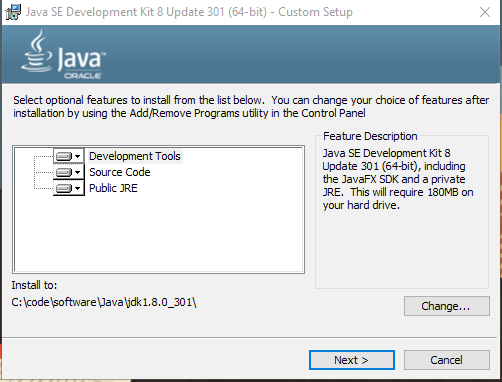
**Install and Configure Java**

1. Go to https://www.oracle.com/java/technologies/downloads/#java8-windows

2. Click on x64 exe. You will be redirected to a oracle login page. You can sign in or create an oracle account.

3. Execute the .exe thus downloaded

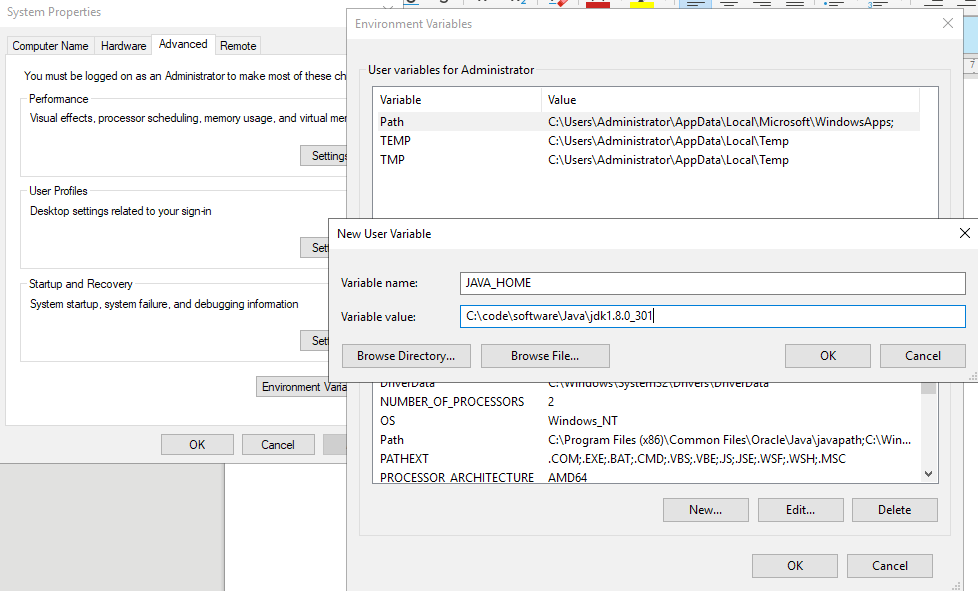
4. Install the Java to your location of choice. Also note where it is installed.



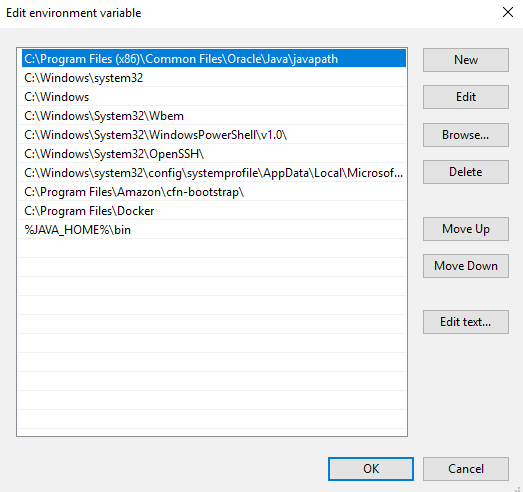
5. Choose the location where you want to install Java and make a note of it.

6. Install the JRE also.

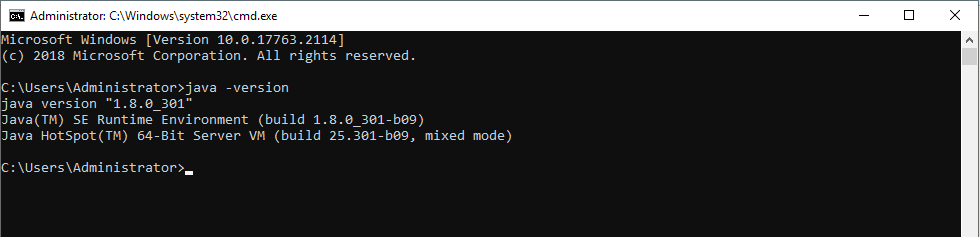
7. Setup an environment variable JAVA\_HOME



