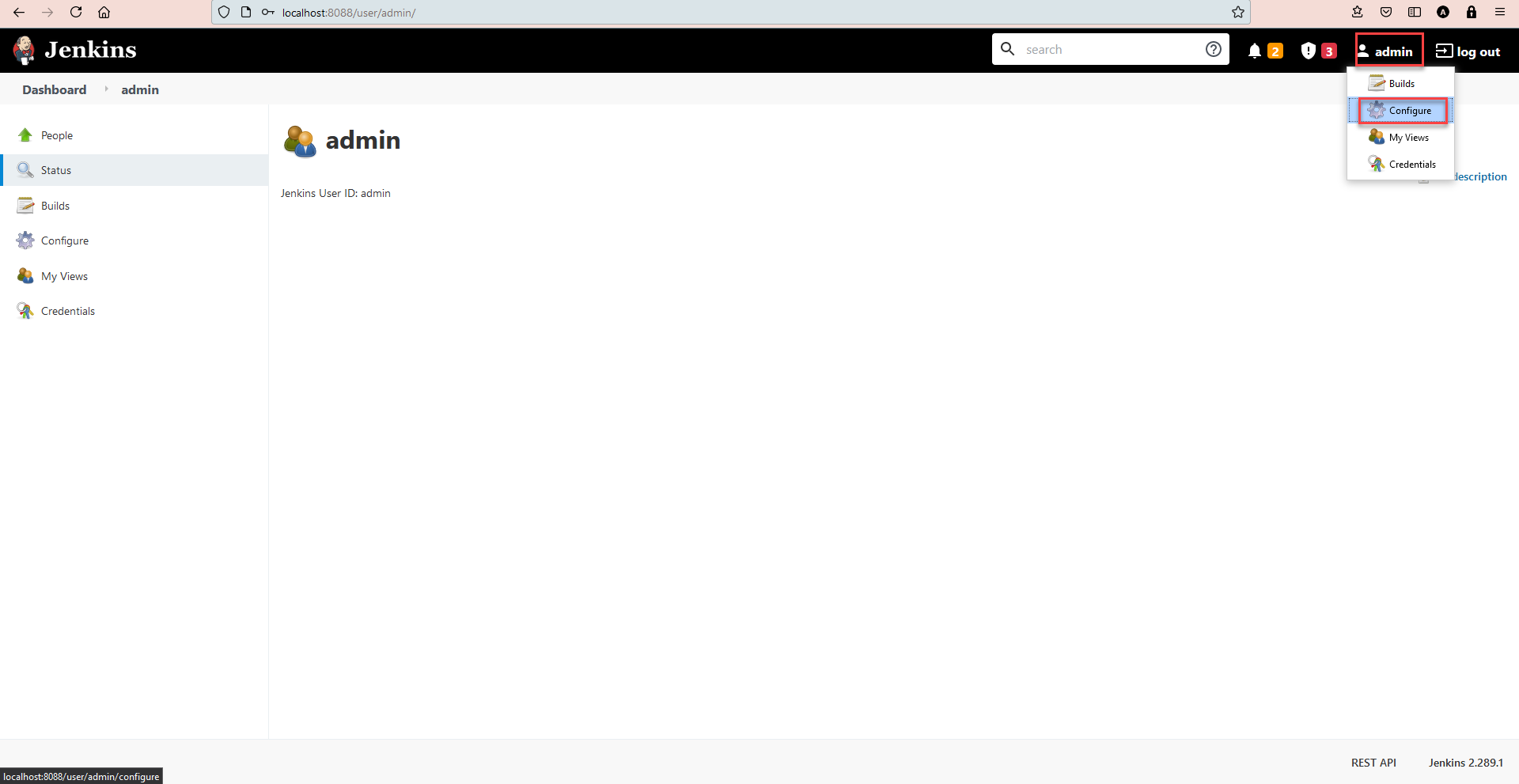


Use the credentials from the terminal and login.

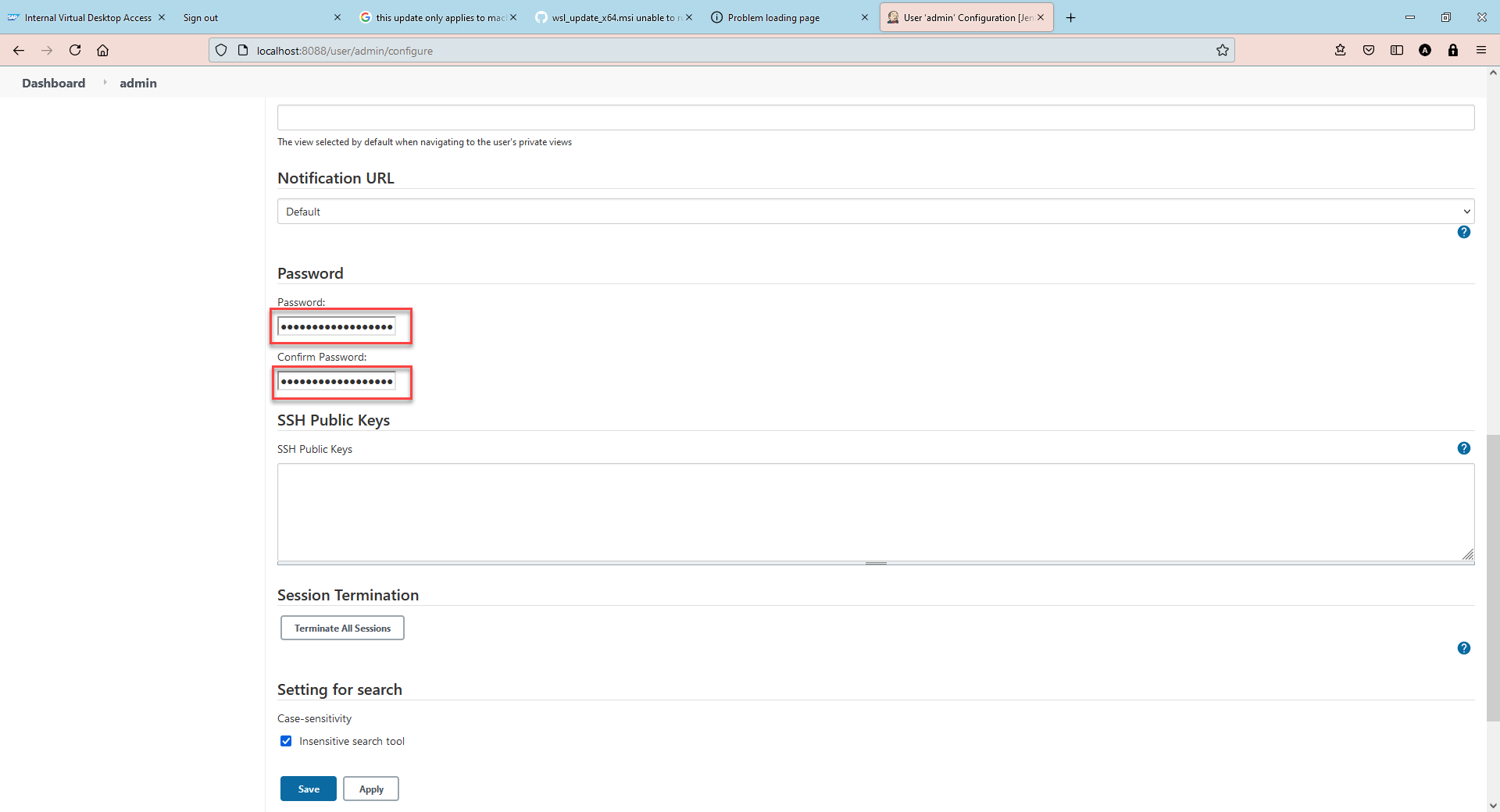


#### Change the password of the user.

Navigate as shown in the picture.

