Lab 2. Getting Started with DevCs and Git

Background

Oracle Developer Cloud Service is a cloud-based software development Platform as a Service (PaaS) and a hosted environment for your application development infrastructure. It provides an open source standards-based solution to develop, collaborate, and deploy applications within Oracle Cloud.

An Oracle Developer Cloud Service project is a collection of Git repositories, branch merge requests, wikis, issues, deployment configurations, and builds.

Objectives:

- Open Oracle Developer Cloud Service Console and Launch an instance in Visual Builder Studio.
- Access Service Instance and create a Project in Oracle Developer Cloud Service
- Create a Git Repository in the Oracle DevCS Project
- Install Git and Clone an Oracle DevCS Project Git Repository on Local Machine
- Add the Sample Application Files to the Cloned Git Repository
- Commit and Push the Sample Application Files to the Project Git Repository

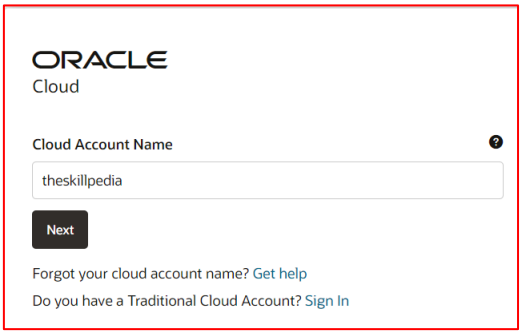
Pre-Requisite

- A web browser – Google chrome or Microsoft Edge
- An ***Oracle Cloud Account*** with an active Oracle Developer Cloud Service subscription
- Your Oracle Cloud account credentials
- An Oracle Cloud account with an active Oracle Developer Cloud Service subscription
- Git

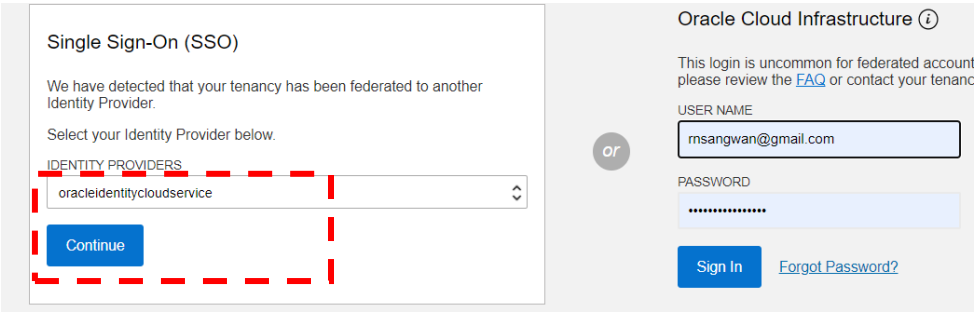
Sequence 1. Open Oracle Developer Cloud Service Console

You can log on to Oracle Developer Cloud Service from the Oracle Cloud Dashboard page.

1. In a web browser, go to https://www.oracle.com/index.html, and click ***View Accounts***. Under ***Cloud Account***, Click on ***Sign in to Cloud***.
2. Under Cloud Account Name, Enter your tenancy Name. Refer to Screen Shot.

