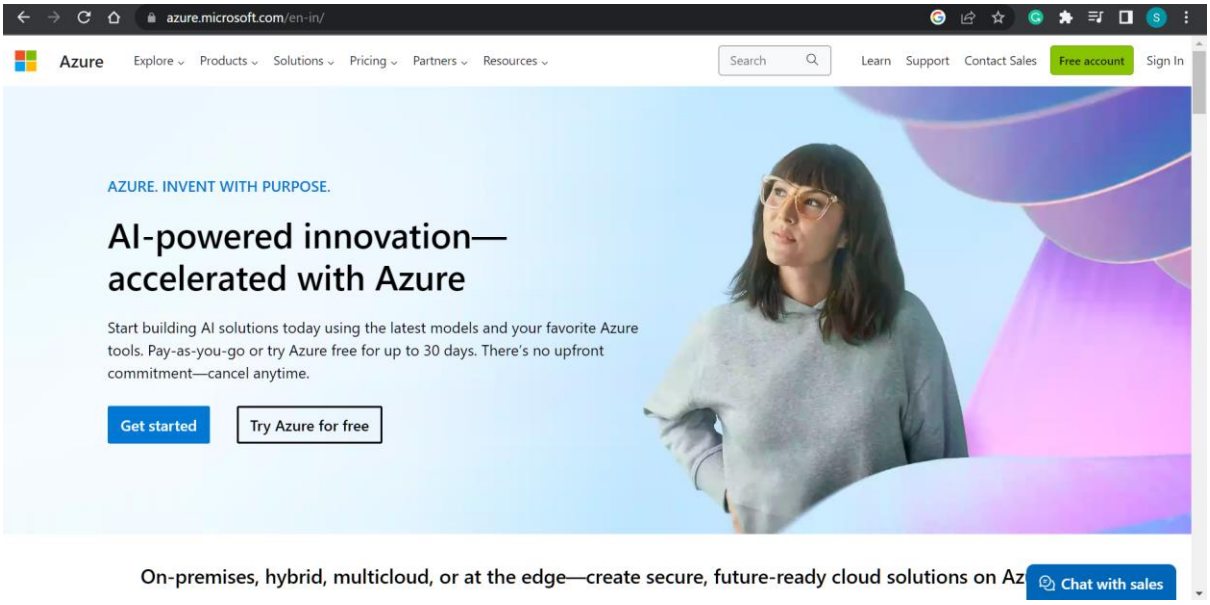
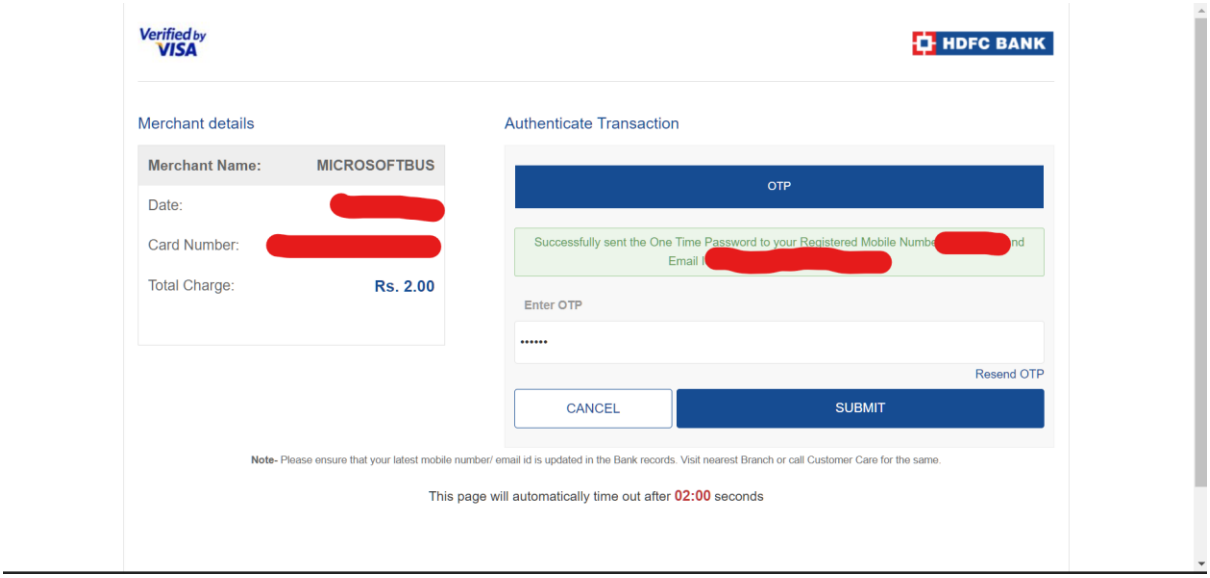
AZURE DEPLOYMENT

1. Go to https://azure.microsoft.com/en-in/ and create a free account:

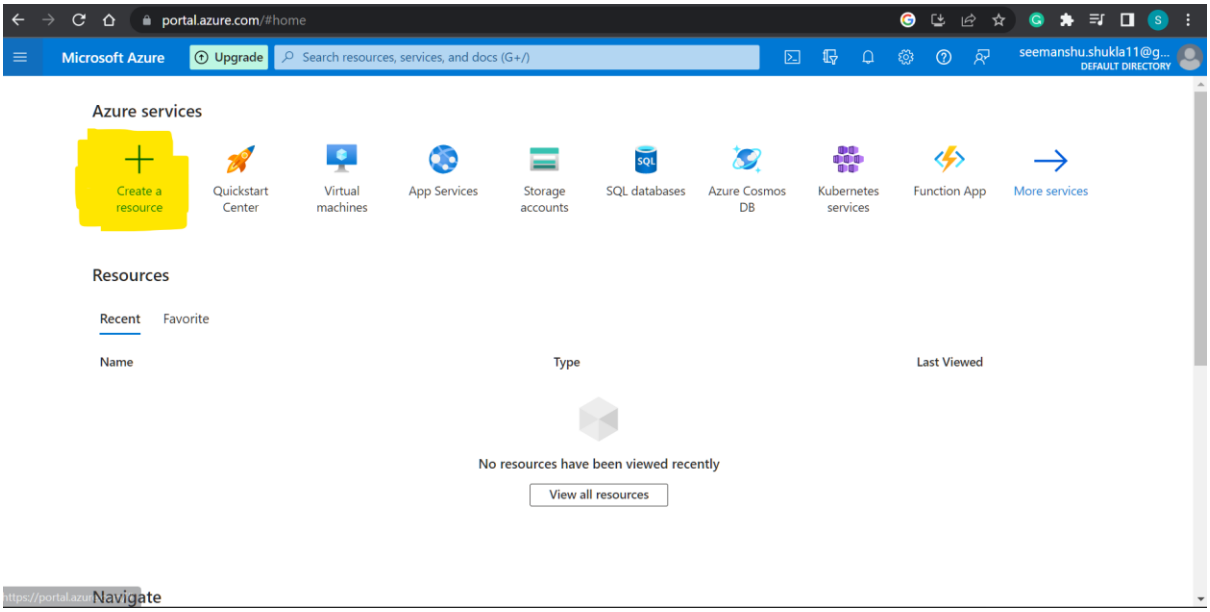


2. While creating account one needs to add a credit or debit card as an Identity verification where we are charged 2 INR:

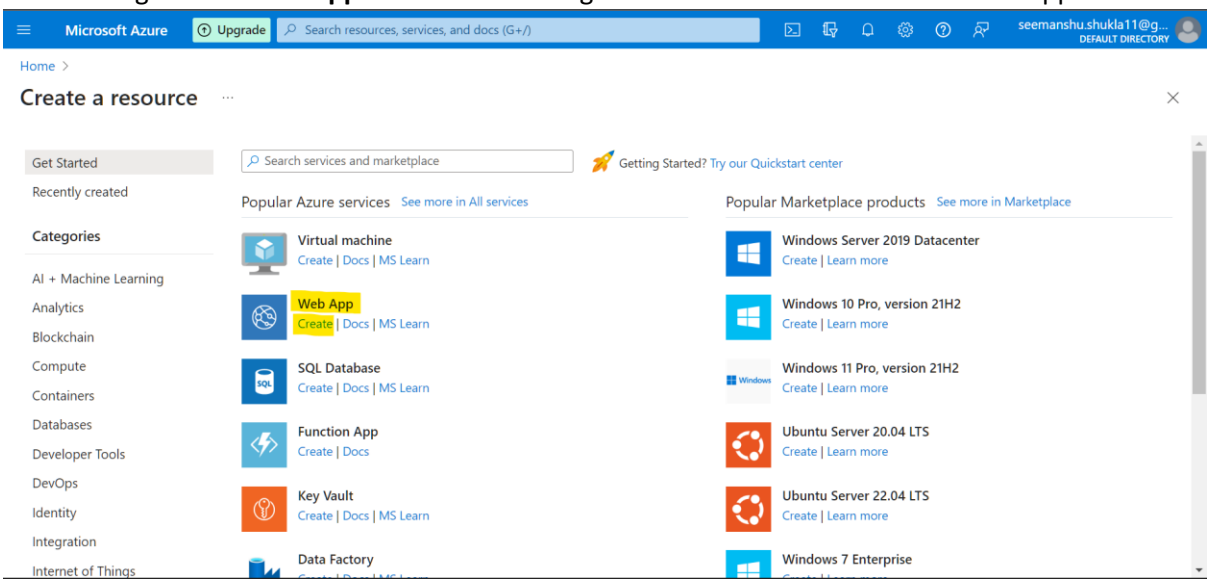


3. 300 $ free credits will be available for first 30 days post that only after getting confirmation from our end, Microsoft will enable **Pay as per use** feature for the services that are chargeable.