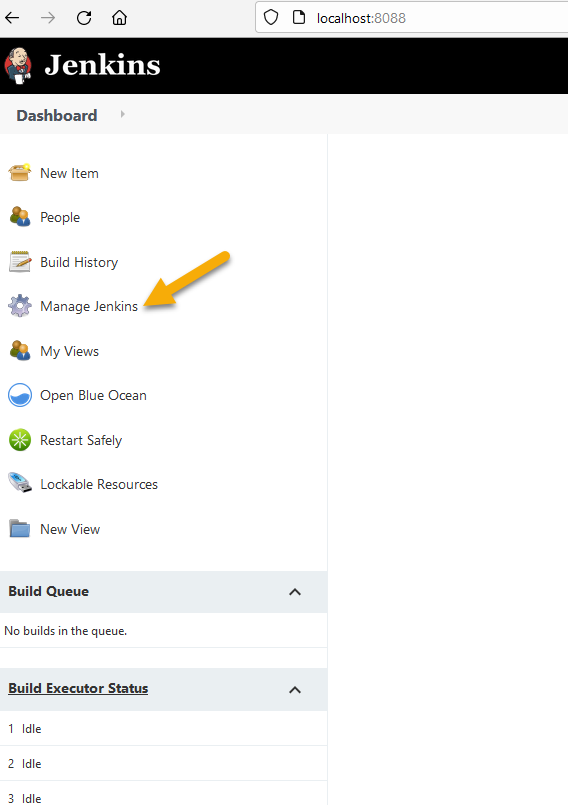


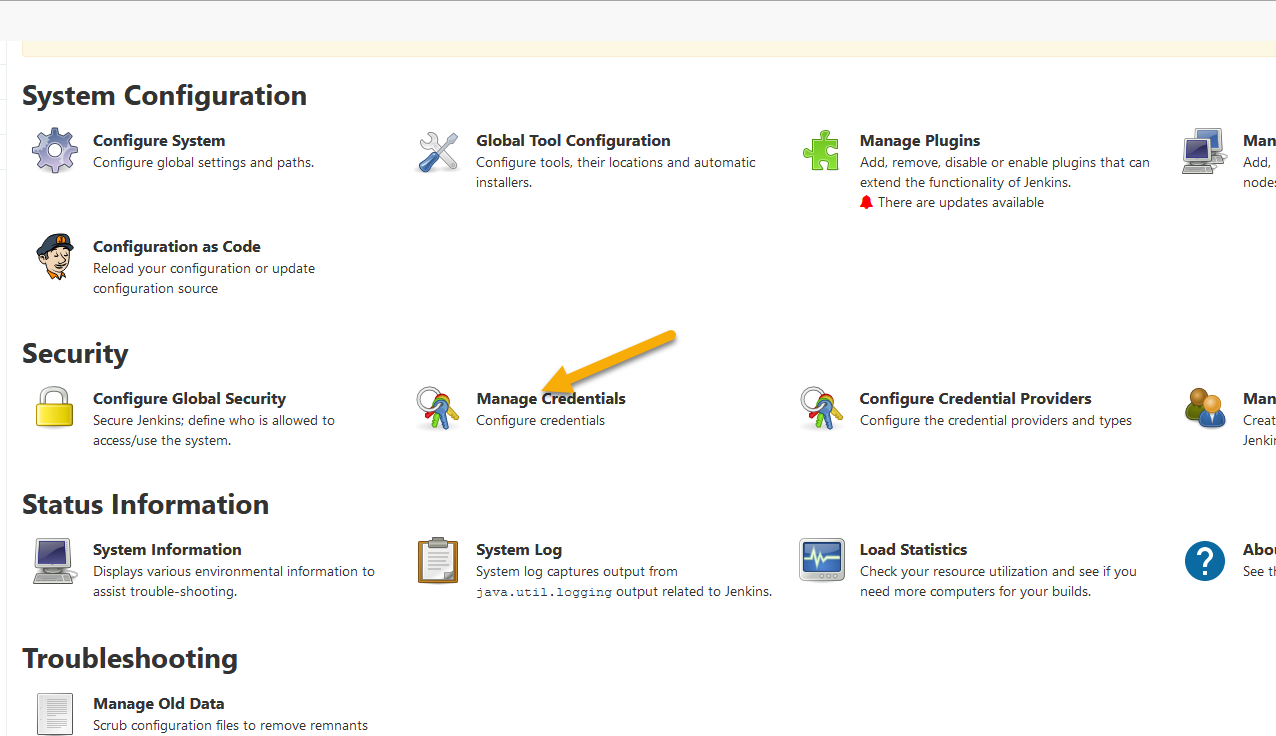
Update the password in the field below and save it.

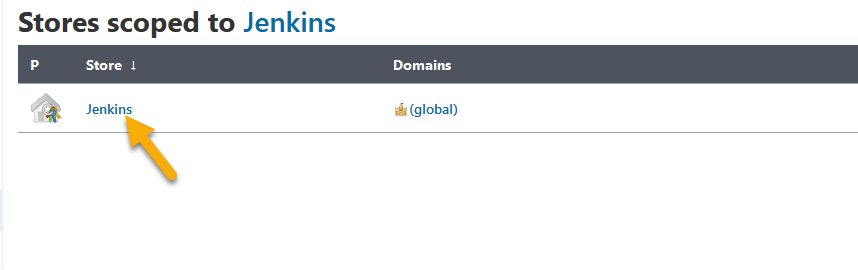


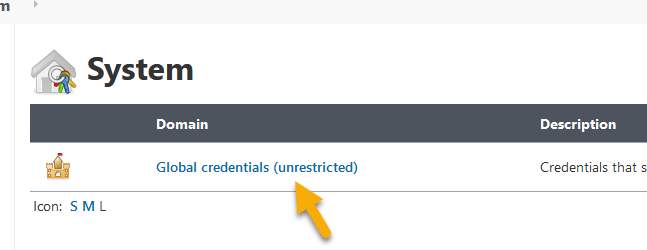
### Configure Github Credentials

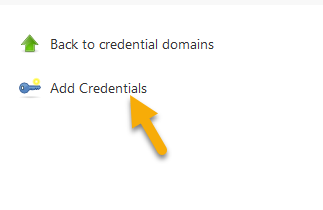
Navigate to the credentials area as shown below.