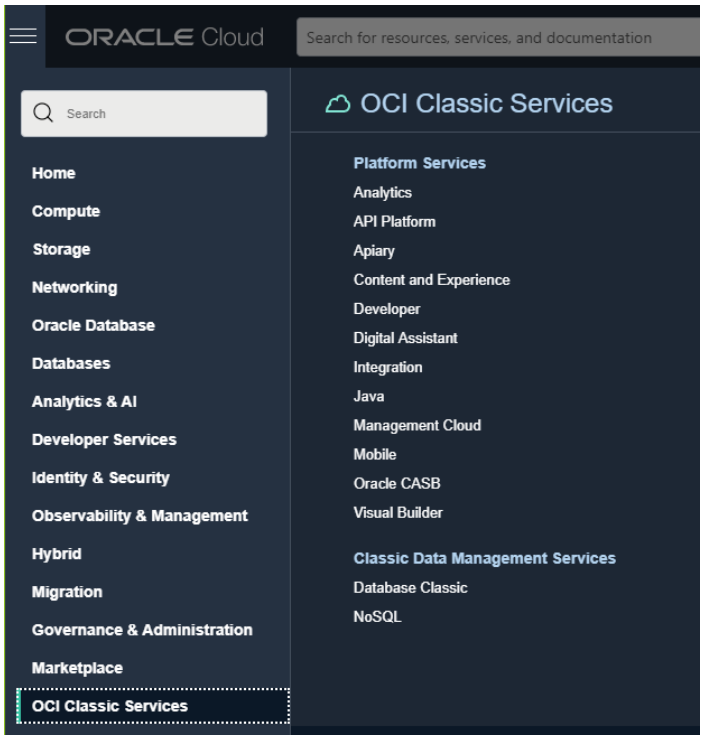


3. From the page that open, under Single Sign On, Click Continue Button.

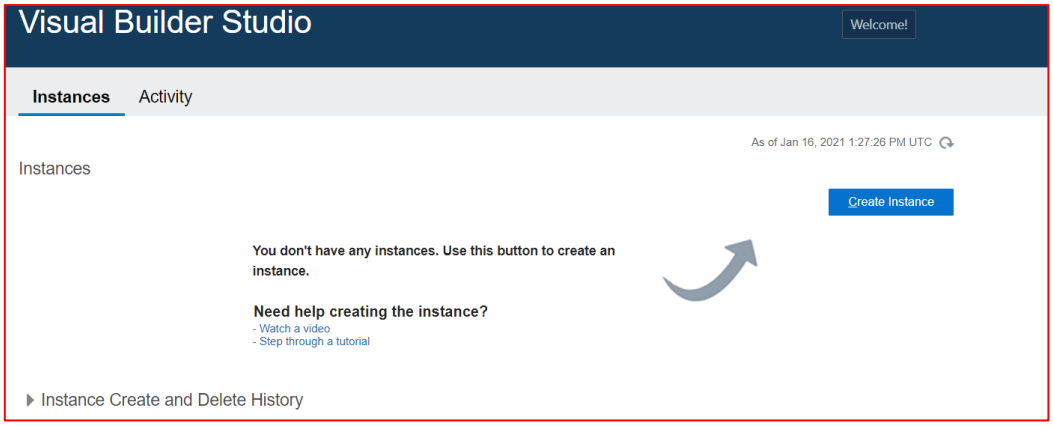


4. It will open a popup. Enter your login credentials for SSO and click **Sign In.**

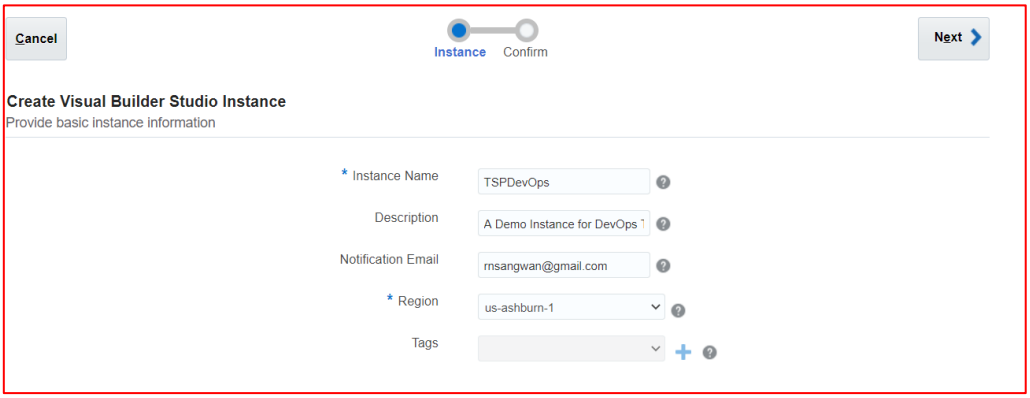
5. From Left Menu Select **OCI Classic Services** -> **Developer**



