Module : 06
Jenkins Exercise

Name : Anubhav Ranjan   *Registration No : 774*

College : IIIT kalyani   *Roll No : ECE/21114*

Exercise 3: Freestyle Job - Flask Application Deployment

1. Objective: Automate the deployment of a Flask application.
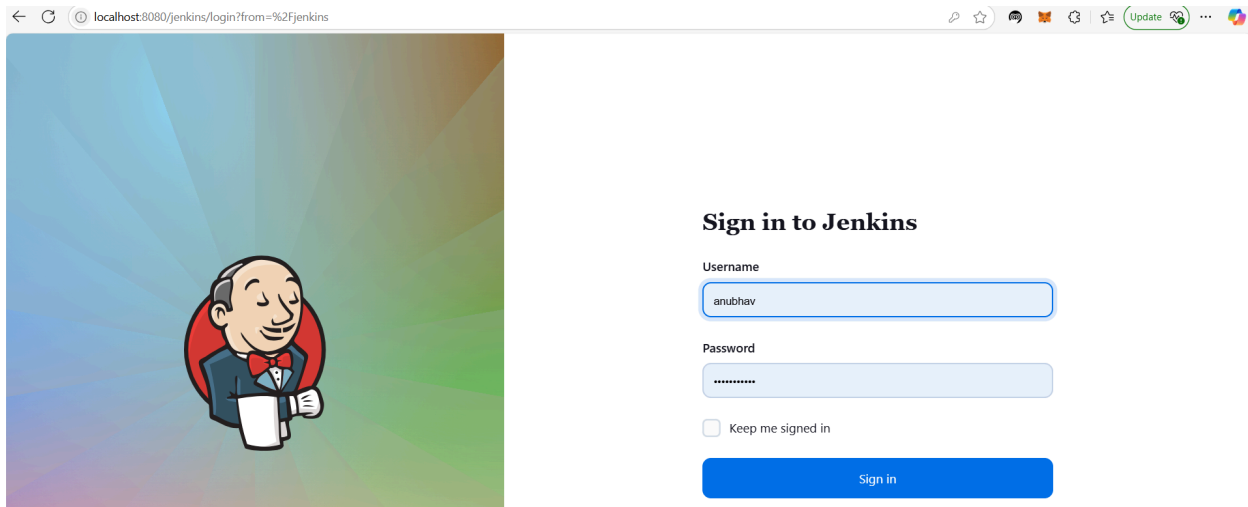
2. Steps:

- Create a Freestyle Job.
- Configure the job to:
    - ▪ Pull the Flask application repository from GitHub.
    - ▪ Set up a virtual environment using Python:

        python -m venv venv

        .\venv\Scripts\activate

        pip install -r requirements.txt

    - ▪ Start the Flask development server.
- Add a post-build action to verify the server is running (e.g., using curl or a similar tool to hit a test endpoint).

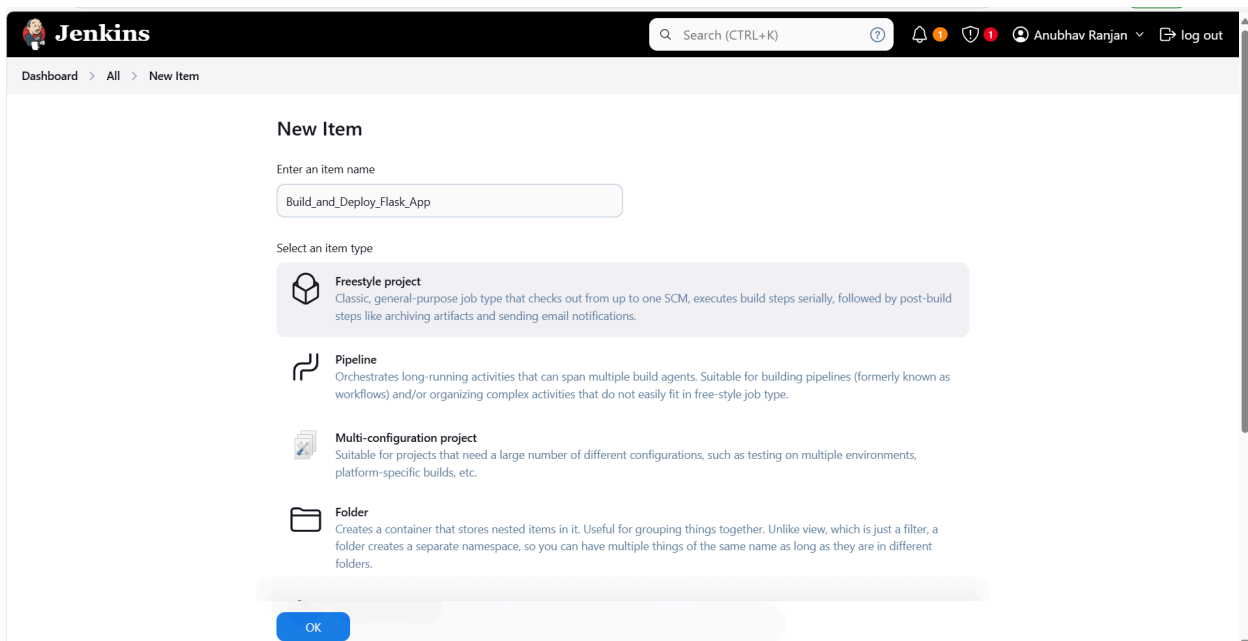**3. Task:** Test the application deployment by accessing it via localhost.