In the next screen enter username, password and an unique ID for your Github credentials.

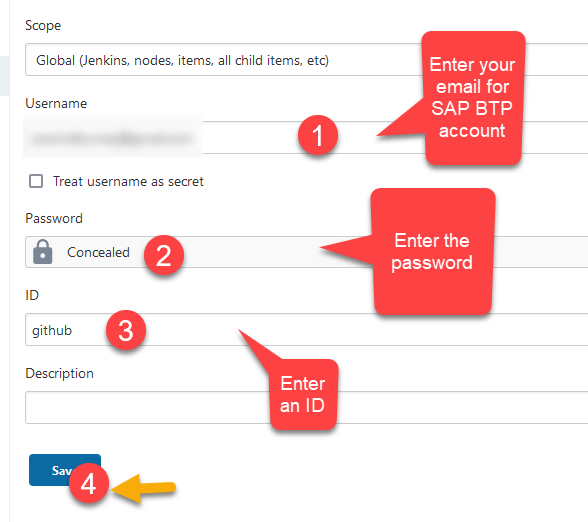


Save it.

Note: You can create a public access token by following the steps in the link below. (Follow steps 1-10)

<https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/creating-a-personal-access-token>

#### Enter credentials for SAP BTP account.



Save it.

Note: This ID will be used in the config.yml file later.

## Create a sample Fiori Application

### Setup SAP BTP Trial Account

Follow the developer tutorial below to setup a SAP BTP Trial Account

<https://developers.sap.com/tutorials/hcp-create-trial-account.html>

### Setup SAP Business Application Studio

Follow the developer tutorial below to setup SAP Business Application Studio.

<https://developers.sap.com/tutorials/appstudio-onboarding.html#af6e8226-4bec-4440-8db1-dda2da67a441>

Note: Step 3 in the tutorial above is optional.

### Create a Dev space in your SAP Business Application Studio

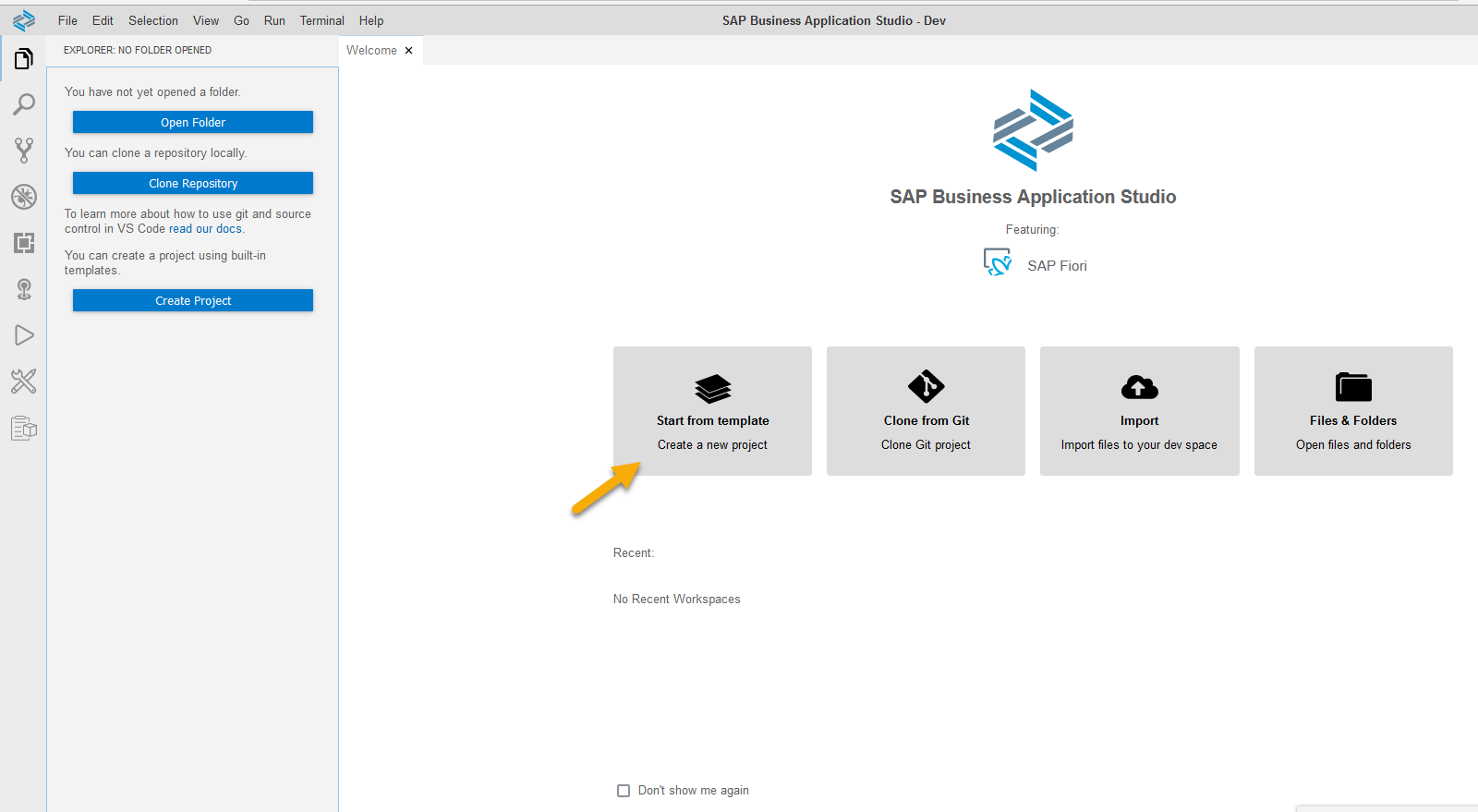
Follow the developer tutorial below to create a Dev Space.