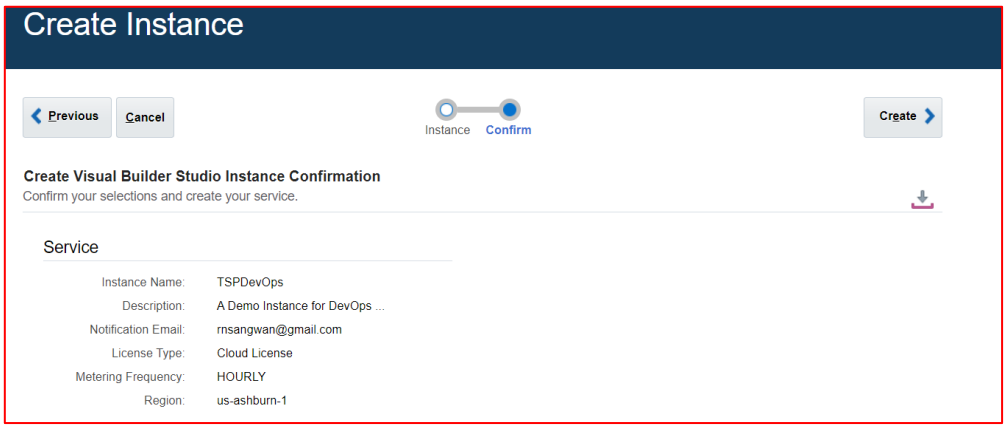
6. It will launch Visual Builder Studio on a separate Tab. Click on **Create Instance** as given in screen shot below



7. Fill in the details and Launch the Instance.
   - **Instance Name :** TSPDevOps
   - **Description :** A Demo Instance for DevOps Training
   - **Notification Email :** <your Email ID>
   - **Region : <Your Region>**