4. Select **Create a resource**:

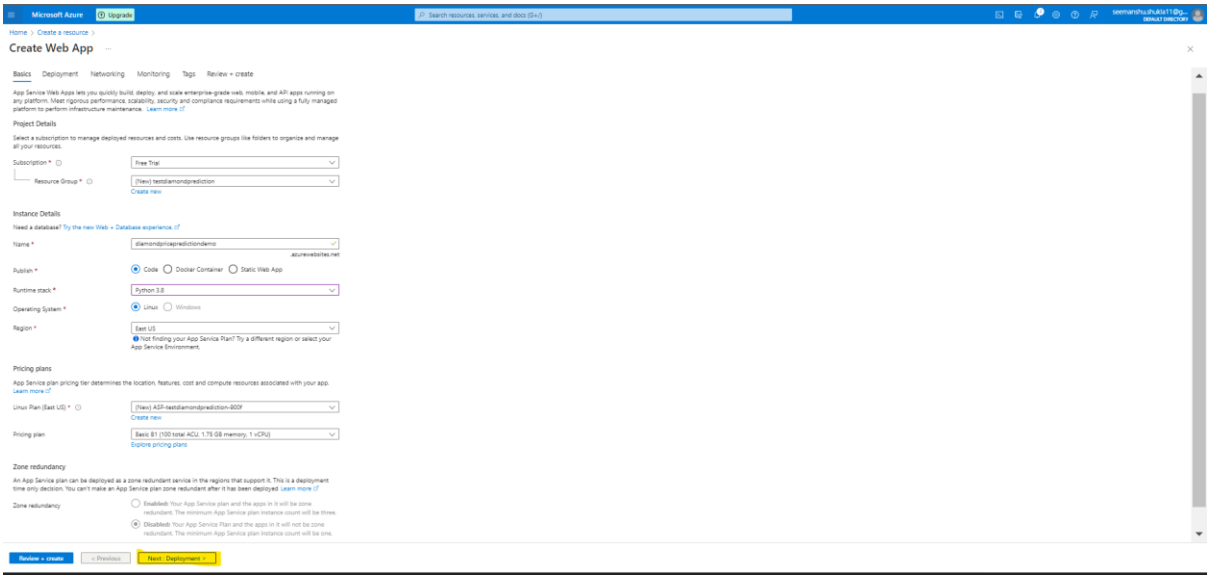


5. We be using Azure's **Web App** service for creating a resource. Click **Create** under Web App:



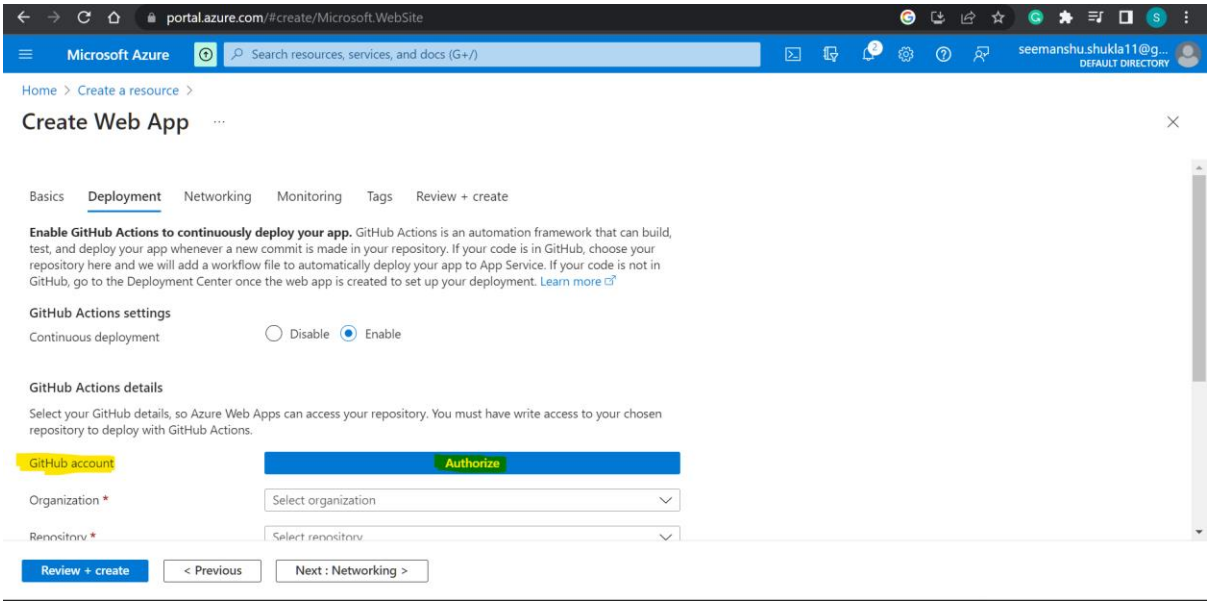
6.

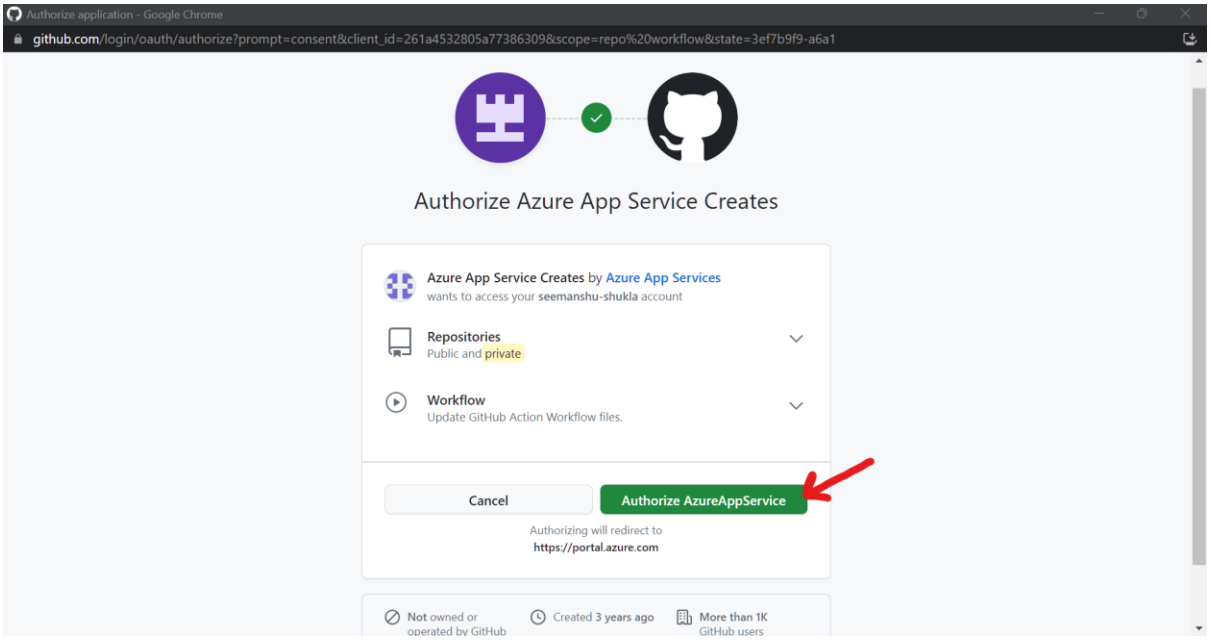
- **Subscription** defines the entity through which charges will be made. In our case it will be free trial.
- **Resource Group** is a container that holds related resources for an Azure solution.
- **Name** specifies the web app name.
- **Publish** defines the way we are deploying our application. It can be as a code or docker container or static web app.
- **Runtime** stack defines the stack or language that would be needed for running our application. In our case it's python 3.8