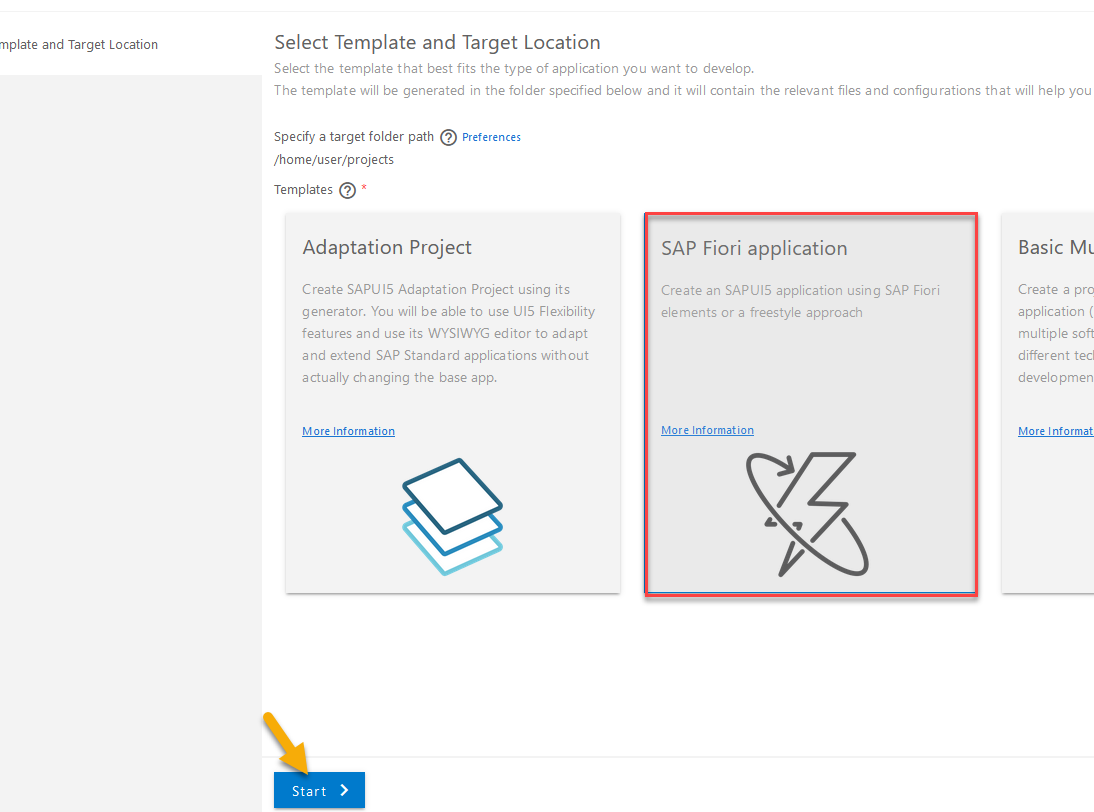
<https://developers.sap.com/tutorials/appstudio-devspace-create.html>

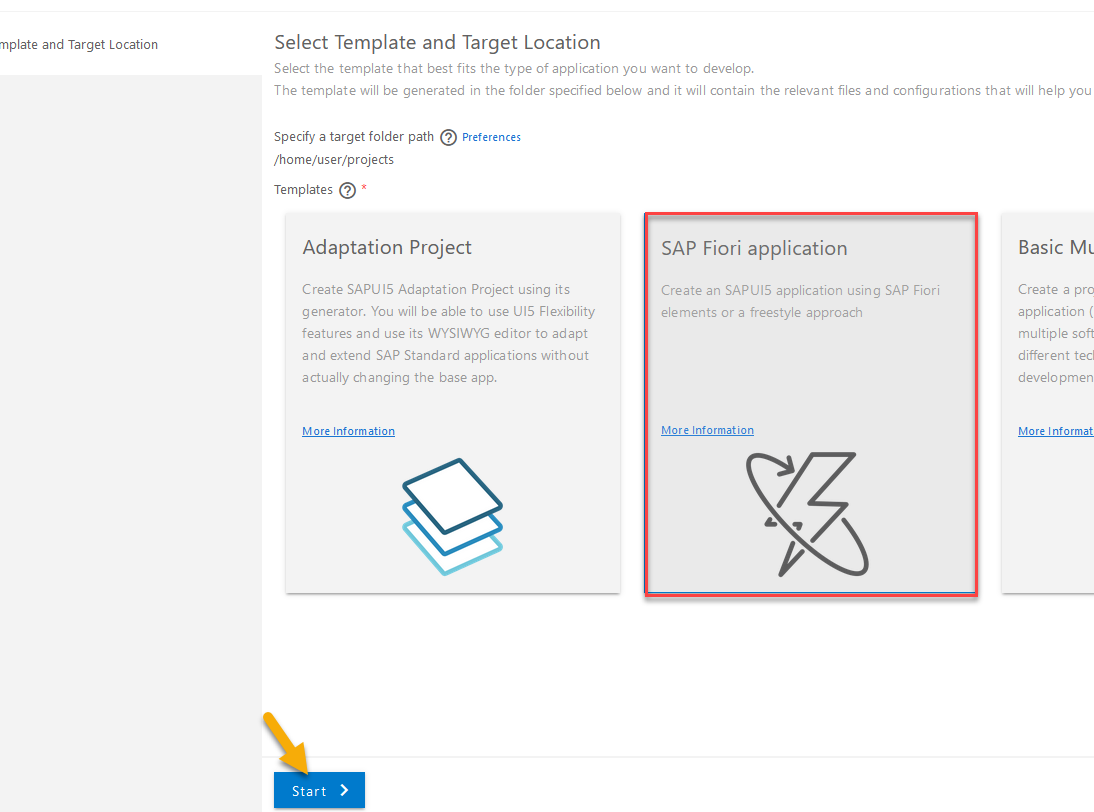
Note: In Step 3, choose SAP Fiori instead of Full Stack Cloud Application