Flask Project GitHub Repo Link:

https://github.com/feelinganubhav/Library-Management-System.git

Let's complete this exercise step by step to build and deploy a Flask-based Library Management System using Jenkins.

Step 1: Prerequisites

Before proceeding, ensure the following:

1. **Jenkins is installed and running.**
2. **Python installed** on the Jenkins server. Verify by running:
   ```
   python --version
   ```


3. A **Flask project repository** (i.e. **LMS**) is available on GitHub.

Step 2: Create a Freestyle Job in Jenkins

1. **Open Jenkins:**

   - Start the Apache Tomcat Server from Services on your Windows

   - Or Open the Terminal..Navigate to the directory where jenkins.war file is present and Run the command:

```
java -jar .\jenkins.war
```


   - Launch Jenkins in your browser (usually `http://localhost:8080` or `http://localhost:8080/jenkins`).
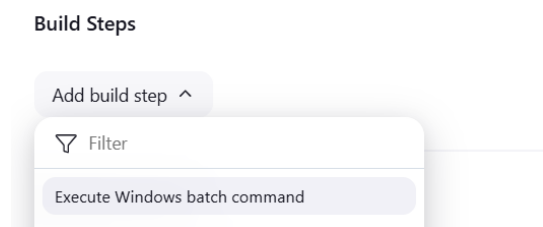   - Log in with your credentials.

2. **Create a New Freestyle Job:**
   ○ Click on **"New Item"** from the Jenkins dashboard.
   ○ Enter a name for the job, i.e.
     ▪ `Build_and_Deploy_Flask_App`
   ○ Select **"Freestyle project"** and click **OK**.

3. **Add a Description :**
    ○ On the job configuration page, add a brief description such as:

Description

This Jenkins Freestyle Job automates the deployment of a Flask-based Library Management System. The job pulls the application's code from GitHub, sets up a Python virtual environment, installs dependencies, and starts the Flask development server. A post-build action verifies the server's functionality by hitting a test endpoint, ensuring the application is successfully deployed and accessible via localhost.

Plain text Preview

Step 3: Configure the Job

Step 3.1: Clone the GitHub Repository

1. In the job configuration page, go to the **Source Code Management** section.
2. Select **Git** and enter the repository URL (i.e., `https://github.com/feelinganubhav/Library-Management-System.git`).
3. (Optional) If the repository is private, add your credentials.

Dashboard > Build_and_Deploy_Flask_App > Configuration

Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

Source Code Management

○ None

● Git ?

Repositories ?

Repository URL ?   ×

https://github.com/feelinganubhav/Library-Management-System.git

Credentials ?

- none -   ⌄

+ Add

Advanced ⌄

Add Repository

Step 3.2: Setup the Environment and Install Dependencies

1. **Scroll to the "Build" Section:**
   - Click **"Add Build Step"** → Select **"Execute Windows batch command"**.