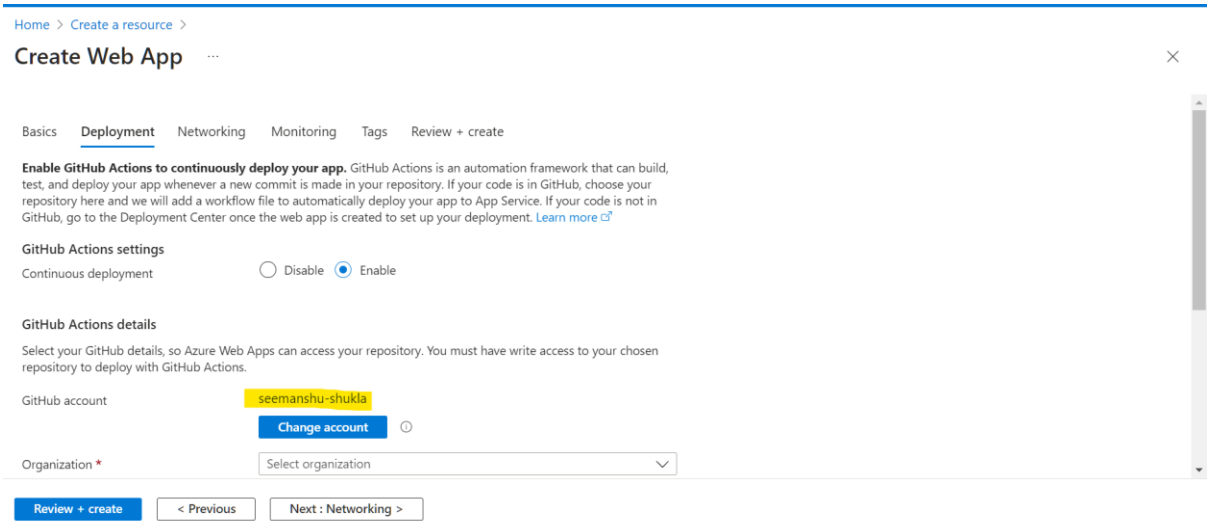
After defining the **Basic** configuration of deployment instance click **Next:Deployment >** button.

7. Defining **Depoyment** details:
   - **Enable** the **GitHub Actions settings**. This is done to enable Continuous deployment using which we are creating a pipeline to continuously move our code from GitHub to Azure cloud.
   - To configure or integrate the GitHub account use **GitHub account** option.

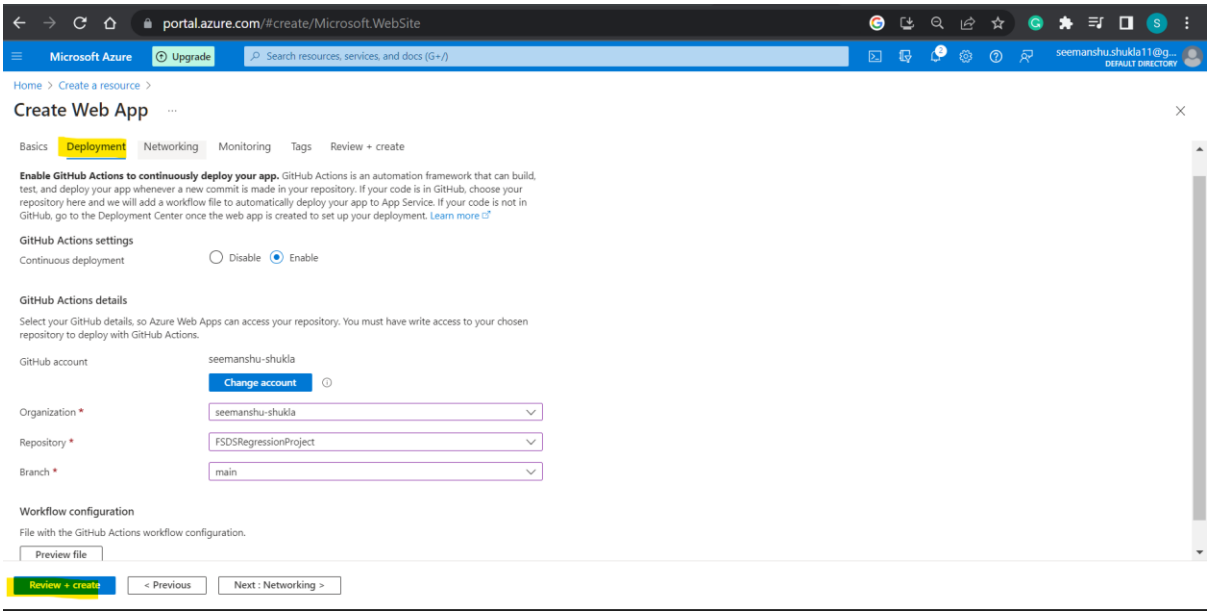




- Once GitHub is integrated with Azure GitHub user name will start reflecting:

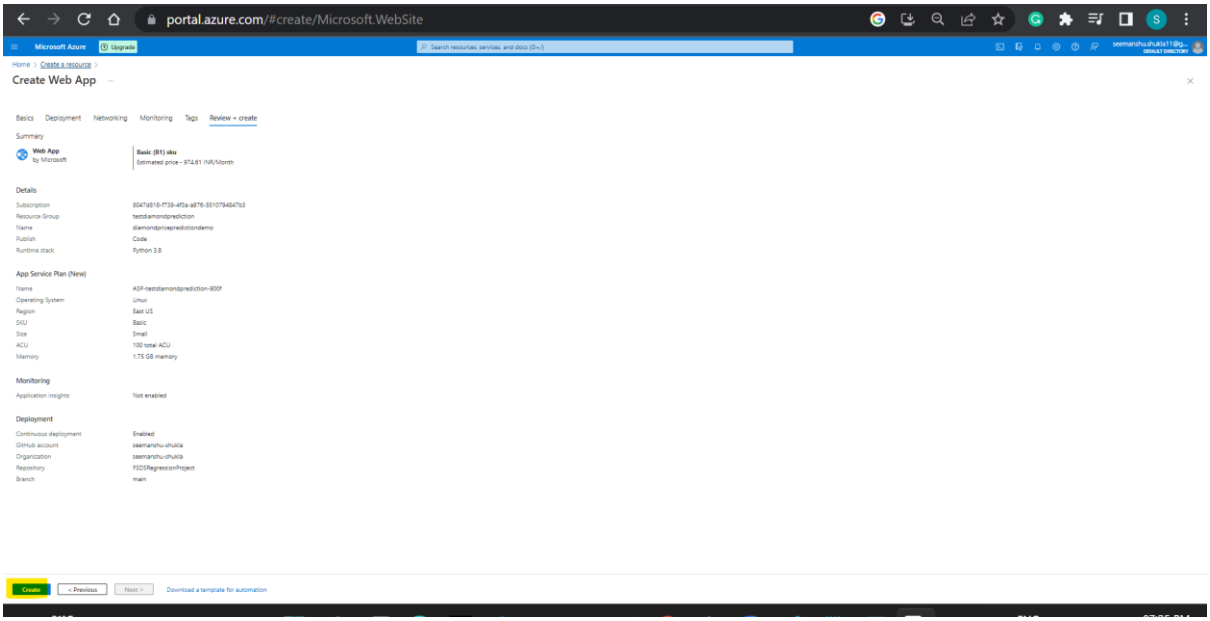


- **Organization** can be self-GitHub username**, Repository** will be the name of GitHub repository where application codebase is available, **Branch** represents the branch of target GitHub repository.