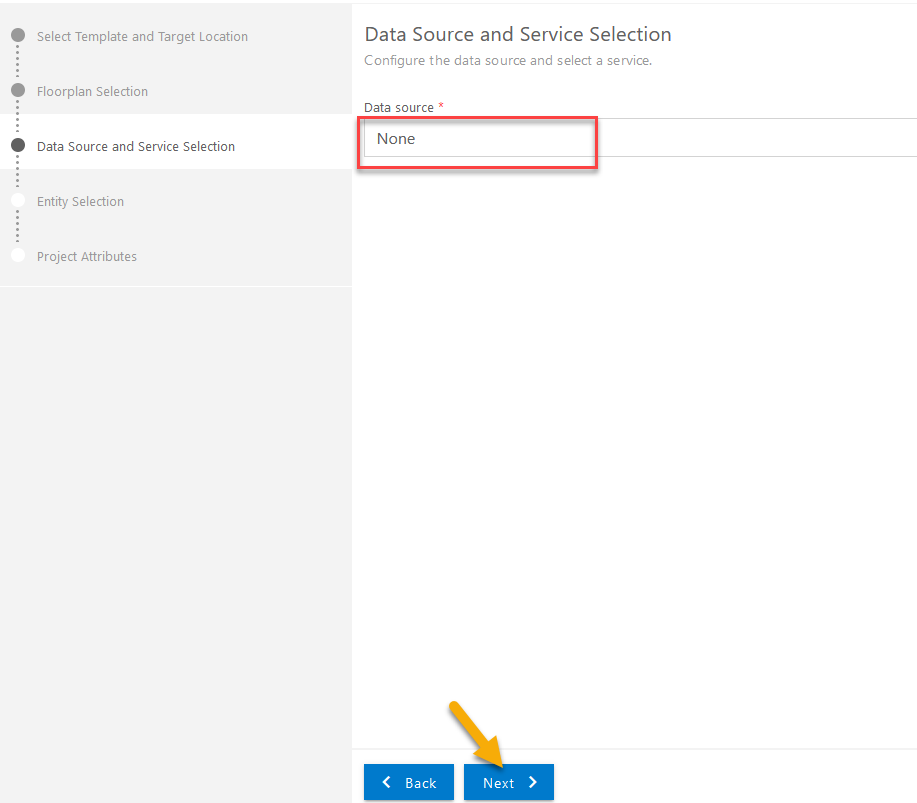
### Create a sample Fiori App

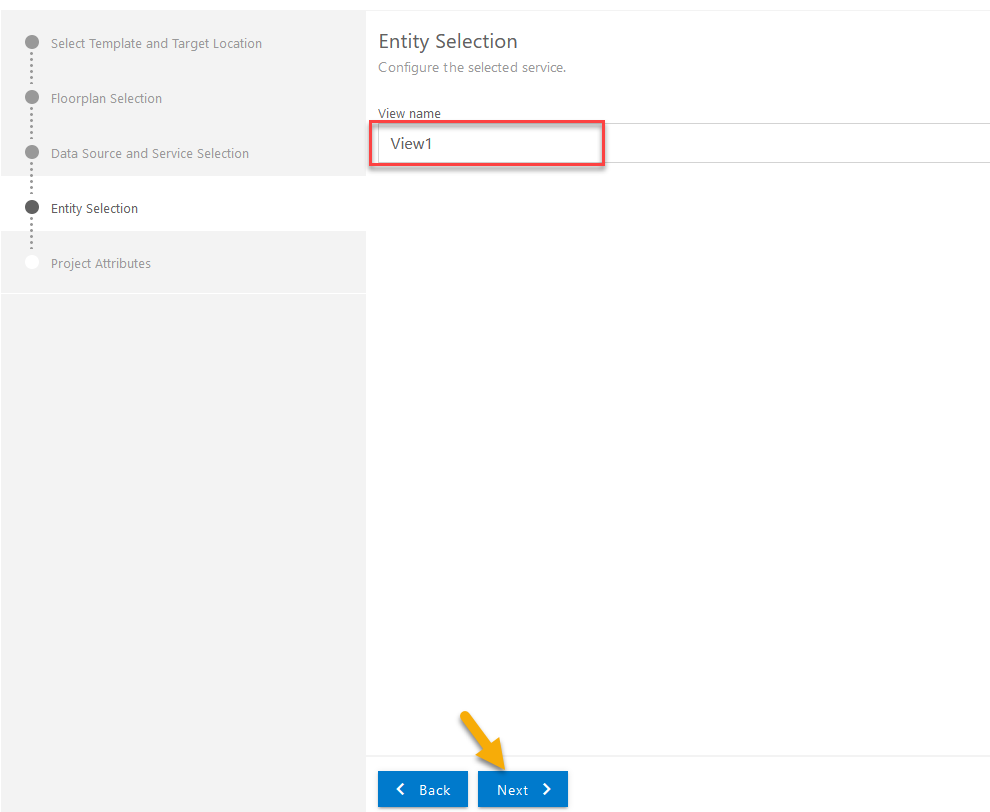
Let’s create a new project.