8. Add JAVA\_HOME\bin to path as follows



9. Click OK on all the dialog boxes. Open a command window and execute java -version. You should get output as follows.



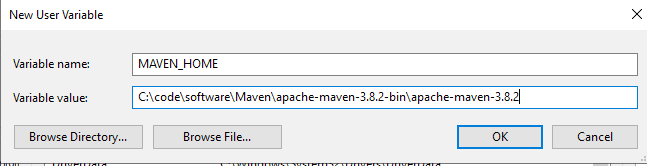
**Install and configure Maven**

1. https://maven.apache.org/download.cgi

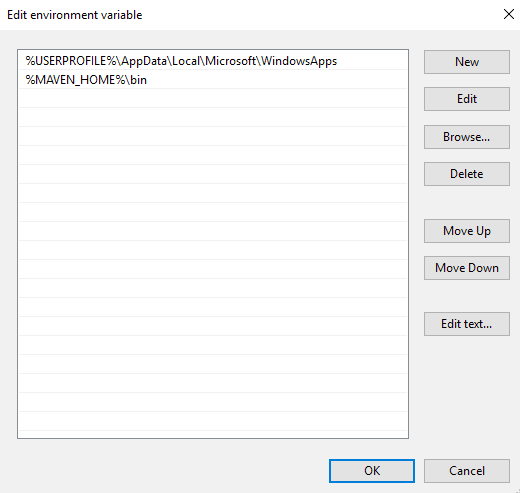
2. Click on [apache-maven-3.8.2-bin.zip](https://dlcdn.apache.org/maven/maven-3/3.8.2/binaries/apache-maven-3.8.2-bin.zip)

3. Extract the zip file thus downloaded to a location of your choice and note it.

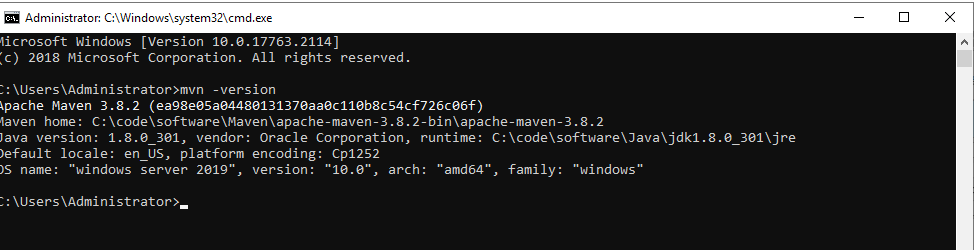
4. Add a new environment variable MAVEN\_HOME as follows



5. Add MAVEN\_HOME\bin to path as follows



6. Open a new command window and execute mvn -version. You should see an output as follows.



7. Download sesstings.xml from https://github.com/akshaytiwari0203/azure\_learning/tree/main/setup

8. Replace the settings.xml in MAVEN\_HOME\conf with the one that is downloaded in step 7.

9. Open the file and search for tag <localRepository>. Modify the value of this tag to appropriate location. Your maven dependencies will be downloaded at this location.

**Install Docker**

1. Open Windows Powershell and execute following commands

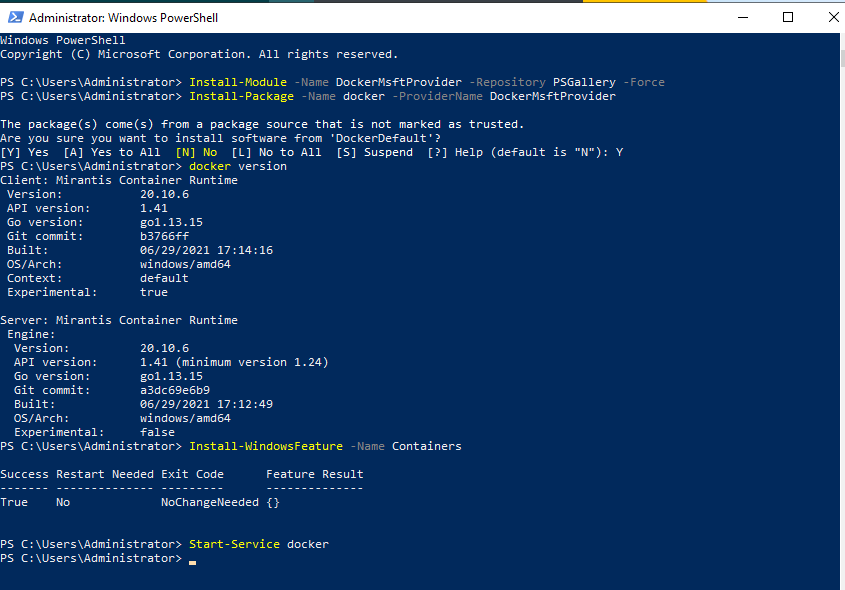
a. Install-Module -Name DockerMsftProvider -Repository PSGallery -Force