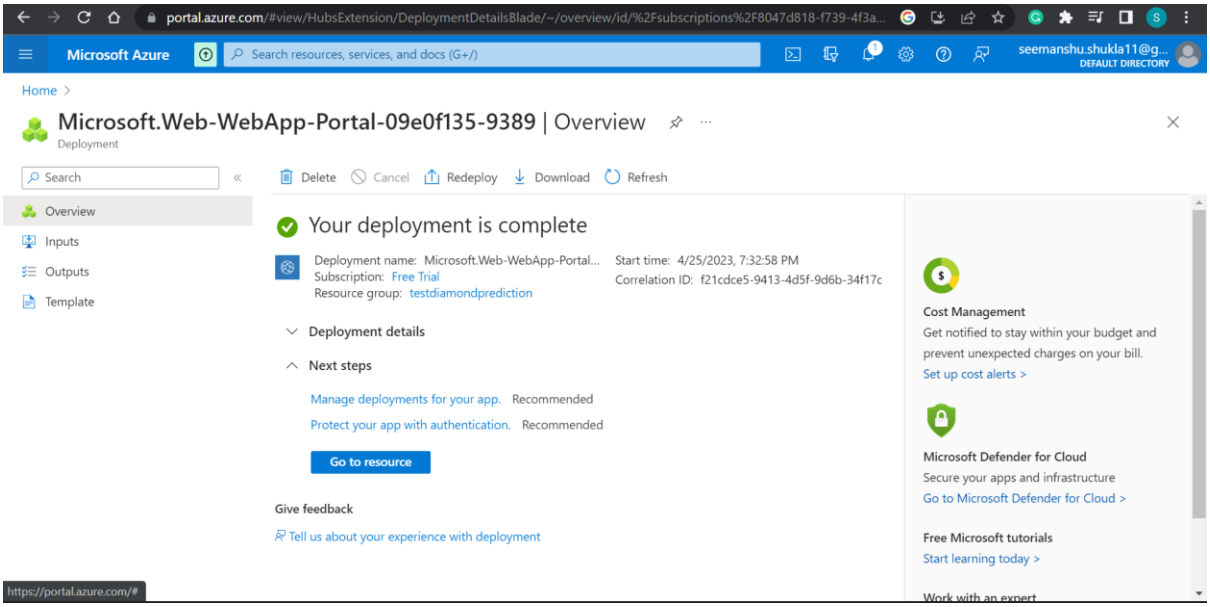


After completing the **Deployment** phase click **Review + create.**

8. Click on **Create** button after validating all the feed information :

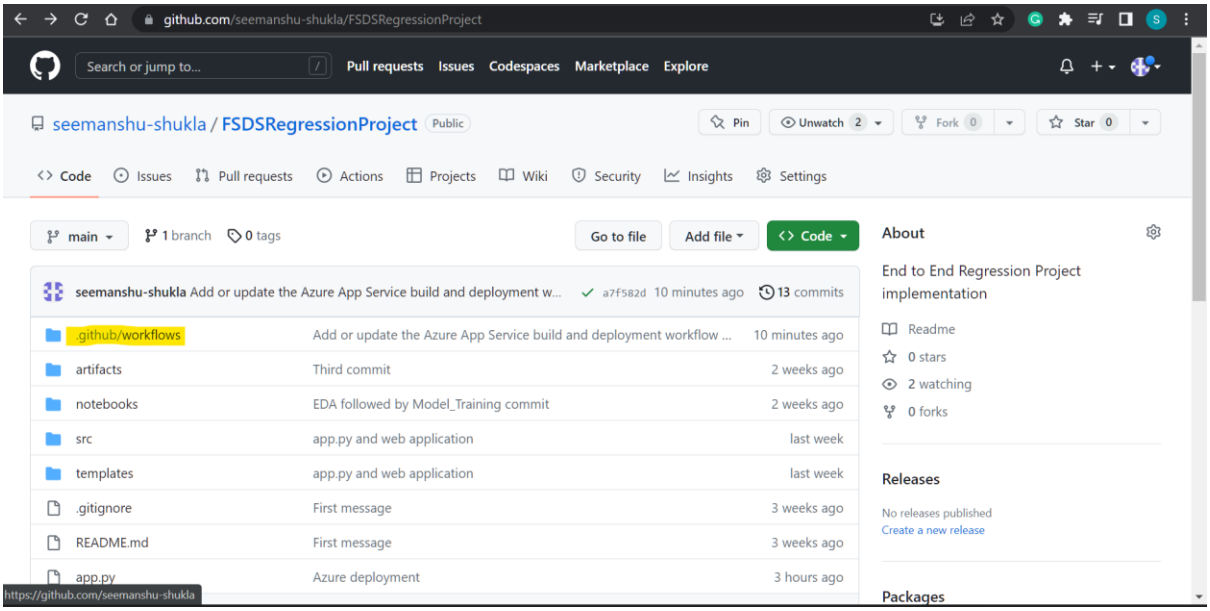


9. Wait for the deployment to get finished and then click **Go to resource** option:

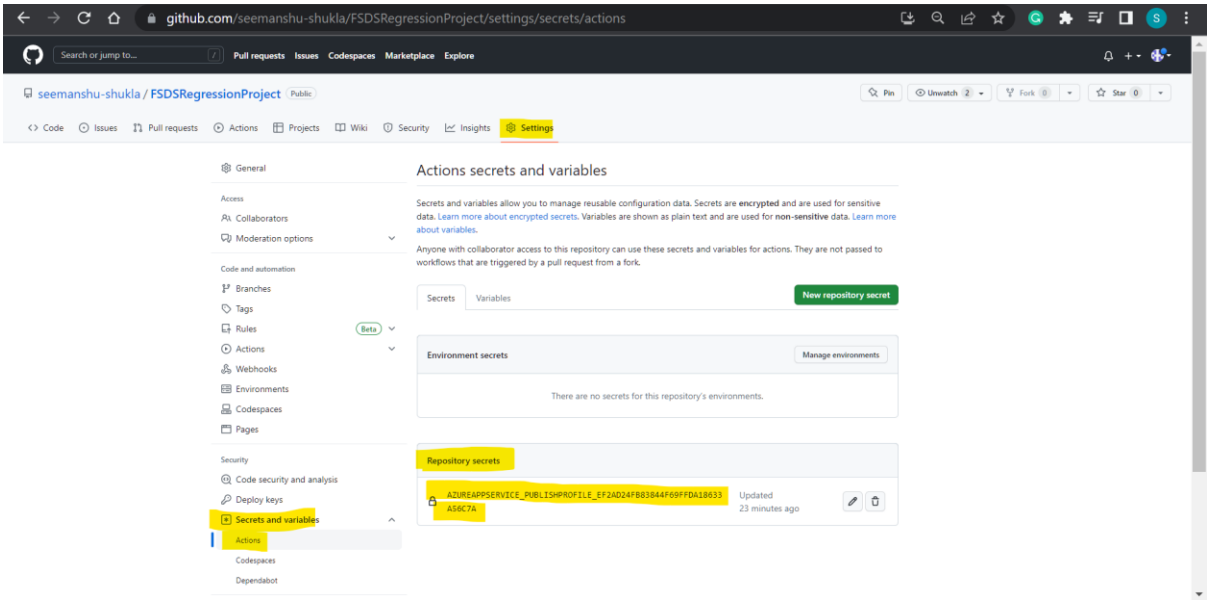


7. Changes in Github:

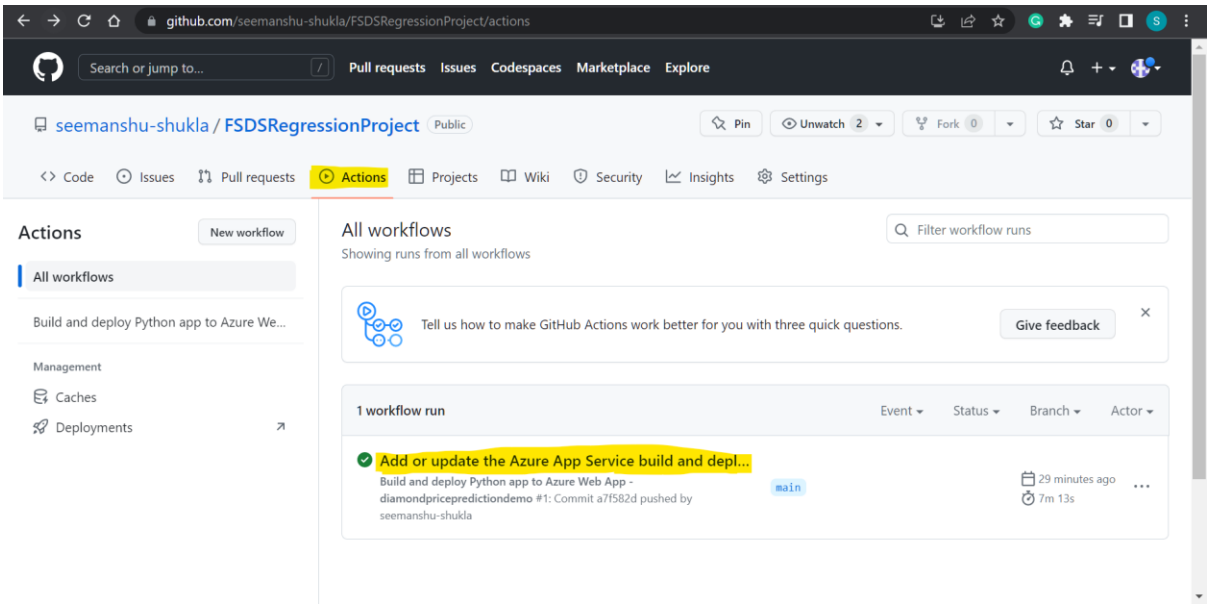
- Meanwhile, if we reload our Github repository a folder **.github/workflows** will be created. This will be containing .yml file which is basically defining the CD (Continuous deployment) pipeline for our deployed solution.