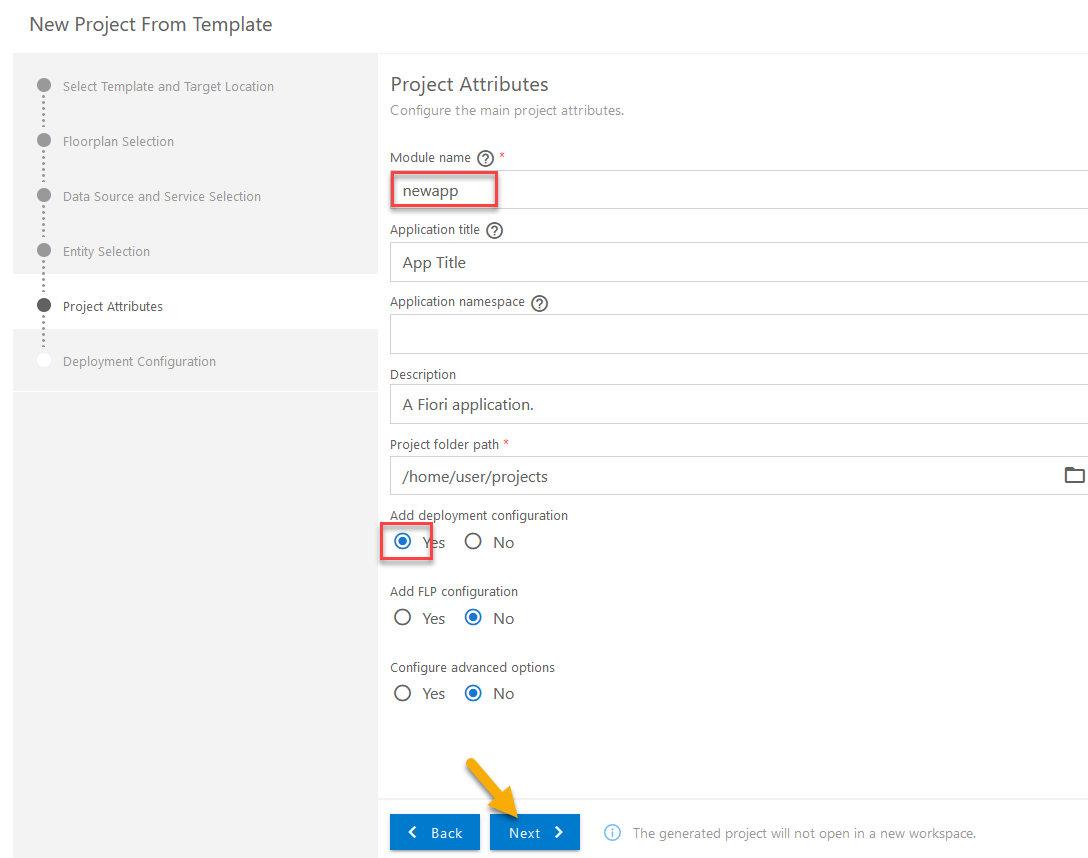


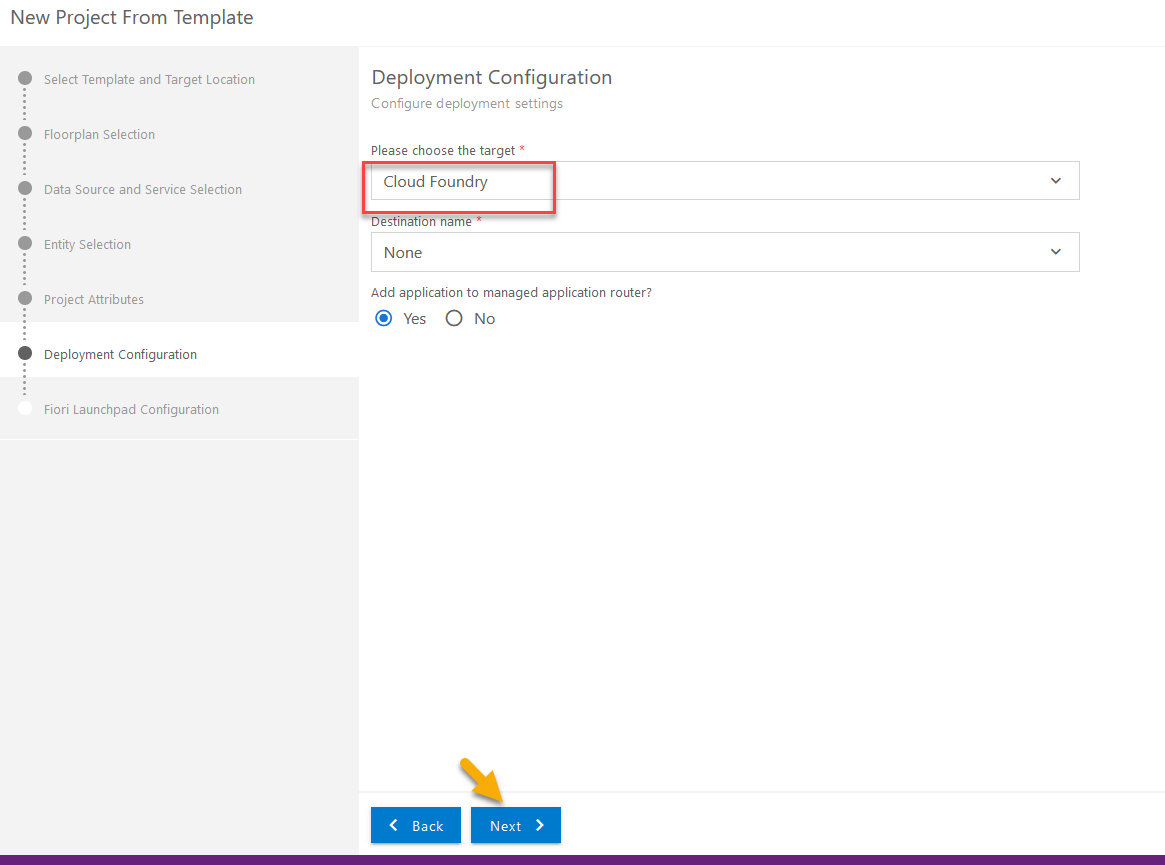


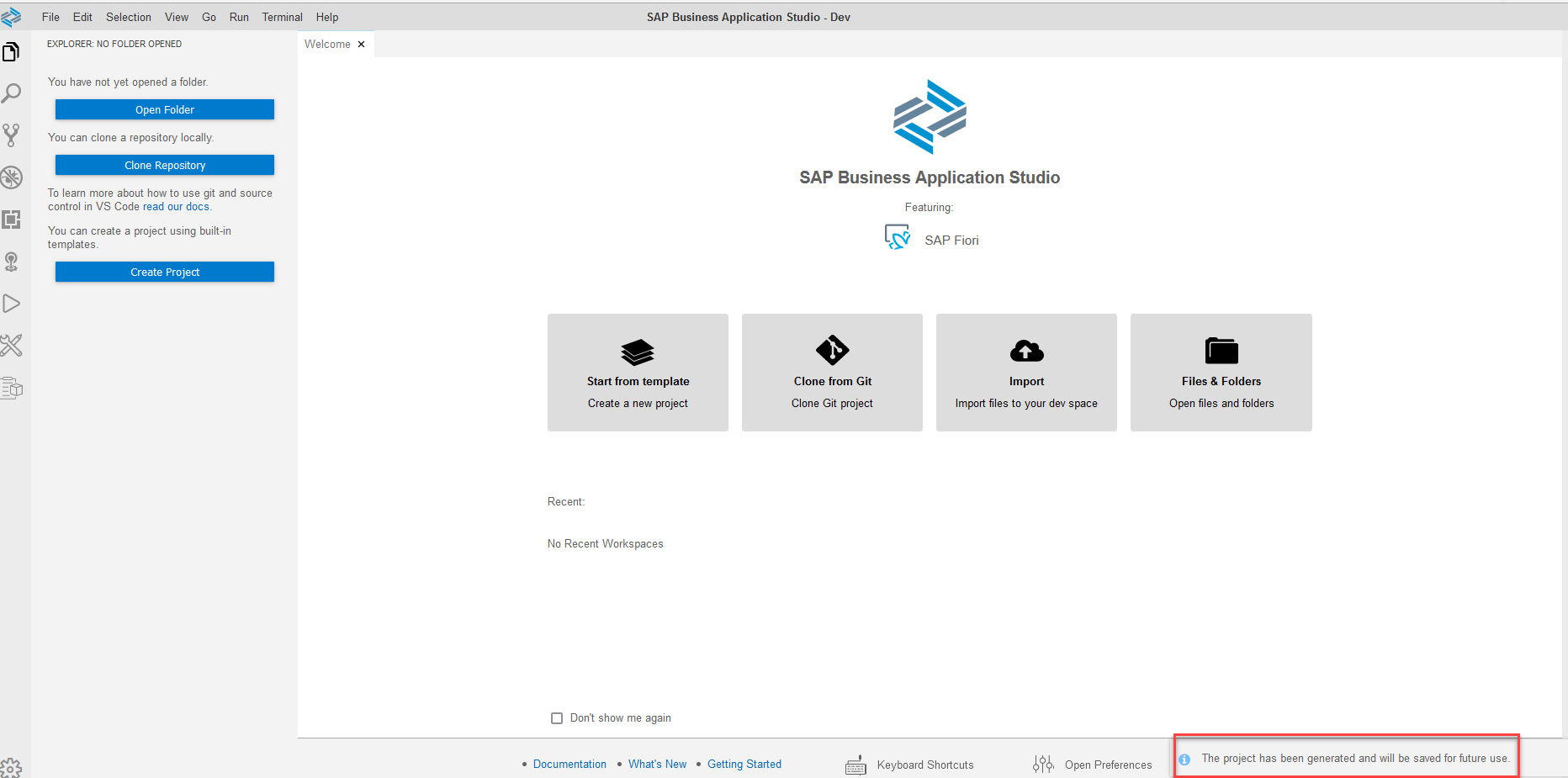