b. Install-Package -Name docker -ProviderName DockerMsftProvider

c. docker version

d. Install-WindowsFeature -Name Containers

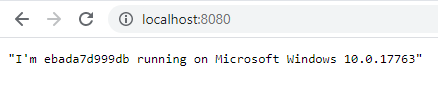
e. Start-Service docker



2. Execute following to run a sample docker application

docker container run -d -p 8080:80 sixeyed/whoami-dotnet:3.0

3. On browser go to <http://localhost:8080/>, You will see a webpage as follows:

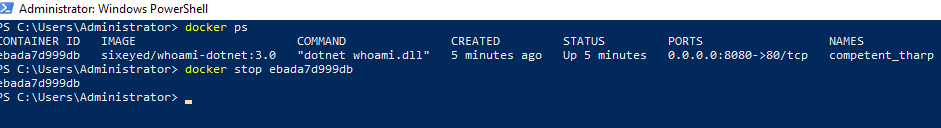


4. Run the following command and note the CONTAINERID

docker ps

5. Run the following command to stop dokcer container

docker stop <CONTAINERID>



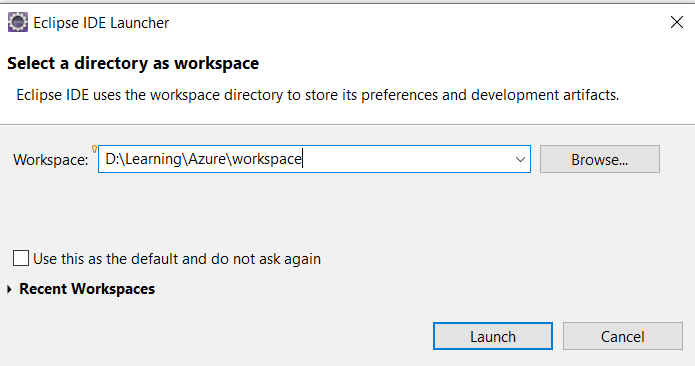
**Install and Setup Eclipse**

1. Go to <https://www.eclipse.org/downloads/packages/>

2. Download “[Eclipse IDE for Enterprise Java and Web Developers](https://www.eclipse.org/downloads/packages/release/2021-09/r/eclipse-ide-enterprise-java-and-web-developers)” as per your OS. A zip file will be downloaded.

3. Extract the zip file into a location of your choice.

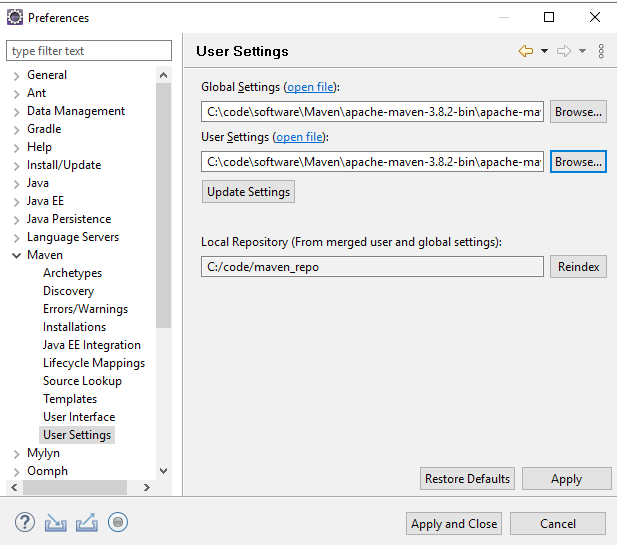
4. Go to the place where you extracted the Eclipse and double click eclipse.exe to launch eclipse. The following screen should appear.



5. For the workspace, browse to a convenient location and click launch

6. In eclipse, go to Windows>Preferences>Maven>User Settings

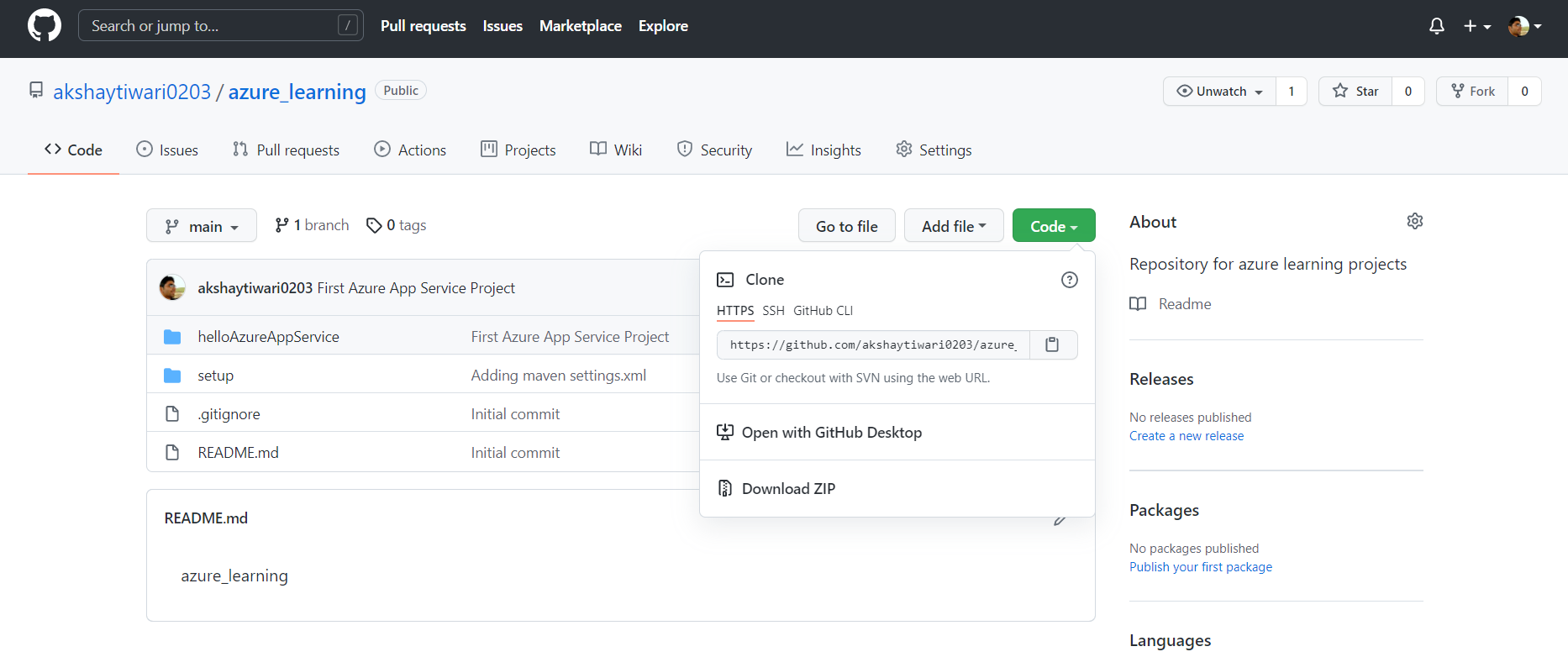
7. In Global Settings, browse to location of MAVEN\_HOME\conf\settings.xml and select it. Do the same for User Settings. Make sure Local Repository is of your choice.