Build Steps

Add build step ⌃

▽ Filter

Execute Windows batch command

2. Add the following commands to setup the Environment:

```
echo "Setting up the environment..."

:: Create a virtual environment if it doesn't already exist

if not exist venv (

    python -m venv venv

)

:: Activate the virtual environment

call .\venv\Scripts\activate

:: Install dependencies

if exist requirements.txt (

    pip install -r requirements.txt

) else (

    echo "ERROR: requirements.txt not found!"

    exit 1

)

call .\venv\Scripts\deactivate
```

Build Steps


```
Execute Windows batch command   ?                                        ×

Command
See the list of available environment variables

echo "Setting up the environment..."

:: Create a virtual environment if it doesn't already exist
if not exist venv (
    python -m venv venv
)

:: Activate the virtual environment
call .\venv\Scripts\activate

:: Install dependencies
if exist requirements.txt (
    pip install -r requirements.txt
) else (
    echo "ERROR: requirements.txt not found!"
    exit 1
)

call .\venv\Scripts\deactivate
```

Step 3.4: Deploy the Project

1. Click **Add build step** > **Execute Windows batch command**.
2. Add the following commands to Start the Flask Development Server in the Background:

```
call .\venv\Scripts\activate

echo "Starting the Flask app..."

:: Start the Flask app in the background

set FLASK_APP=app.py

start /B python -m flask run --host=0.0.0.0 --port=5000

:: Wait for a few seconds to allow the Flask app to start

powershell -Command "Start-Sleep -Seconds 3"
```

3. **Verify** the **Server** is Running - using **curl** to hit all the **endpoints**:
   ○ Home, Books, and Members..

Save   Apply

4. **Save Your Job:**
   ● Scroll to the bottom and click "Save".

Step 5: Run and Test the Job

1. Run the Job:
- Go to the job's dashboard.
- Click "**Build Now**" to execute the **Flask** Application **Deployment** and **Verification** task.
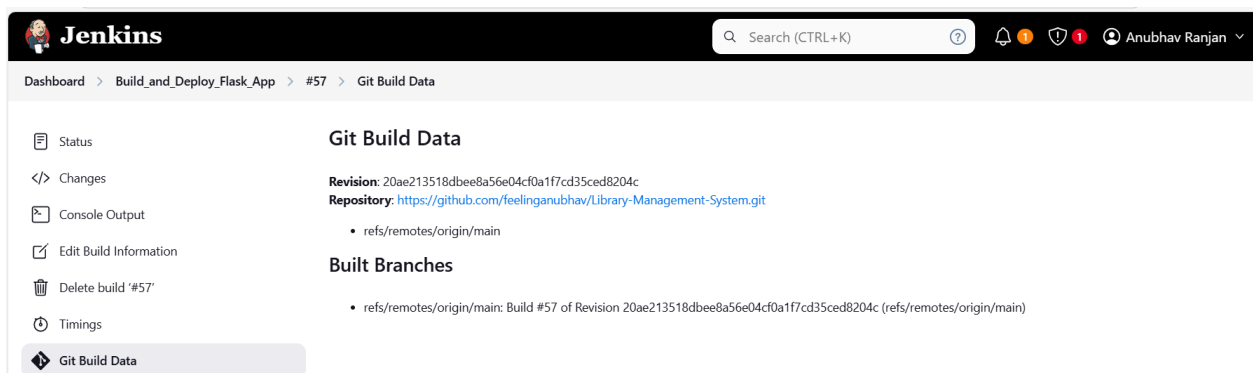
2. Verify the Output:
- Check the Console Output in Jenkins for the success message:

GitHub Repository fetched and Cloned Sucessfully:


```
Started by user Anubhav Ranjan
Running as SYSTEM
Building in workspace D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App
The recommended git tool is: NONE
No credentials specified
 > git.exe rev-parse --resolve-git-dir D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App\.git # timeout=10
Fetching changes from the remote Git repository
 > git.exe config remote.origin.url https://github.com/feelinganubhav/Library-Management-System.git # timeout=10
Fetching upstream changes from https://github.com/feelinganubhav/Library-Management-System.git
 > git.exe --version # timeout=10
 > git --version # 'git version 2.38.1.windows.1'
 > git.exe fetch --tags --force --progress -- https://github.com/feelinganubhav/Library-Management-System.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
 > git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 20ae213518dbee8a56e04cf0a1f7cd35ced8204c (refs/remotes/origin/main)
 > git.exe config core.sparsecheckout # timeout=10
 > git.exe checkout -f 20ae213518dbee8a56e04cf0a1f7cd35ced8204c # timeout=10
Commit message: "Merge branch 'main' of https://github.com/feelinganubhav/Library-Management-System"
 > git.exe rev-list --no-walk 20ae213518dbee8a56e04cf0a1f7cd35ced8204c # timeout=10
[Build_and_Deploy_Flask_App] $ cmd /c call C:\Users\ANUBHA~1\AppData\Local\Temp\jenkins1037246504571156296.bat
```