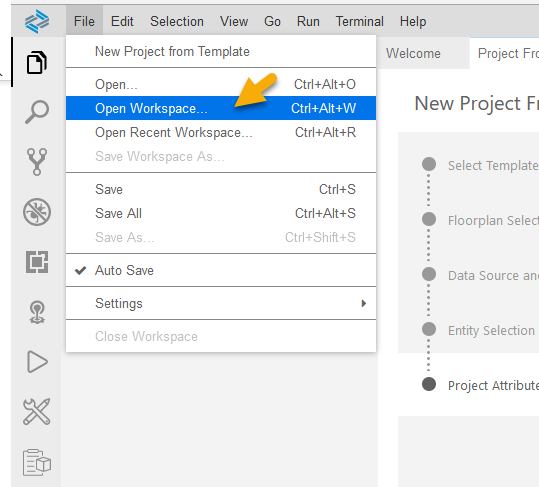


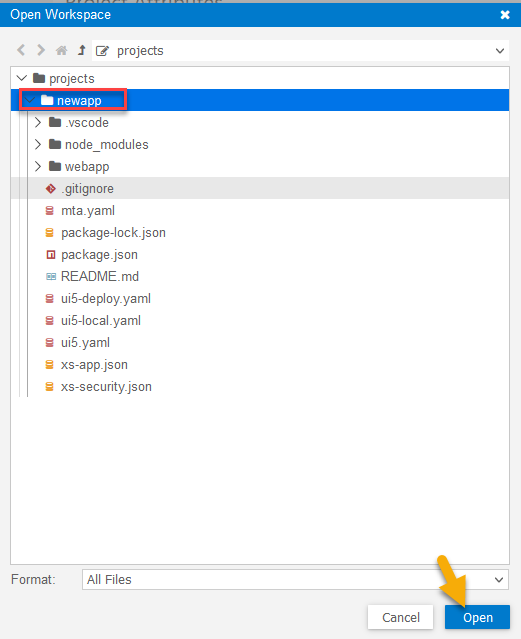






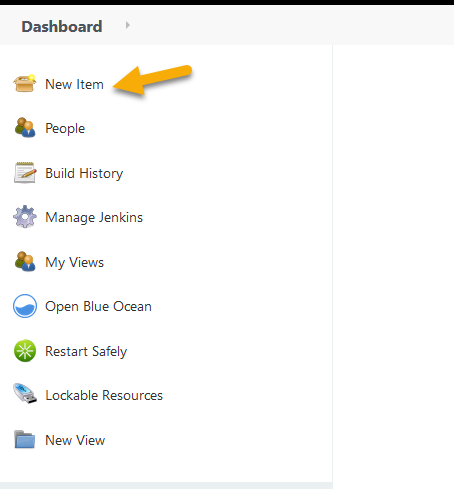
Open the new project.