9. Click Apply and Close

**Setting Up Project In Eclipse**

1. Download the project as a zip from <https://github.com/akshaytiwari0203/azure_learning> and extract it.

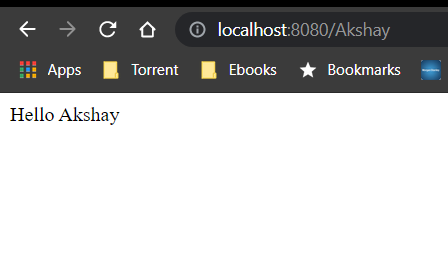


2. Copy the folder hello-azure-app-svc to a location where you would like to keep your code.

3. Import this folder as a maven project into eclipse.

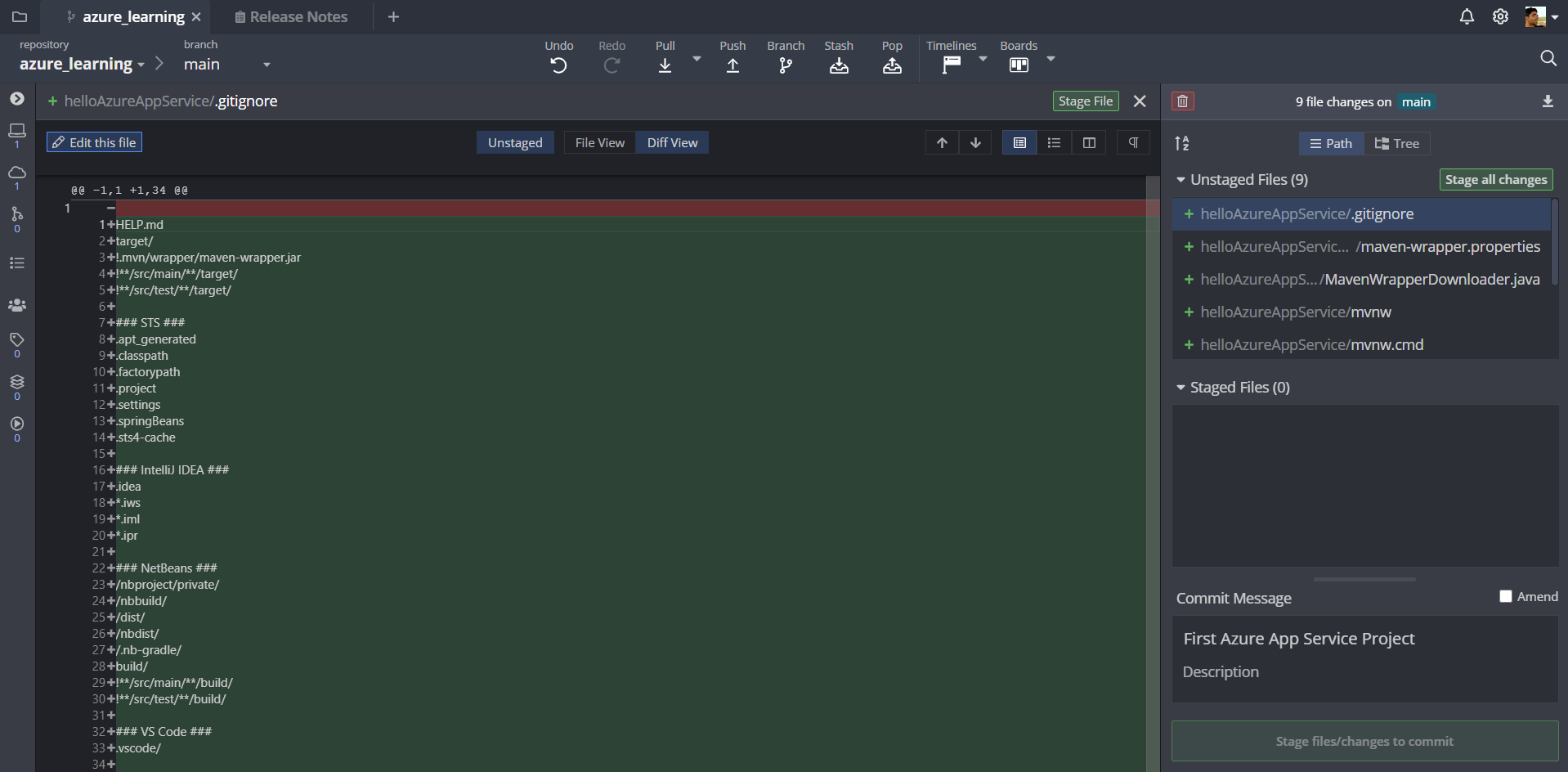
4. Once project is built, run the project as a java application.