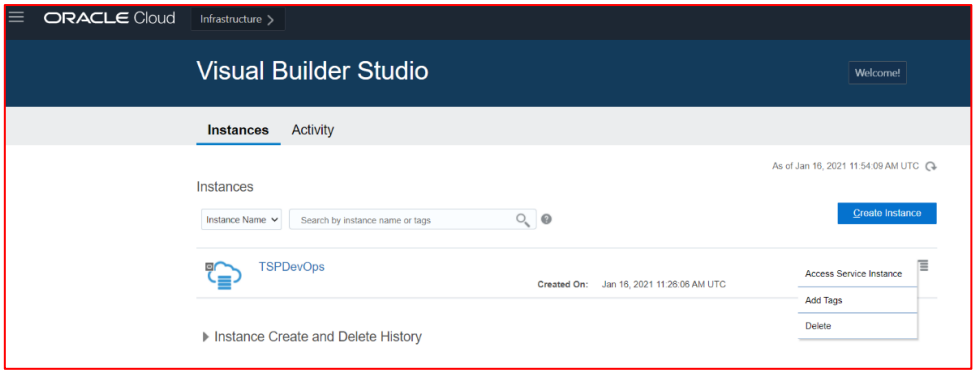


8. Click on **Next**. It will show a summary screen. Click on **Create**.

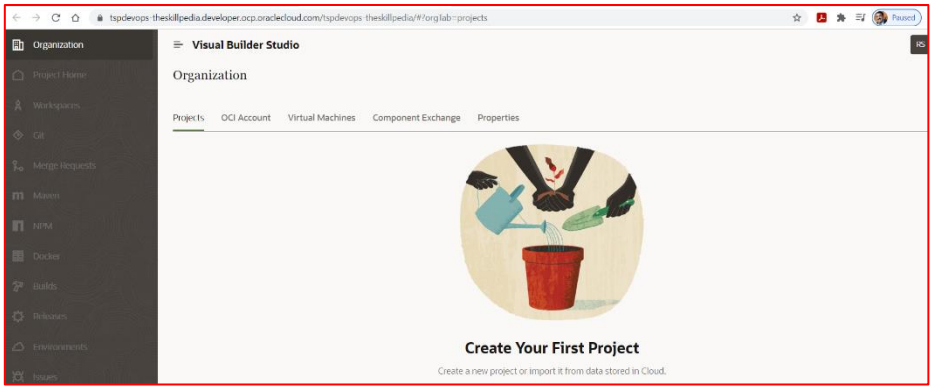


Sequence 2. Access Service Instance and create a Project

1. Once the Instance is ready, Click on Icon on Extreme Right and Select **Access Service Instance**

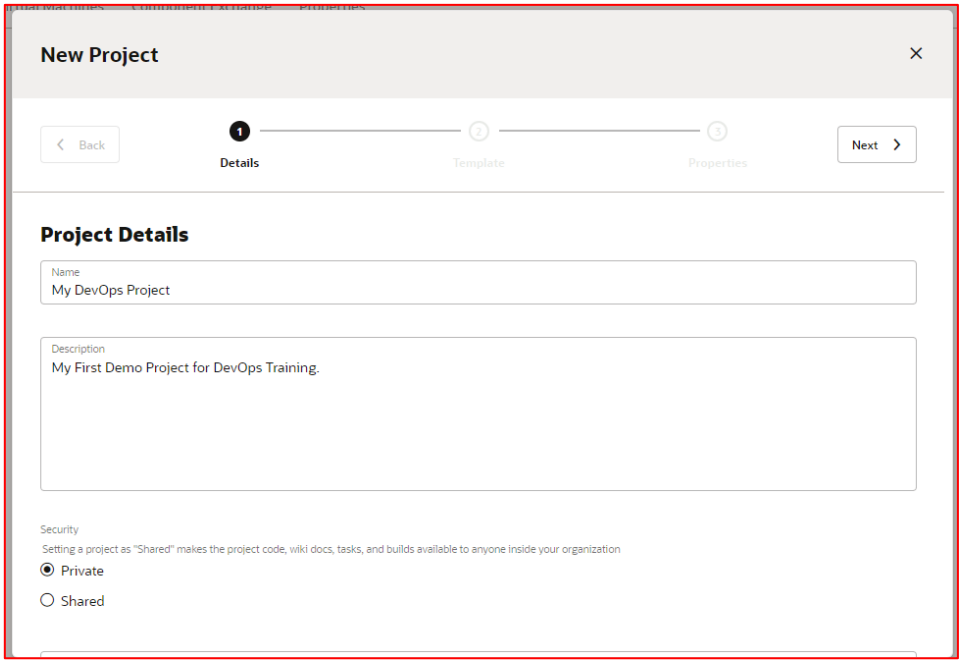


2. Create your First Project by clicking the link given in the centre.

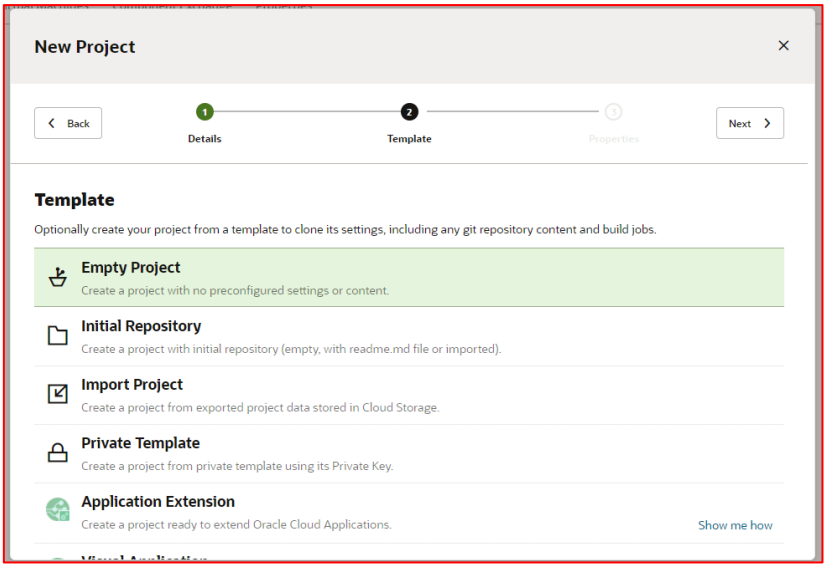


3. Fill in the Details as

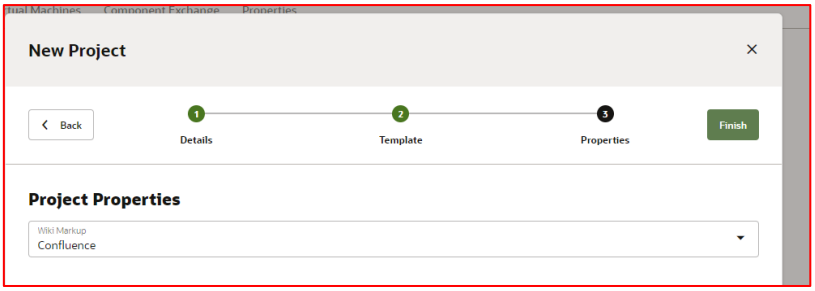
- **Name:** My DevOps Project
- **Description:** My First Demo Project for DevOps Training.
- **Security:** Private