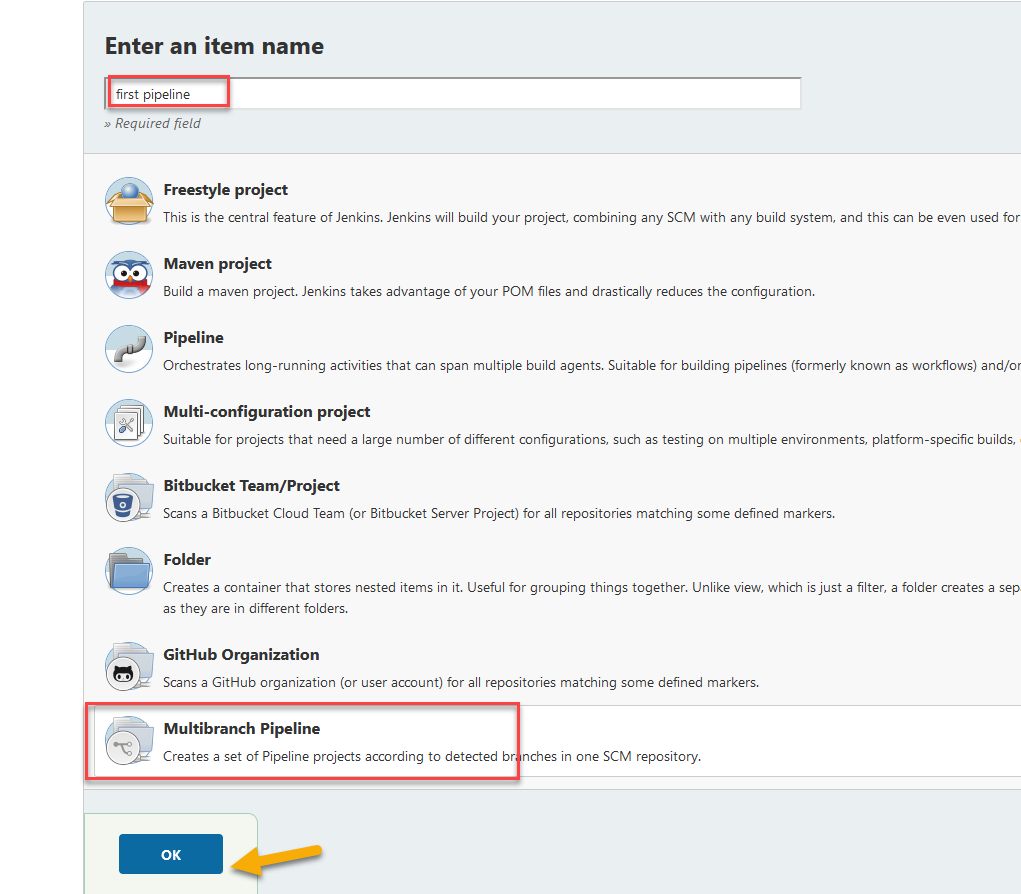


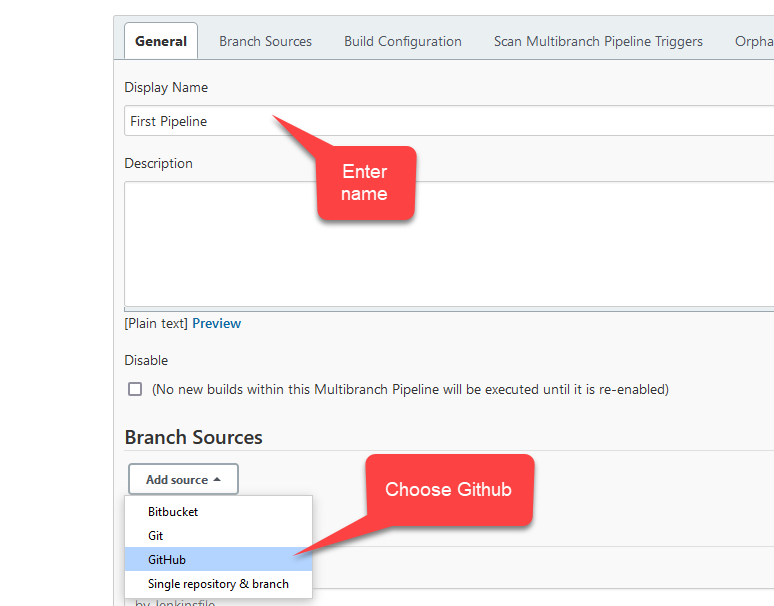


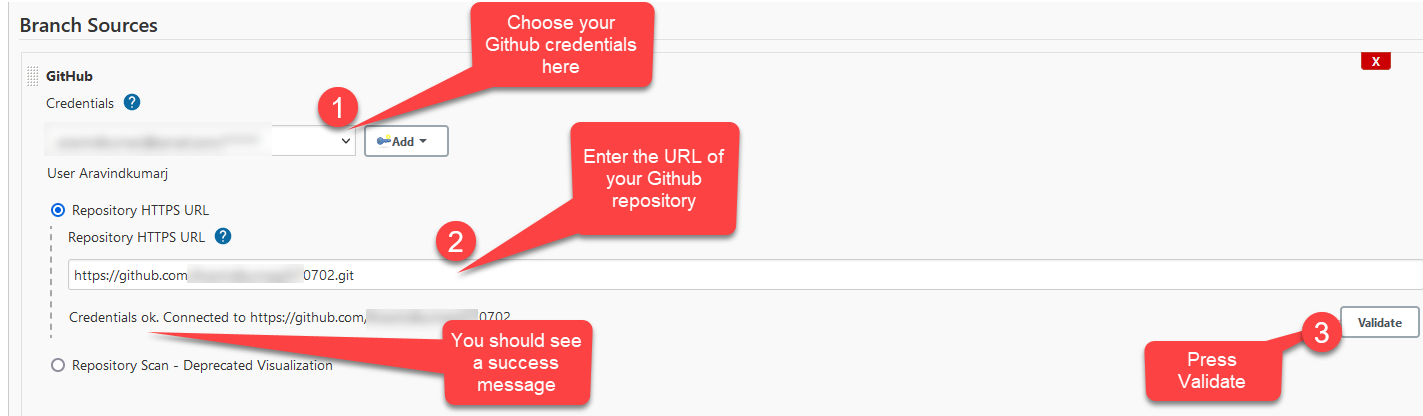
## Create a Job

Login to Jenkins and click on ‚New Item‘.

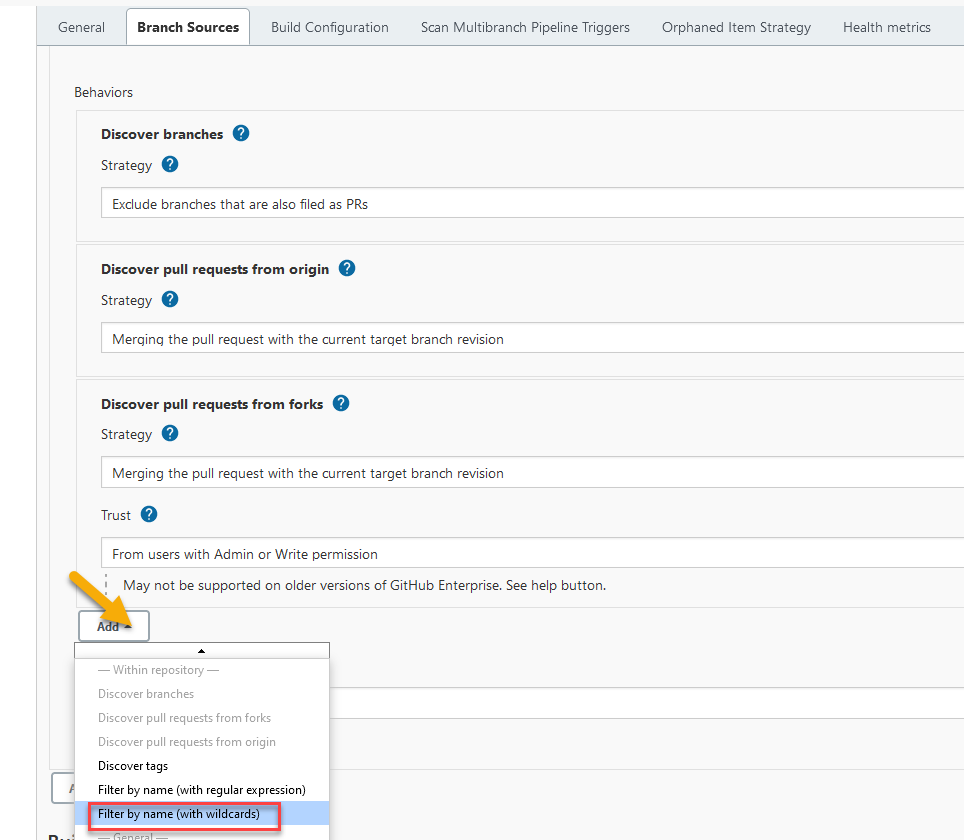


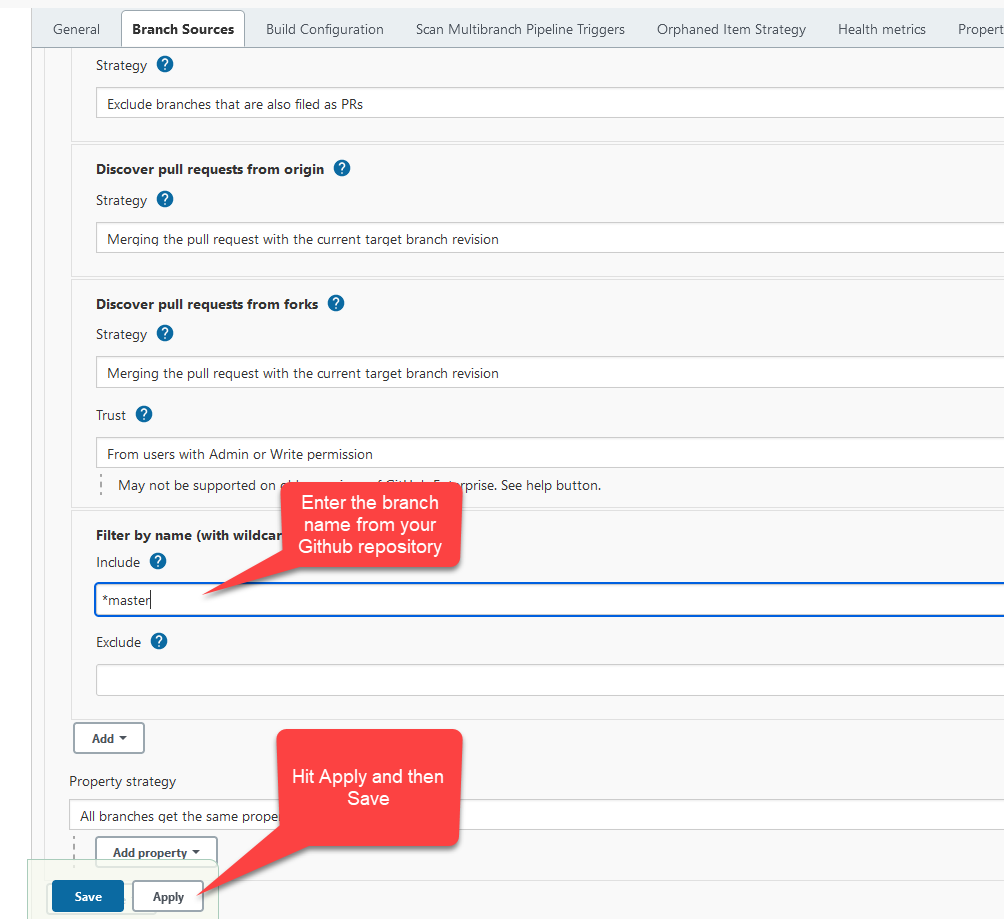






We will now select the branches in scope.





As soon as you hit Save, the job starts running and you can see the status by clicking on Status tab. In this picture you see the job is running.