5. Go to the browser and type url http://localhost:8080/Akshay



Push the project to GitHub

1. Go to your Git Kraken UI, you should see a screen as follows



2. Click on Stage All changes.

3. Enter a commit message and commit.

4. Push the changes to remote

Create a Git Hub Account

1. Go to <https://github.com/>

2. Click on signup

3. Complete the signup process

Install Git Kraken

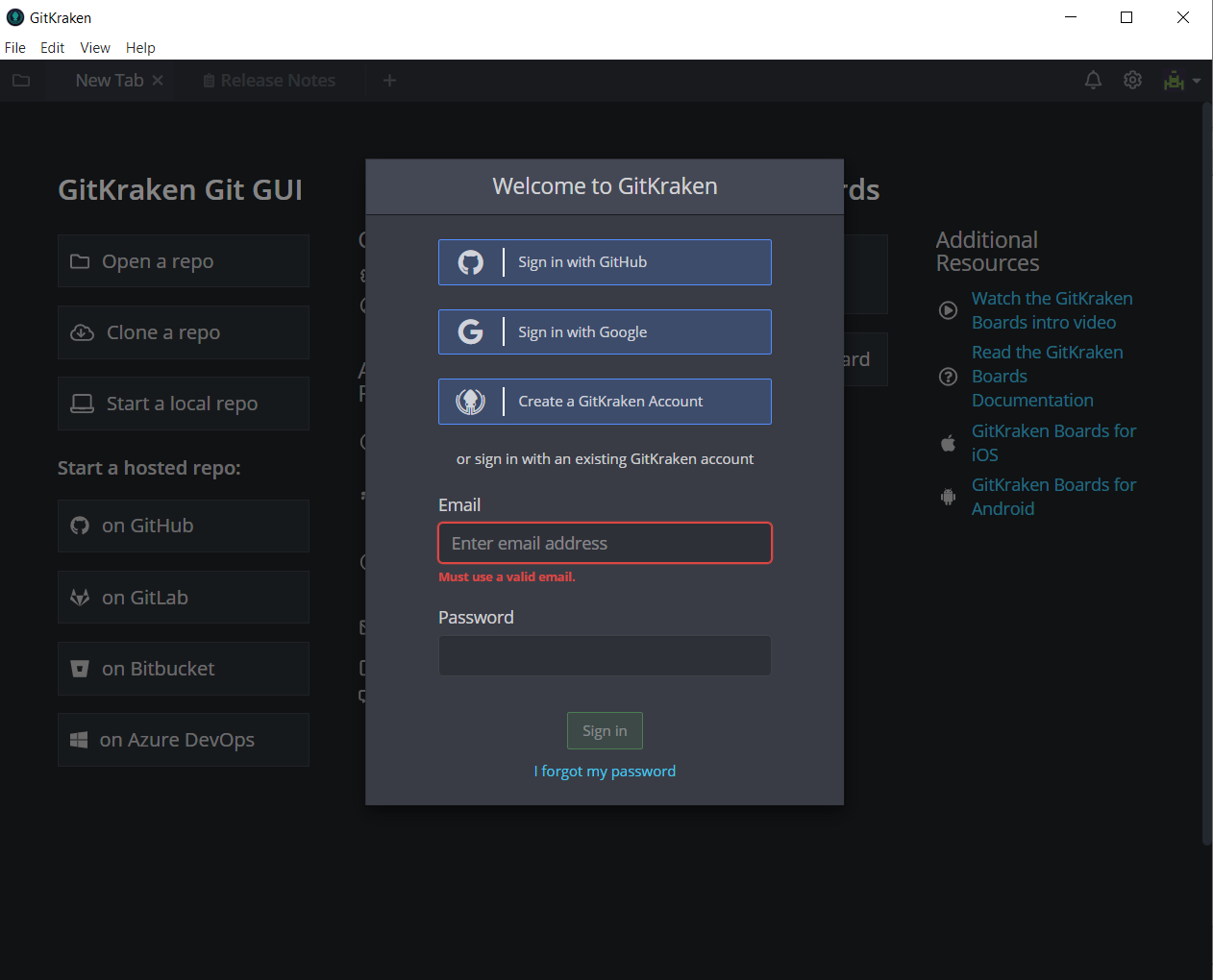
1. Go to <https://www.gitkraken.com/download>

