Program 8: Set Up a Jenkins CI Pipeline for a Maven Project, Use Ansible to Deploy Artifacts Generated by Jenkins

Step 1: Create a simple Maven 'HelloWorld' project using either Eclipse IDE or the terminal and Run a test build.

Step 2: Create a Jenkins Job

- **Go to Jenkins Dashboard** → Click **New Item** → **Freestyle Project**
- **Enter Job Name:** `HelloWorld-Maven-Build`
- **Select** `Git` as Source Code Management
    - **Repository URL**: Use a local Git repo or push your Maven project to GitHub and use that link.

Step 3: Configure Build

- Under **Build,** add:

```
mvn clean package
```

- Under Post-build Actions, select:
    - Archive the artifact: `target/helloworld-1.0-SNAPSHOT.jar`

Save and **Build** the Project.

Step 4: Create an Ansible Playbook for Deployment

Create the directory for the playbook:

```
mkdir -p ~/ansible-deploy
cd ~/ansible-deploy
nano deploy.yml
```

Add the following Ansible playbook:

```
---
- name: Deploy HelloWorld App
  hosts: local
  tasks:
    - name: Copy the JAR file to the deployment directory
      copy:
        src: /var/lib/jenkins/workspace/HelloWorld-Maven-
Build/target/helloworld-1.0-SNAPSHOT.jar
        dest: /opt/myapp/helloworld.jar
        mode: '0755'

    - name: Restart the application
      shell: "nohup java -jar /opt/myapp/helloworld.jar >
/opt/myapp/app.log 2>&1 &"
```

Save and exit.

Step 5: Integrate Ansible with Jenkins

- Open Jenkins Dashboard → Select **HelloWorld-Maven-Build** job
- Go to **Configure → Post-build Actions**
- Click "**Invoke Ansible Playbook**"
    - **Playbook Path**: `/home/user/ansible-deploy/deploy.yml`
    - **Inventory File**: `/etc/ansible/hosts`

  Save and Trigger the Build.