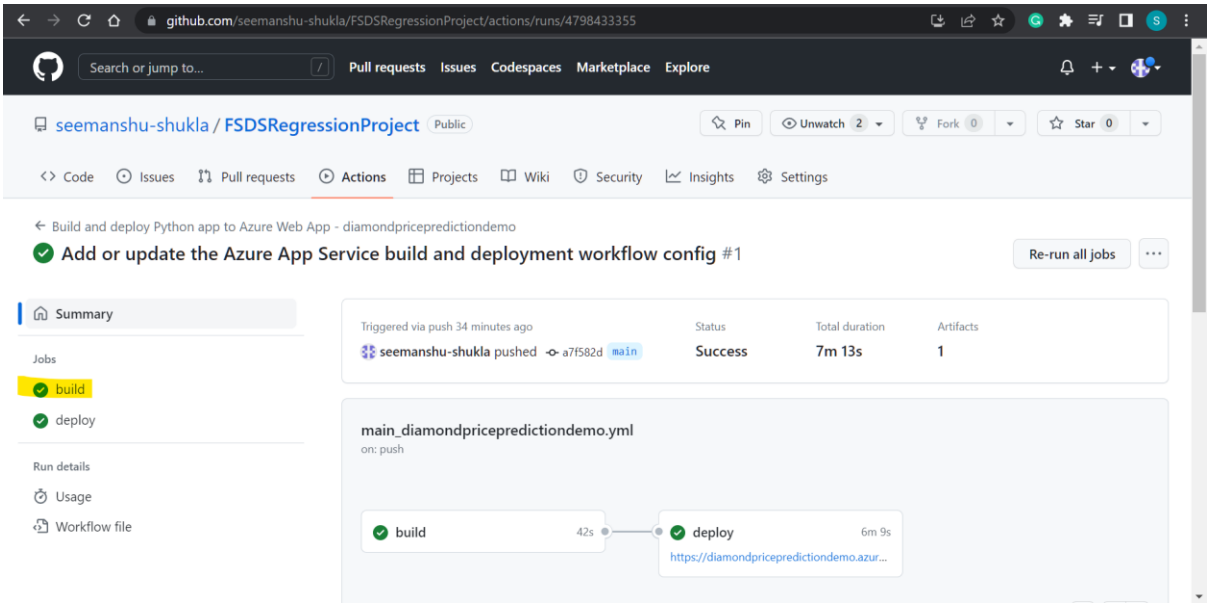


- Also, notice that Repository secrets is also created. Repository secrete is responsible for the creation of .yml file.

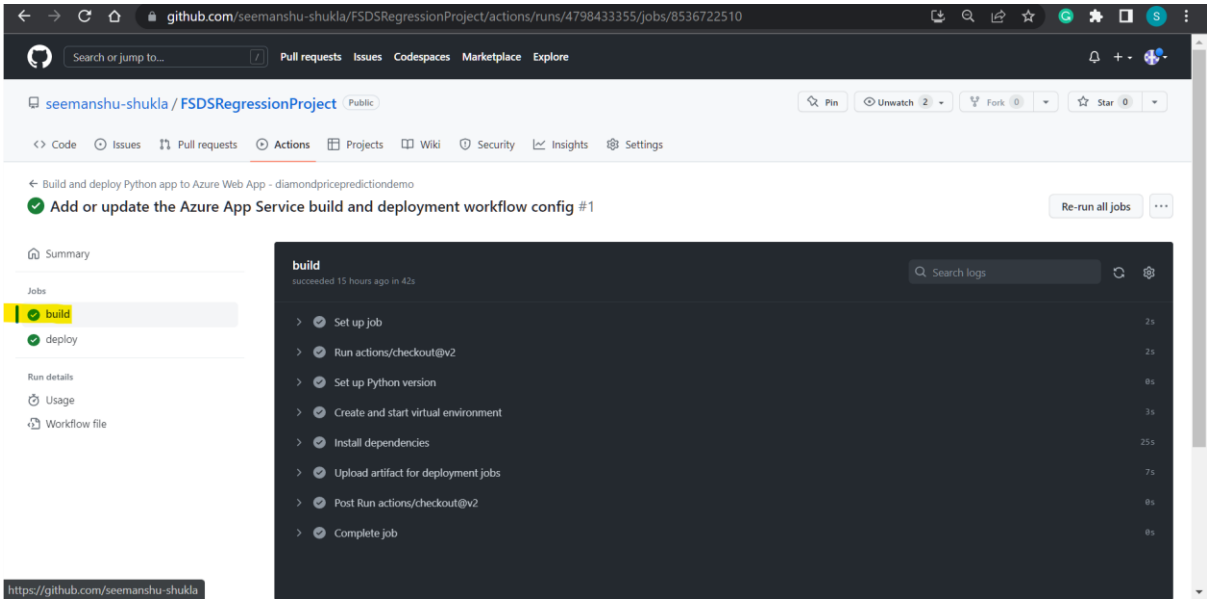


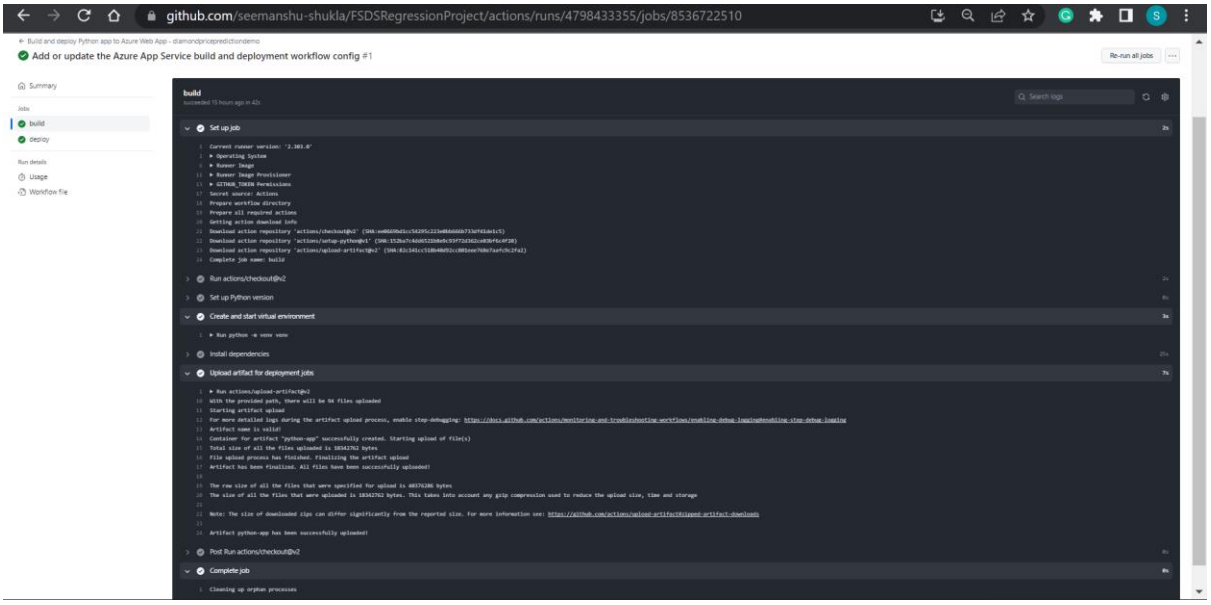
- Please note that in the current scenario GitHub actions is the deployment provider. That's why **Actions** is now reflecting with Build and Deploy actions:



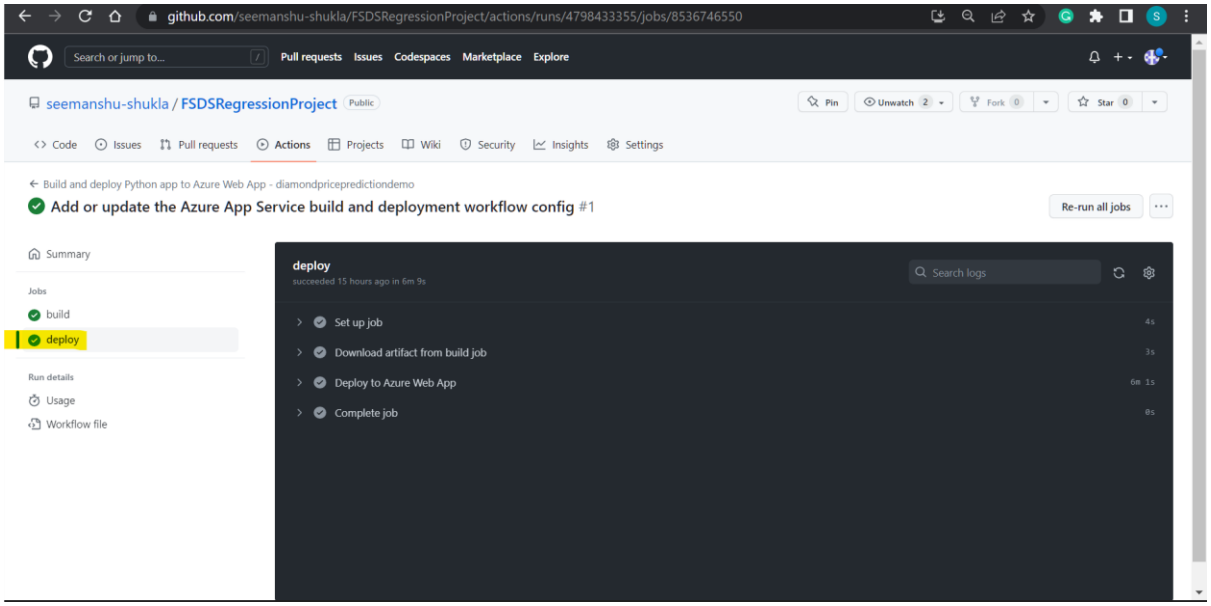


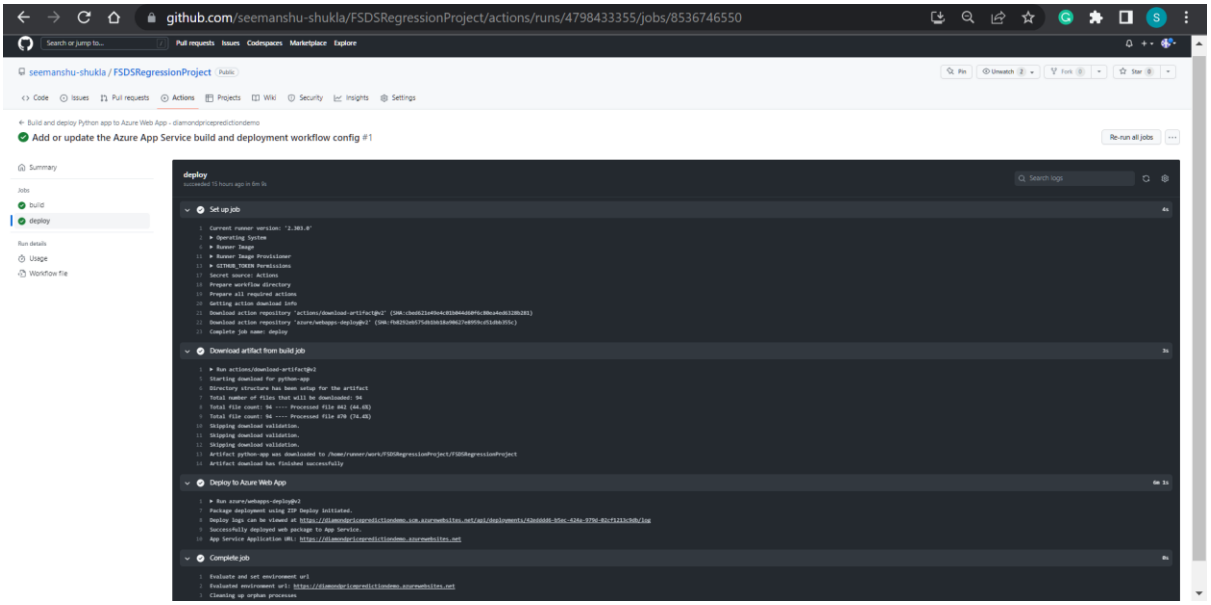
- We can visualize the Build configuration inside Build GitHub Actions. Please note that here we are visualizing the build configurations as defined under .yml file:



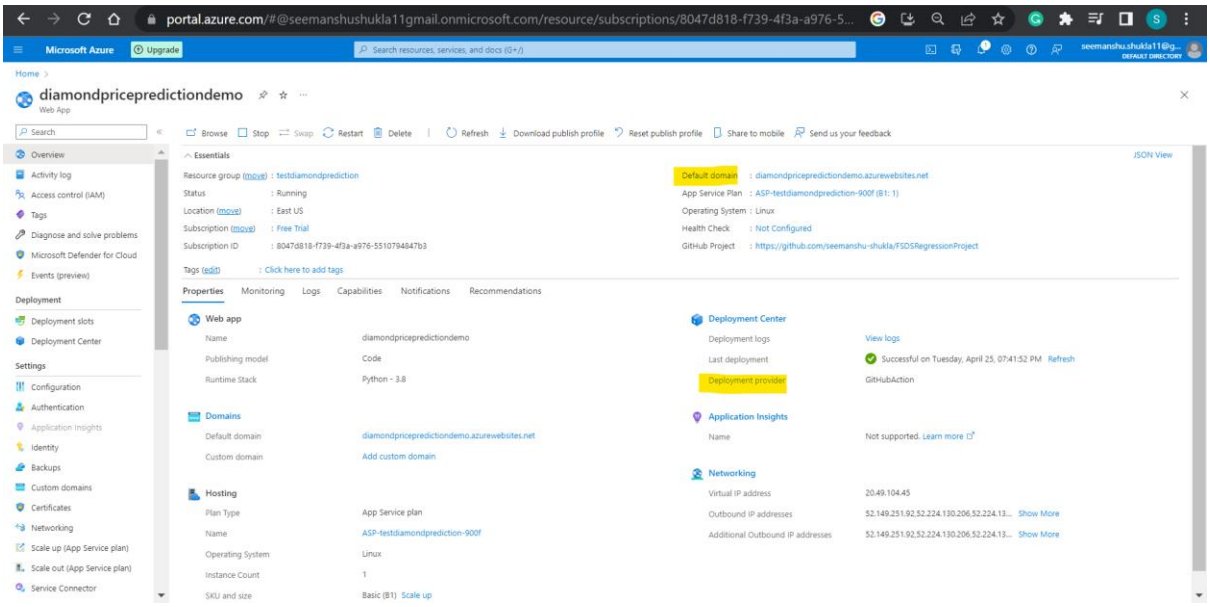


- In the same way we can also, visualize the Deploy configuration inside Deploy GitHub Actions. Please note that here we are visualizing the Deploy configurations that are defined under .yml file:



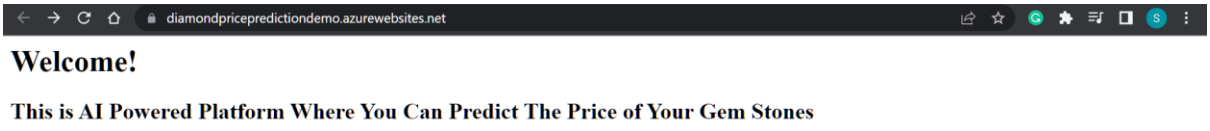


10. In Azure under **Overview** (after clicking Go to resource as mentioned in the earlier steps) we can get the complete summary of our deployment. Default domain represents the URL using which one can access our deployed application. Also, notice that the deployment provider is GitHub Actions:

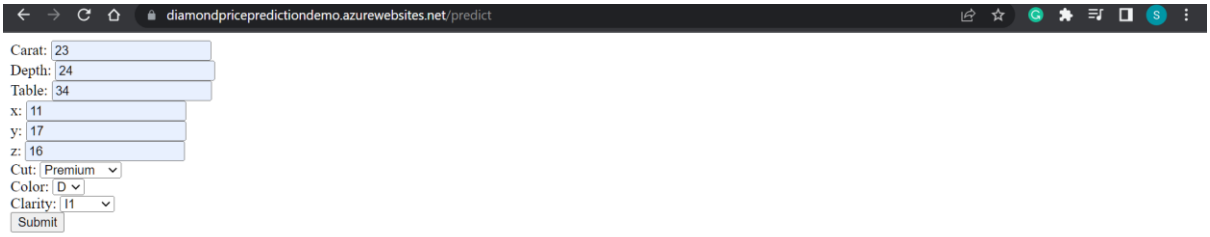


In above snip notice that Azure has provided with Stop, Restart etc option which can be helpful to troubleshoot in case of any issues by stopping or restarting the services deployed inside the server.