2. Select the release as per your OS and download it.

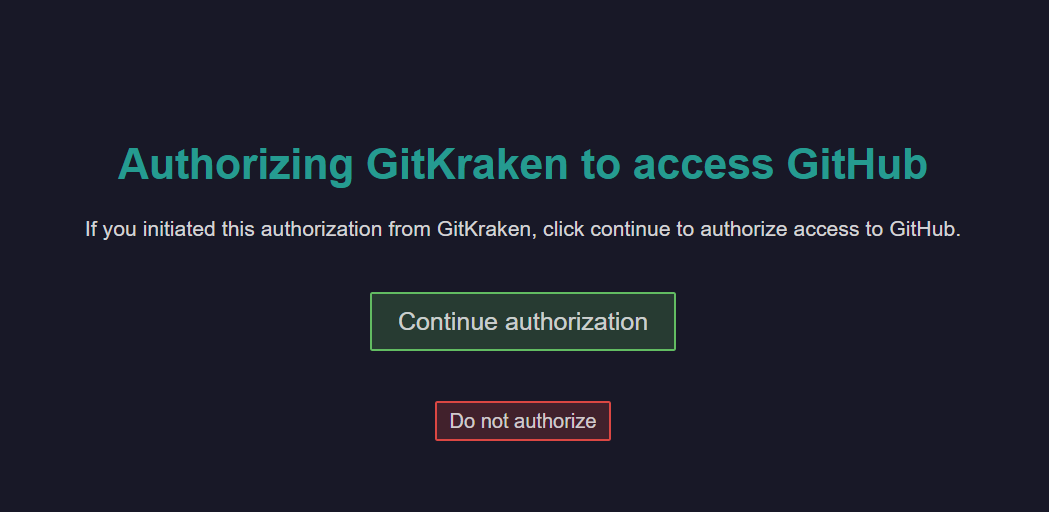
3. Copy the .exe file downloaded to a convenient location.

4. Execute the .exe file to start the Git Kraken UI. The following screen will appear:

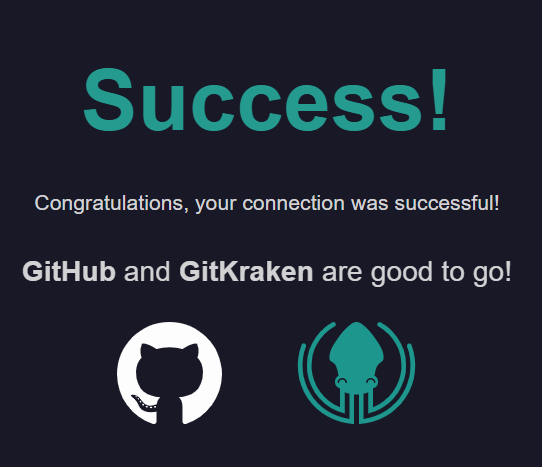
5. Click on Sign In with GitHub. A browser window as follows will appear.



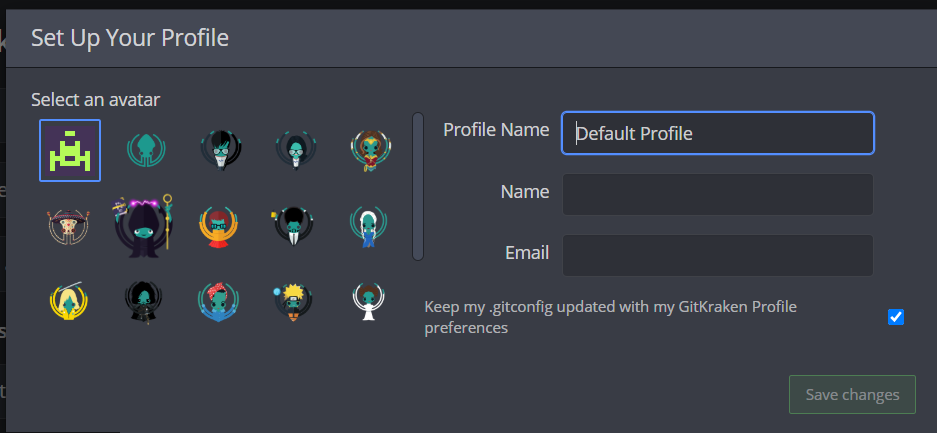
6. Click on Continue authorization.