Virtual Environment Setup successfully and Requirements already Satisfied:

```
D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App>echo "Setting up the environment..."
"Setting up the environment..."

D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App>if not exist venv (python -m venv venv )

D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App>call .\venv\Scripts\activate
Requirement already satisfied: flask in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from -r
requirements.txt (line 1)) (3.0.3)
Requirement already satisfied: importlib-metadata>=3.6.0 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages
(from flask->-r requirements.txt (line 1)) (8.5.0)
Requirement already satisfied: Jinja2>=3.1.2 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from flask-
>-r requirements.txt (line 1)) (3.1.5)
Requirement already satisfied: click>=8.1.3 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from flask->-
r requirements.txt (line 1)) (8.1.8)
Requirement already satisfied: itsdangerous>=2.1.2 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from
flask->-r requirements.txt (line 1)) (2.2.0)
Requirement already satisfied: Werkzeug>=3.0.0 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from
flask->-r requirements.txt (line 1)) (3.0.6)
Requirement already satisfied: blinker>=1.6.2 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from flask-
>-r requirements.txt (line 1)) (1.8.2)
Requirement already satisfied: colorama in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from
click>=8.1.3->flask->-r requirements.txt (line 1)) (0.4.6)
Requirement already satisfied: zipp>=3.20 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from importlib-
metadata>=3.6.0->flask->-r requirements.txt (line 1)) (3.20.2)
Requirement already satisfied: MarkupSafe>=2.0 in d:\jenkins_new_home\workspace\build_and_deploy_flask_app\venv\lib\site-packages (from
Jinja2>=3.1.2->flask->-r requirements.txt (line 1)) (2.1.5)

[notice] A new release of pip is available: 23.0.1 -> 24.3.1
[notice] To update, run: python.exe -m pip install --upgrade pip
[Build and Deploy Flask App] $ cmd /c call C:\Users\ANUBHA~1\AppData\Local\Temp\jenkins14364796549755961245.bat
```

Flask Application Deployed Sucessfully: App is Up and Running...

```
D:\jenkins_New_Home\workspace\Build_and_Deploy_Flask_App>call .\venv\Scripts\activate
"Starting the Flask app..."
 * Serving Flask app 'app.py '
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://192.168.43.49:5000
Press CTRL+C to quit
```

All Routes/Endpoints of Library Management System are Verified Sucessfully..

```
"Verifying Flask App Routes..."
127.0.0.1 - - [26/Jan/2025 00:08:01] "GET / HTTP/1.1" 200 -
HTTP Code: 200
{"message":"Welcome to the Library Management System API!"}
Home route verified successfully!
127.0.0.1 - - [26/Jan/2025 00:08:01] "GET /books/ HTTP/1.1" 200 -
HTTP Code: 200
[{"author":"R. K. Narayan","book_id":1,"category":"Non-Fiction","status":"borrowed","title":"The Guide"},{"author":"Robert T.
Kiyosaki","book_id":2,"category":"Non-Fiction","status":"borrowed","title":"Rich Dad Poor Dad"},{"author":"Aravind
Adiga","book_id":3,"category":"Fiction","status":"borrowed","title":"The White Tiger"},{"author":"Arundhati Roy","book_id":4,"category":"Non-
Fiction","status":"borrowed","title":"God of Small Things"},{"author":"Napolean Hill","book_id":5,"category":"Non-
Fiction","status":"available","title":"Think and Grow Rich"}]
books route verified successfully!
127.0.0.1 - - [26/Jan/2025 00:08:01] "GET /members/ HTTP/1.1" 200 -
HTTP Code: 200
[{"borrowed_books":["The Guide"],"member_id":1,"membership":"Premium","name":"Anubhav Ranjan"},{"borrowed_books":
[],"member_id":2,"membership":"Premium","name":"Shirish Manoj Bobde"},{"borrowed_books":["Rich Dad Poor Dad","The White Tiger","God of Small
Things"],"member_id":3,"membership":"Regular","name":"Rushil Ranjan"},{"borrowed_books":["Harry Potter and the prisoner of
Azkaban"],"member_id":4,"membership":"Regular","name":"Sundeep Nayak"},{"borrowed_books":
[],"member_id":5,"membership":"Regular","name":"Divyanshu Kumar"}]
members route verified successfully!
SUCCESS: The process "python.exe" with PID 56664 has been terminated.
SUCCESS: The process "python.exe" with PID 95188 has been terminated.
All routes verified successfully!
Finished: SUCCESS
```