4. Click on **Next**.



5. Select ***Confluence*** for Wiki Markup in ***Project Properties***.

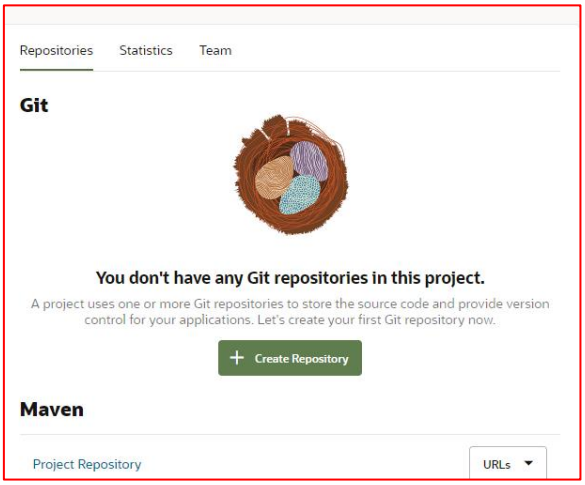


6. Click ***Finish***

Sequence 3. Create a Git Repository in the Oracle DevCS Project

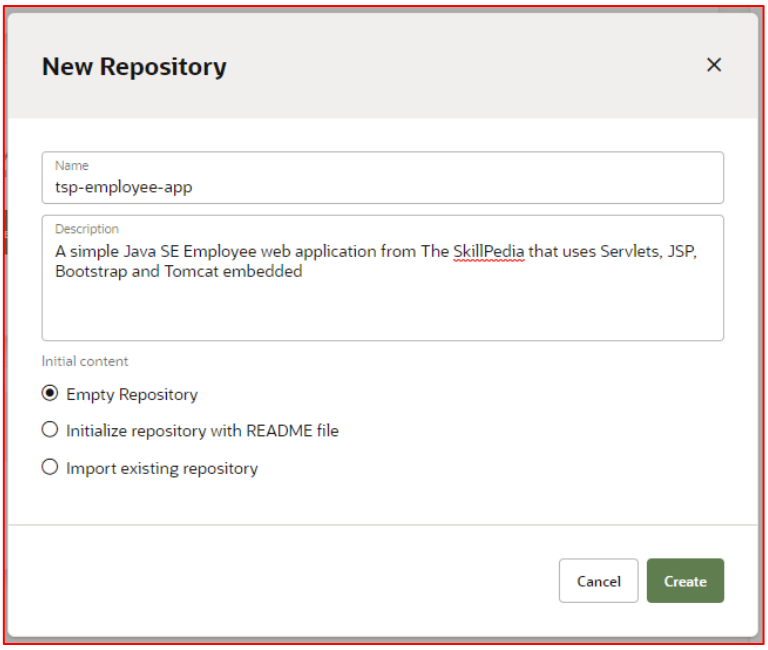
You need the project Owner role to create a Git repository in the project. If you do not have the project Owner role, ask the Owner to create a Git repository in the project.

1. On the Project page, if necessary, click **Repositories** on the right.
2. In the Repositories tab, click **+ New Repository.**