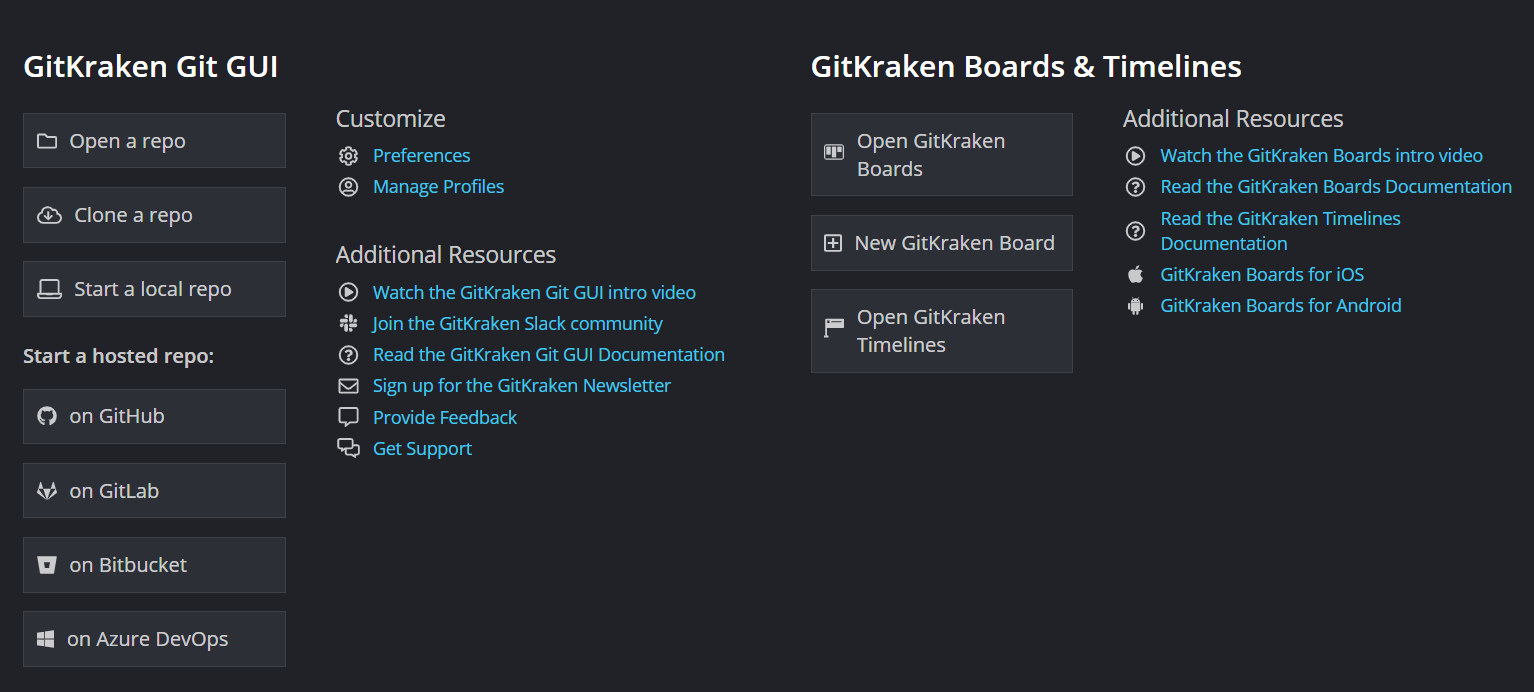
7. Enter username and password on next screen and a success message should appear.



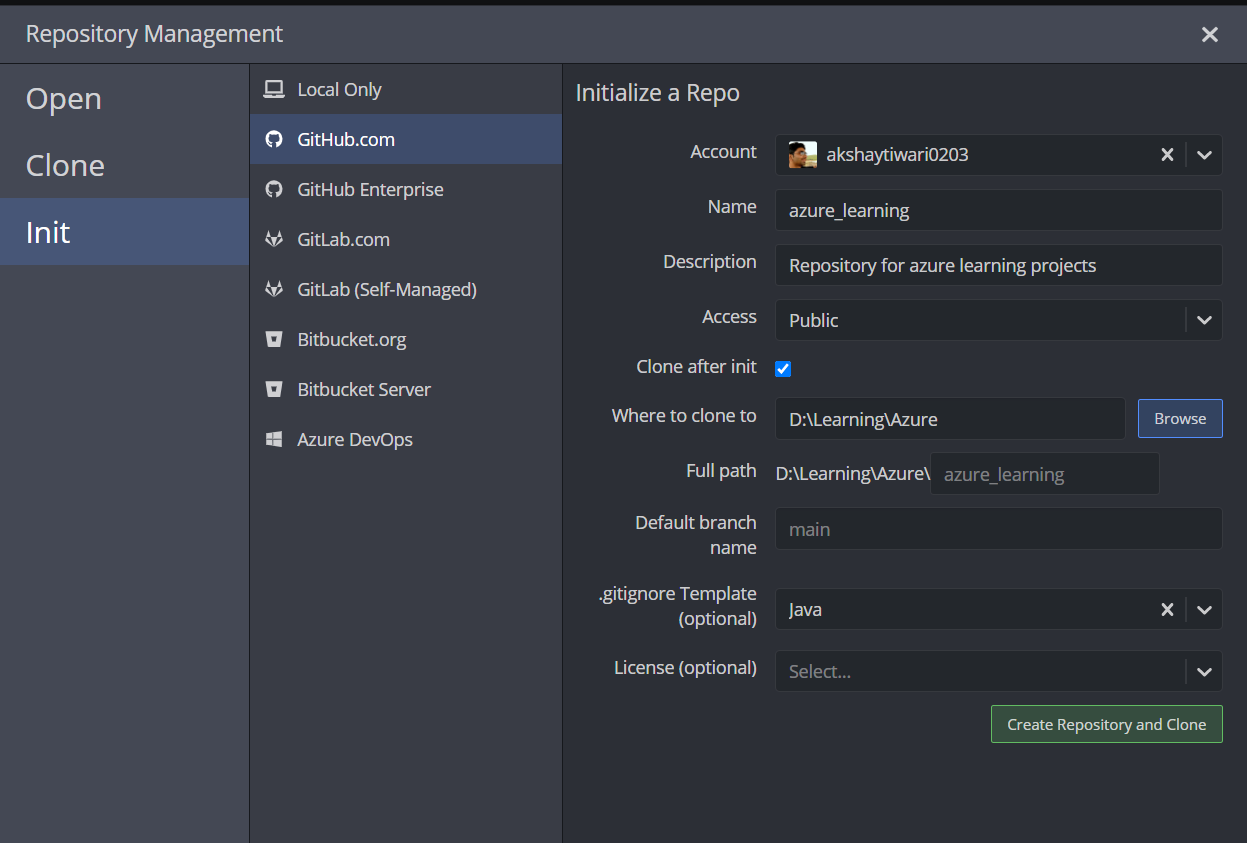
8. Go back to git kraken and setup your profile



