8.1 Deploying a Python Application

This section will guide you to:

* Connect and deploy a python application to an application server.

This guide has three subsections, namely:

8.2.1 Login to Jenkins

8.2.2 Create Jenkins job for Python

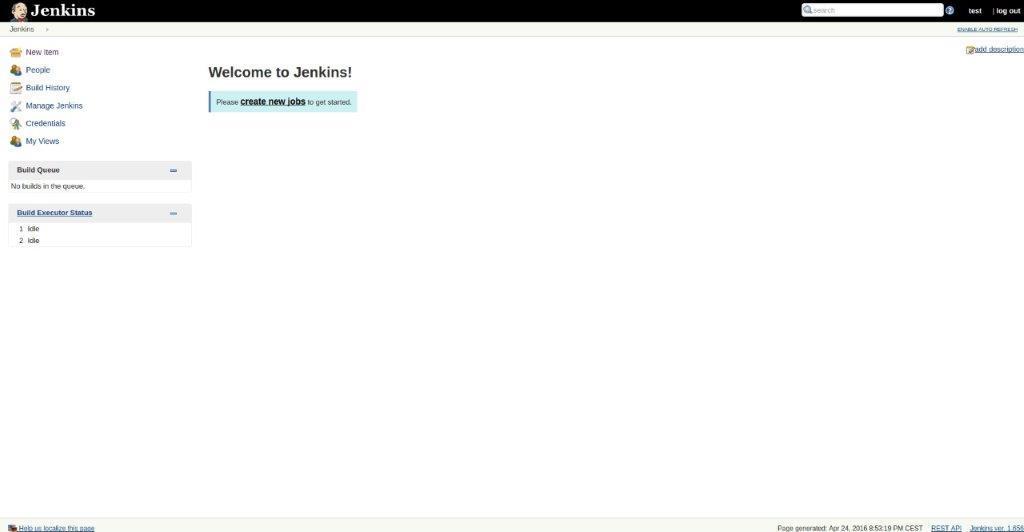
8.2.3 Push code to GitHub repositories

**Step 8.2.1:** Login to Jenkins

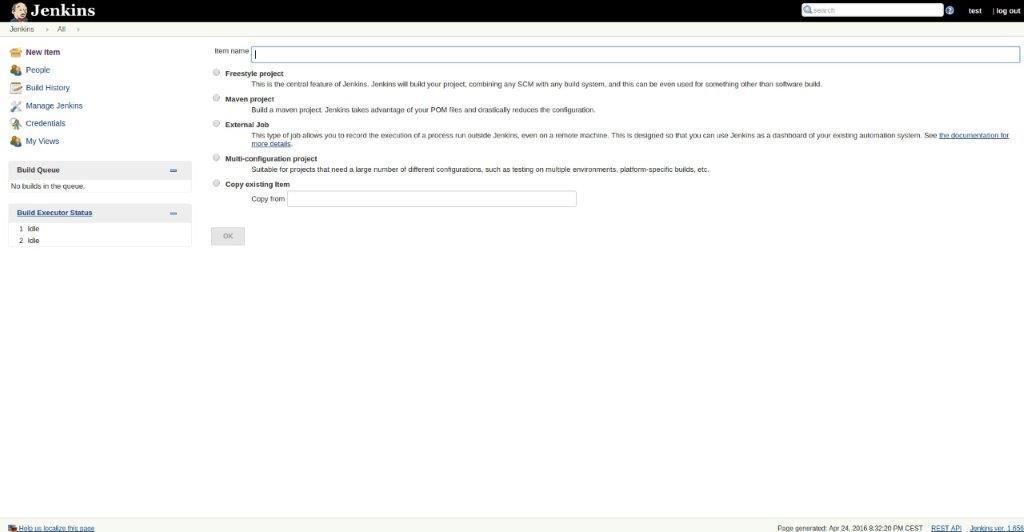
* Open your browser and navigate to **localhost:8081**
* Provide your username and password, and click on **Login.**

**Step 8.2.2:** Create Jenkins job for Python

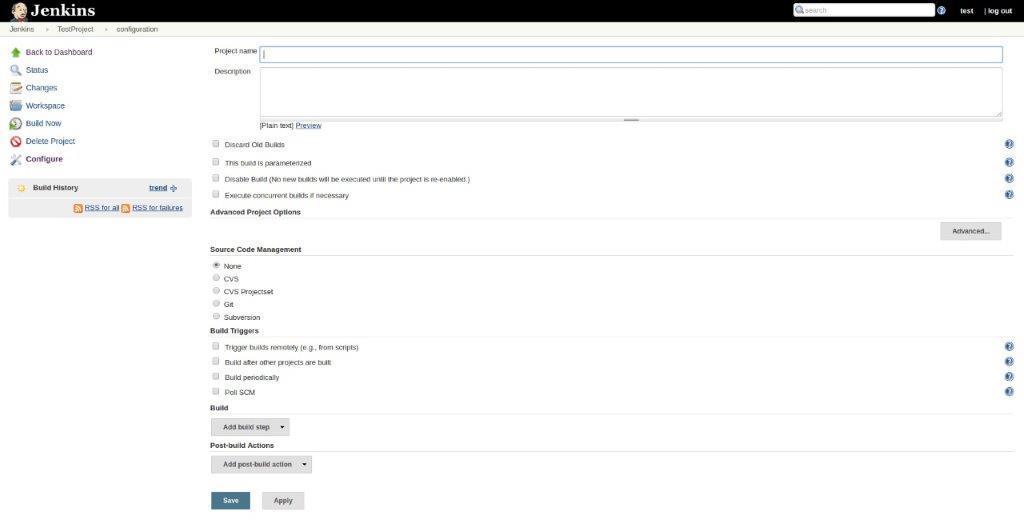
* Enter the Jenkins dashboard and click **New Item**on the left menu as shown below:



* Enter the item name and select the type of the project as freestyle project.



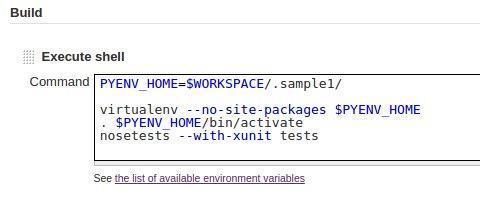
* Ensure a file containing the list of dependencies is present in your source under **/**envs.



* Enter the code repository URL under the **Source Code Management** as shown below:



* Under the **Build**section, enter the shell script shown below to create or recreate the virtual environment for your web application and install the required libraries.



***PYENV\_HOME=$WORKSPACE/.sample1/  
if [ –d $PYENV\_HOME ]; then  
   rm –rf $PYENV\_HOME  
fi***

***virtualenv —no–site–packages $PYENV\_HOME  
. $PYENV\_HOME/bin/activate  
pip install –r envs/req.txt***

* Create a job and build.

**Step 8.2.3:** Push the code to your GitHub repositories

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your Git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**