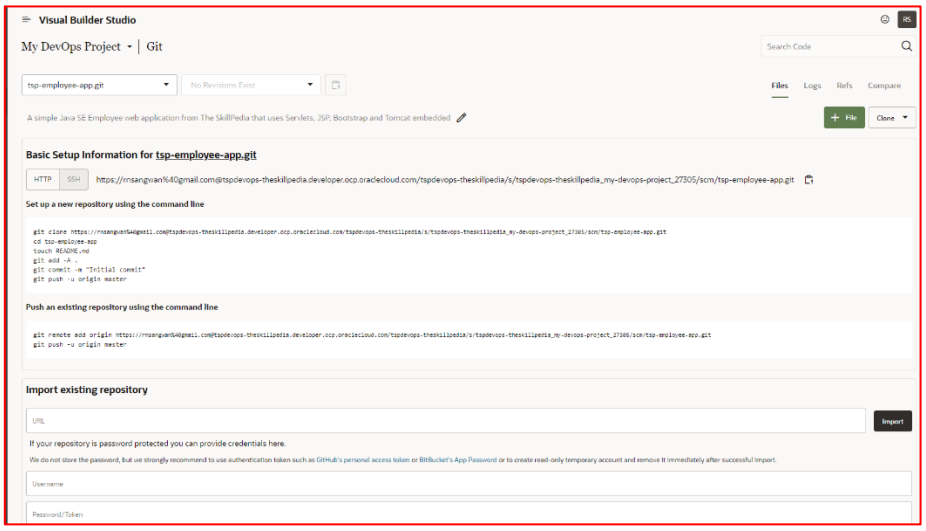


3. In the New Repository dialog box, enter the following values, and click Create.

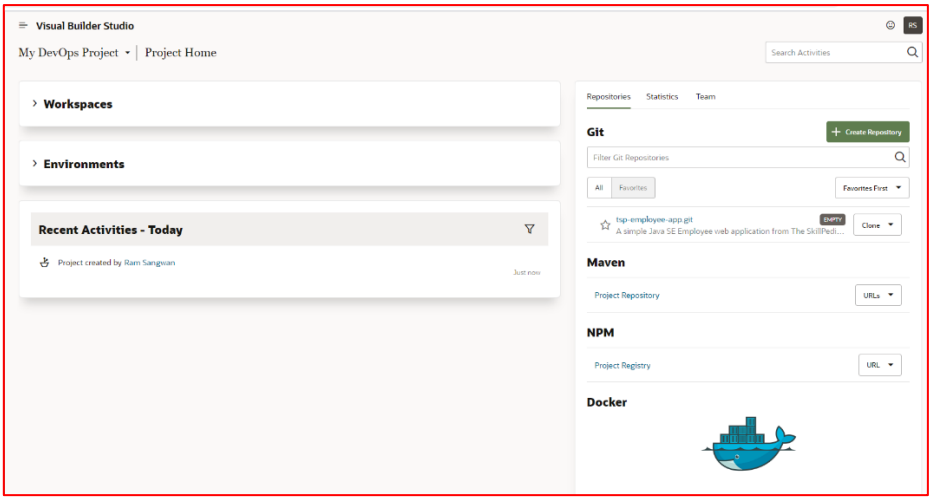
- **Name**: tsp-employee-app
- **Description**: A simple Java SE Employee web application from The SkillPedia that uses Servlets, JSP, Bootstrap and Tomcat embedded.
- **Initial Content**: **Empty Repository.**



*The **tsp-employee-app** Git repository will be created. The Code page opens showing the Git commands that you must run to clone the Git repository and populate it with a ReadMe file.*



4. In the navigation bar, click **Project.** The Project page shows the Git repository in the **Repositories** section and a notification in the **Recent Activities** feed.



Sequence 4. Install Git and Clone an Oracle DevCS Project Git Repository

You will use terminal to run the Git command that clones the project Git repository to your computer.

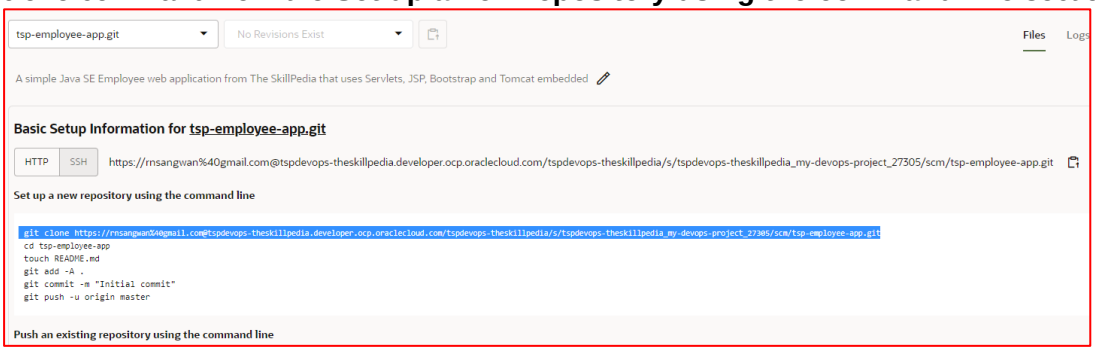
1. Start your Virtual Machine, **server,** and Login as root.
2. Open the terminal, install git and create a directory for the project