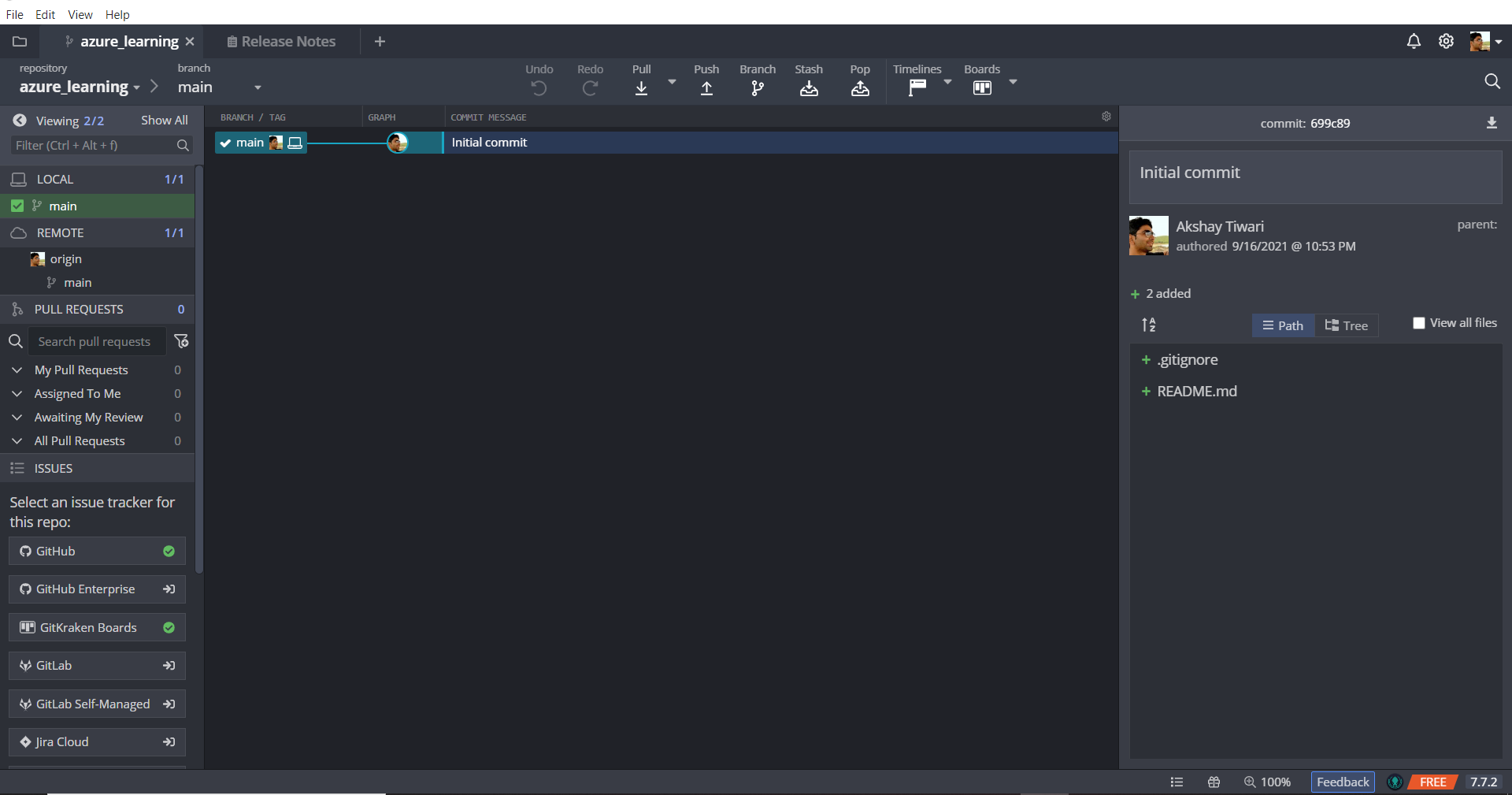
9. Start a new repo on git hub by clicking appropriate button on next screen.



10. On the next screen fill up the values as follows. Please note your account will be different. You can also select other values as per your choice.



11. Your Repository will be created and you should land on a page as follows:



12. Click on Pull button to synch remote with local repo

13. Push the .gitigonre and README file by putting appropriate commit message and clicking Push button (You may have to force push first time)