11. Accessing the deployed application:
    - Copy the URL corresponding to Default domain (as explained above): https://diamondpricepredictiondemo.azurewebsites.net/

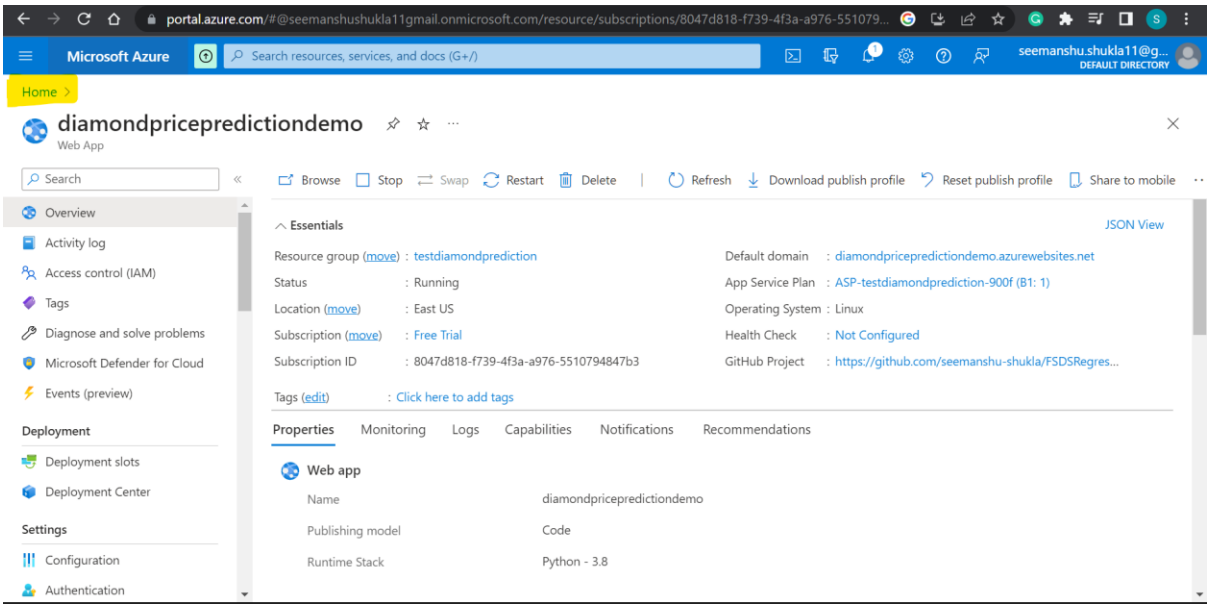


- Routing to /predict web page(as defined while building our Flask application) for entering Input features based on which prediction will be made: https://diamondpricepredictiondemo.azurewebsites.net/predict

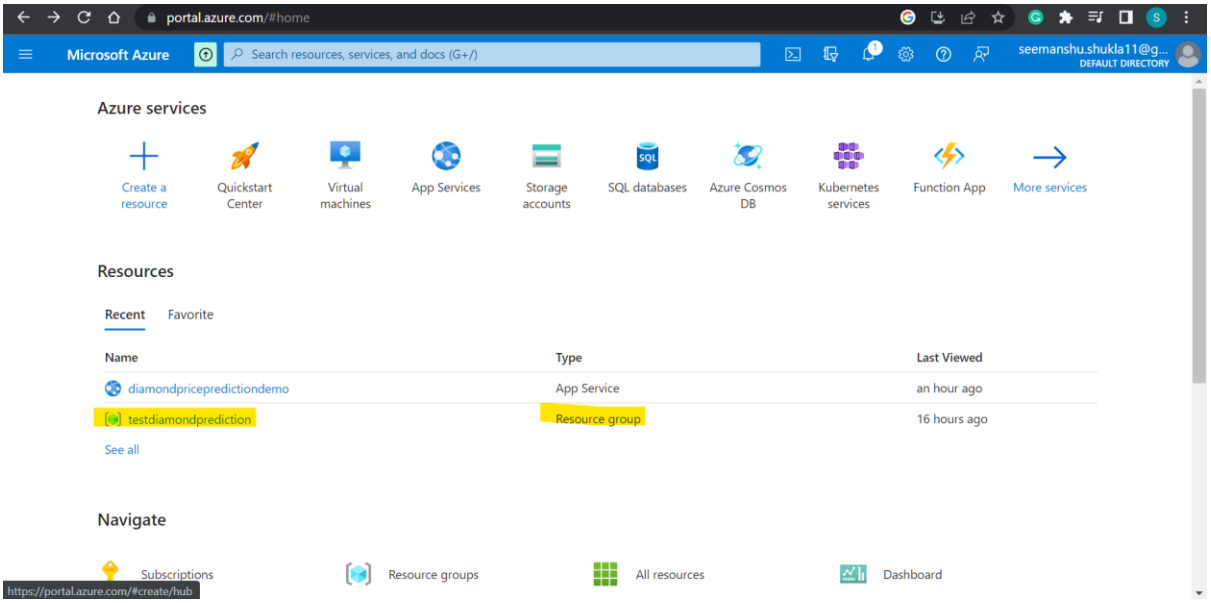


- On clicking Submit we will get redirected to results webpage where user can see predicted price of their Gem stone:

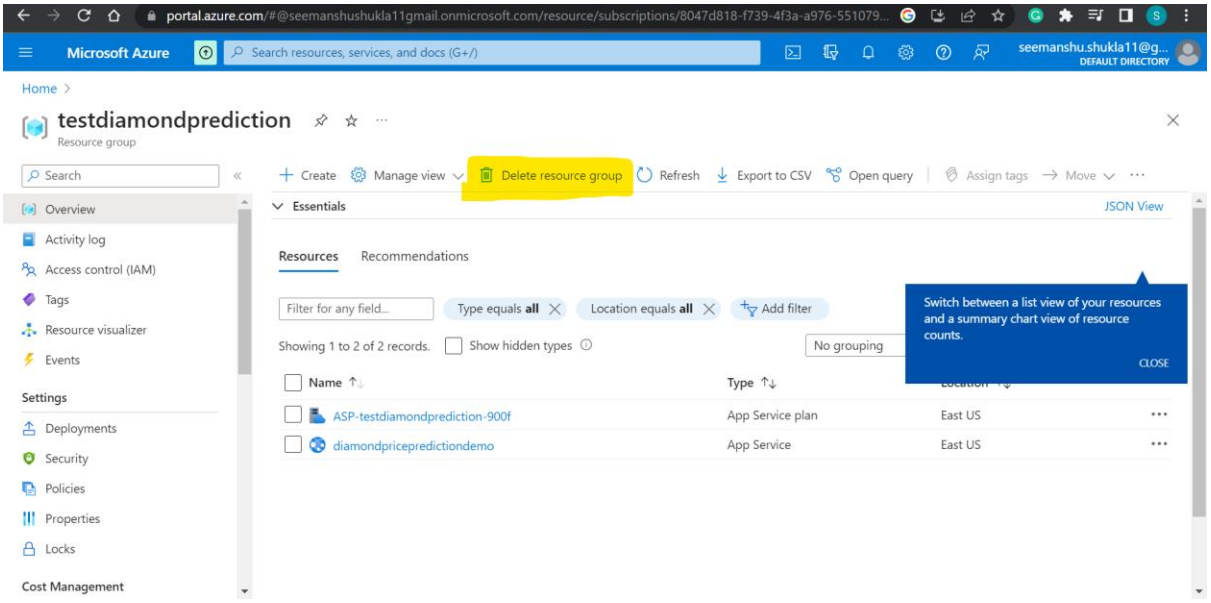
12. Deleting the deployment:
    - Click the Home icon as depicted in the snip below:



    - Select the Resource group that was defined while defining Basic details of deployment:



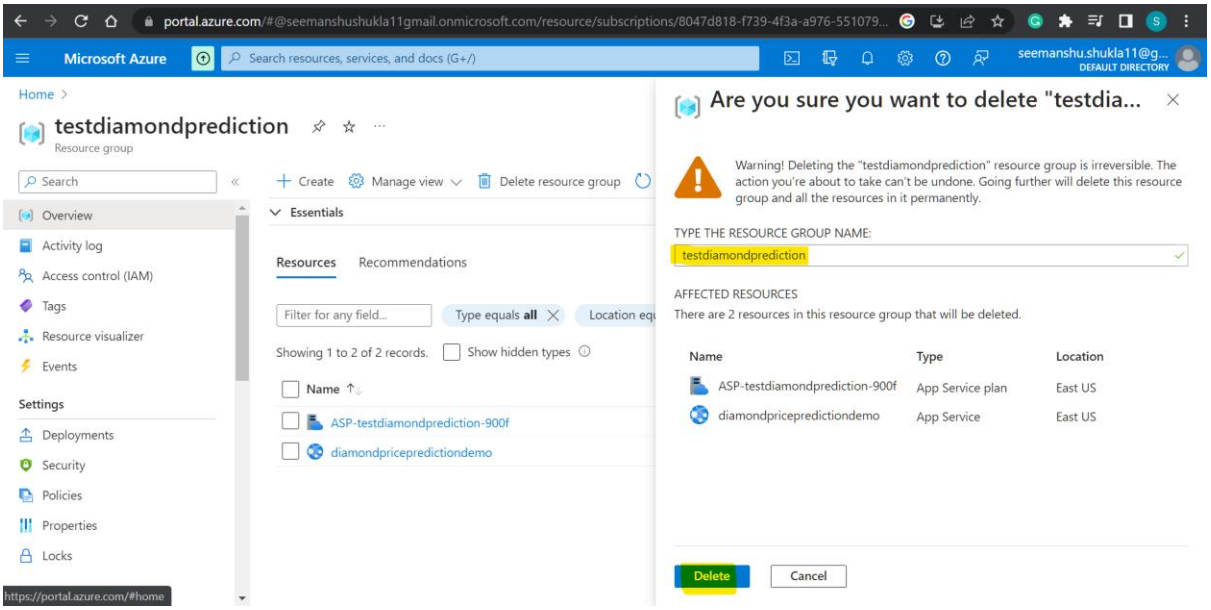
- Select **Delete resource group** to delete the entire resources of our deployment:



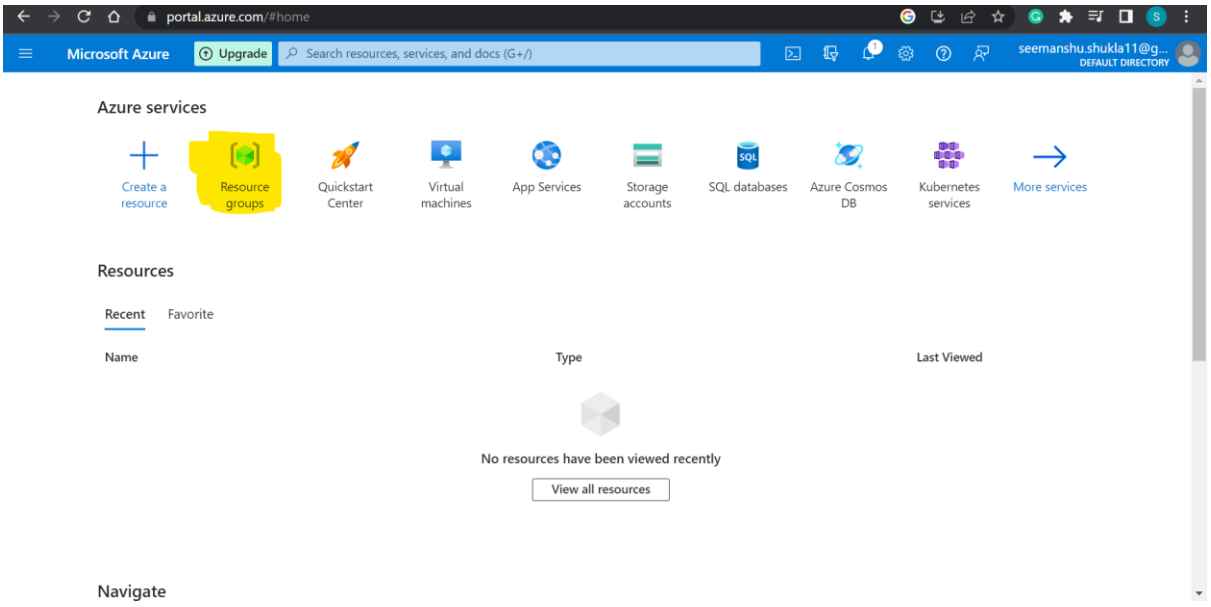
Observe above under Resources sub section we can see that currently there are 2 resources under this Resource group:

1. ASP-testdiamondprediction-900f which is of type **App Service plan**.
2. diamondpricepredictiondemo which is of type **App service**.

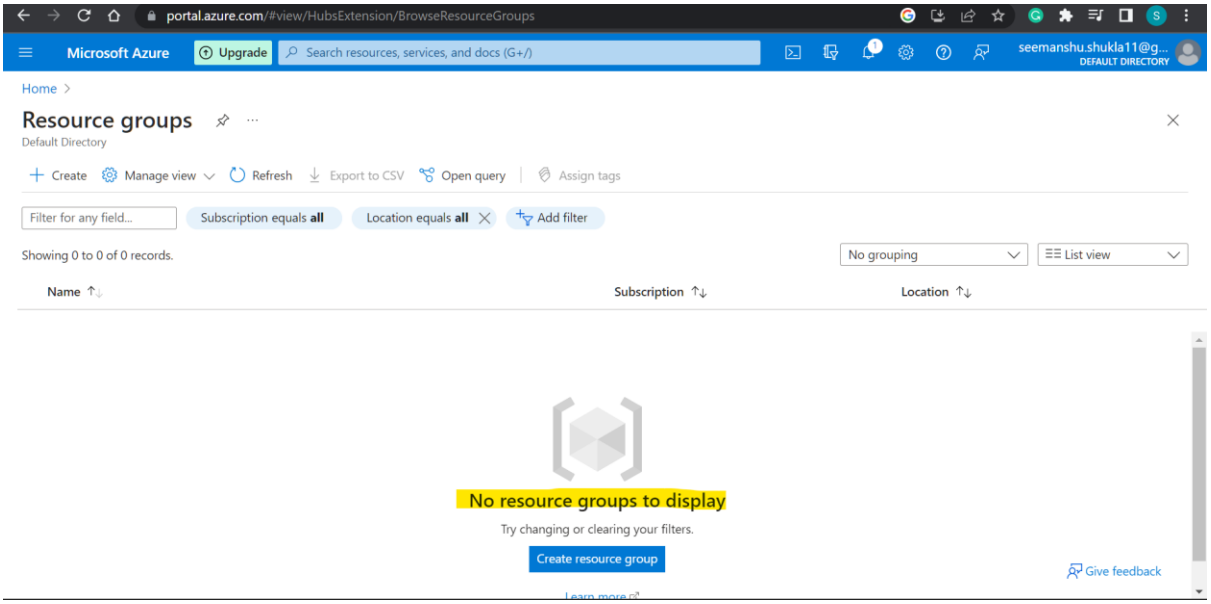
- After this we will get prompted with a window where we need to enter the resource group name to confirm the deletion:



- To verify whether the deployment is deleted successfully or not go to **Resource groups** section:



- If we don't see our Resource group, then it indicates that it is deleted successfully. In case someone has deployed their work just for learning purpose then they should ideally delete it later to ensure that Microsoft Azure do not charge anything extra.


portal.azure.com/#view/HubsExtension/BrowseResourceGroups
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
seemanshu.shukla11@g...
DEFAULT DIRECTORY
Home
Resource groups
Default Directory
Create
Manage view
Refresh
Export to CSV
Open query
Assign tags
Filter for any field...
Subscription equals all
Location equals all
Add filter
Showing 0 to 0 of 0 records.
No grouping
List view
Name
Subscription
Location
No resource groups to display
Try changing or clearing your filters.
Create resource group
Give feedback

******THANK YOU*******