   [root@server ~]# yum install git wget -y

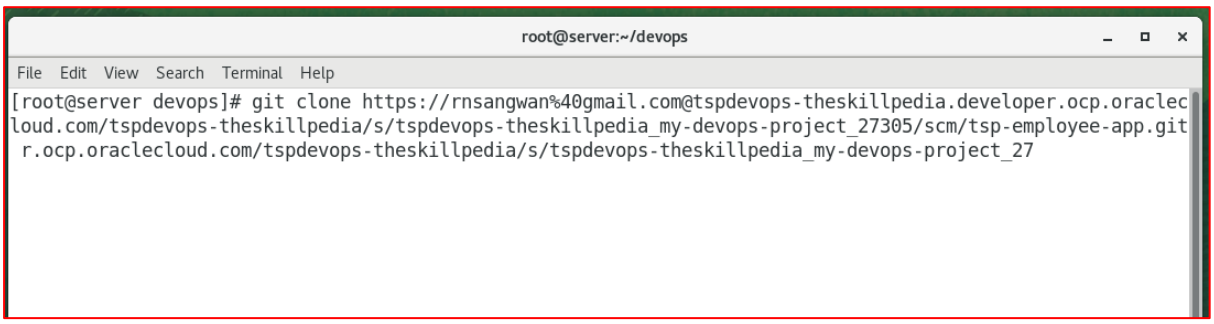
   [root@server ~]# mkdir devops

   [root@server ~]# cd devops

   [root@server devops]#
3. Open the **Code** page of the Oracle Developer Cloud Service project and copy the git clone command from the **Set up a new repository using the command line** section.



4. Paste the ***git clone*** command in the Git Bash prompt. Example:



Your command will be different, as it will use your service URL, organization name, and project name.

5. Press **Enter**.
   The cloning process starts. Enter your Oracle Developer Cloud Service password, when prompted. Example:

```
[root@server devops]# git clone https://rnsangwan%40gmail.com@tspdevops-theskillpedia.developer.ocp.oraclecloud.com/tspdevops-theskillpedia
/tspdevops-theskillpedia_my-devops-project_27305/scm/tsp-employee-app.git
Cloning into 'tsp-employee-app'...
Password for 'https://rnsangwan@gmail.com@tspdevops-theskillpedia.developer.ocp.oraclecloud.com':
warning: remote HEAD refers to nonexistent ref, unable to checkout.

[root@server devops]# ls -al
total 4
drwxr-xr-x   3 root root   30 Jan 16 07:59 .
dr-xr-x---. 19 root root 4096 Jan 16 07:54 ..
drwxr-xr-x   3 root root   18 Jan 16 07:59 tsp-employee-app
[root@server devops]# cd tsp-employee-app/
[root@server tsp-employee-app]# ll
total 0
[root@server tsp-employee-app]# ls -al
total 0
drwxr-xr-x 3 root root  18 Jan 16 07:59 .
drwxr-xr-x 3 root root  30 Jan 16 07:59 ..
drwxr-xr-x 7 root root 138 Jan 16 08:01 .git
[root@server tsp-employee-app]#
```

   Ignore the warning as it indicates that the repository is empty. So, don't be alarmed.

   This is necessary because after the project Git repository is cloned, you are not automatically navigated to it.

6. Open the Explorer and navigate to the cloned repository directory. Notice the .git directory created in the cloned repository directory. Do not delete or rename the .git directory as it contains necessary Git repository files.

Sequence 5. Add the Sample Application Files to the Cloned Git Repository

Sample application. Click here to download the Employee sample application. The application is a Java SE web application that uses Servlets, JSP, Bootstrap and embedded Tomcat.

1. Download and extract a Sample App.

   # cd tsp-employee-app/

   # wget https://github.com/Sangwan70/devops/raw/main/employees-app.zip

```
[root@server tsp-employee-app]# wget https://github.com/Sangwan70/devops/raw/main/employees-app.zip
--2021-01-16 08:04:50--  https://github.com/Sangwan70/devops/raw/main/employees-app.zip
Resolving github.com (github.com)... 13.234.176.102
Connecting to github.com (github.com)|13.234.176.102|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.githubusercontent.com/Sangwan70/devops/main/employees-app.zip [following]
--2021-01-16 08:04:51--  https://raw.githubusercontent.com/Sangwan70/devops/main/employees-app.zip
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 199.232.252.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|199.232.252.133|:443... connected
HTTP request sent, awaiting response... 200 OK
Length: 150536 (147K) [application/zip]
Saving to: ‘employees-app.zip’

100%[=====================================================================================================

2021-01-16 08:04:52 (1.30 MB/s) - ‘employees-app.zip’ saved [150536/150536]

[root@server tsp-employee-app]#
```

2. Extract the sample application zip to the cloned repository directory and remove the zip file.

   # unzip employees-app.zip

   [root@server tsp-employee-app]# rm employees-app.zip

rm: remove regular file 'employees-app.zip'? y

3. Run the "***git add .***" command to add or stage all files of the current directory to the local Git repository index of the default master branch.

   # ***git add .***

   The above command adds all files and sub-directories of the current directory to the repository index.

4. Run the git status command to verify that all files have been added to the index.

   # git status

```
[root@server tsp-employee-app]# git add .
[root@server tsp-employee-app]# git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   manifest.json
#       new file:   pom.xml
#       new file:   src/assembly/distribution.xml
#       new file:   src/main/java/com/example/employees/Employee.java
#       new file:   src/main/java/com/example/employees/EmployeeList.java
#       new file:   src/main/java/com/example/employees/EmployeeService.java
#       new file:   src/main/java/com/example/employees/EmployeeServlet.java
#       new file:   src/main/java/com/example/employees/Main.java
#       new file:   src/main/webapp/META-INF/context.xml
#       new file:   src/main/webapp/WEB-INF/web.xml
#       new file:   src/main/webapp/css/bootstrap.min.css
#       new file:   src/main/webapp/fonts/glyphicons-halflings-regular.eot
#       new file:   src/main/webapp/fonts/glyphicons-halflings-regular.svg
#       new file:   src/main/webapp/fonts/glyphicons-halflings-regular.ttf
#       new file:   src/main/webapp/fonts/glyphicons-halflings-regular.woff
#       new file:   src/main/webapp/fonts/glyphicons-halflings-regular.woff2
```

Sequence 6. Commit and Push the Sample Application Files to the Project Git Repository

1. In Git Bash, run the git commit -m "Initial commit" command to commit the staged files to the local Git repository and add Initial commit as the commit message.

   # git commit -m "Initial commit"

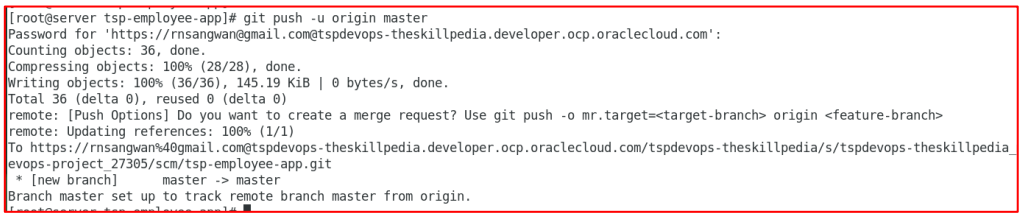
   [master (root-commit) db040ae] Initial commit

```
[root@server tsp-employee-app]# git commit -m "Initial commit"
[master (root-commit) b5a682d] Initial commit
 Committer: root <root@server.example.com>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 20 files changed, 955 insertions(+)
 create mode 100644 manifest.json
```

The above command commits all added files to the default master branch of the local Git repository.

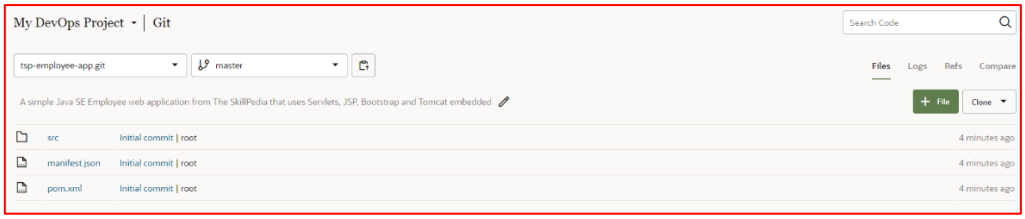
2. Run the git push -u origin master command to push the commits of the local Git repository master branch to the project Git repository. In the command, origin is the default name assigned to the remote project Git repository. Enter your Oracle Developer Cloud Service password, when prompted.

   # git push -u origin master



   After the command successfully runs, the sample application files are uploaded to the project Git repository on Oracle Developer Cloud Service.

3. In Oracle Developer Cloud Service project, click **Code** in the navigation bar. Browse and verify that all files are added to the repository.



4. In the navigation bar, click **Project.** A notification of the commit is added to the **Recent Activities** feed.
5. Congratulations. You have successfully completed the Lab.