Phase – 1:

Creating and deploying your own WordPress Site on Azure Cloud Environment.

Resource group snapshot ( Name Should be as instructed during lab).

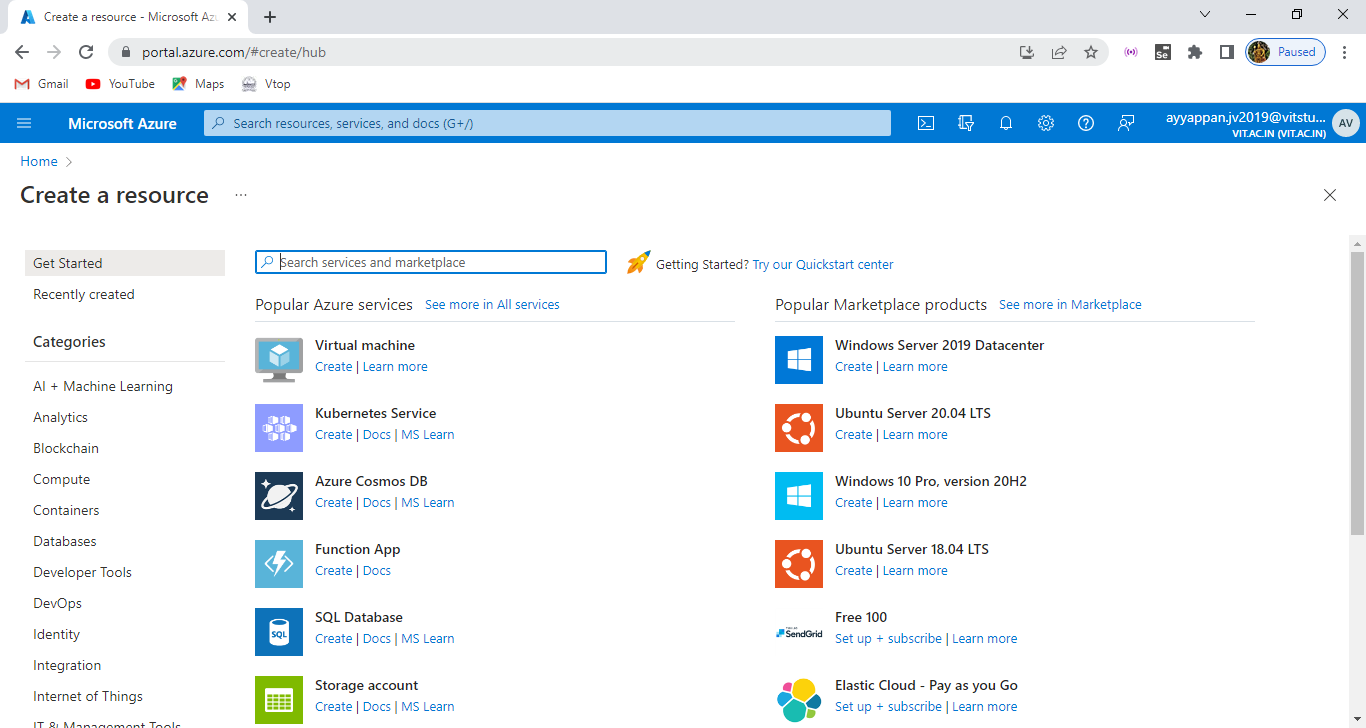
Instance  name should be as instructed during Lab.

Snap shot- Overview of resources.

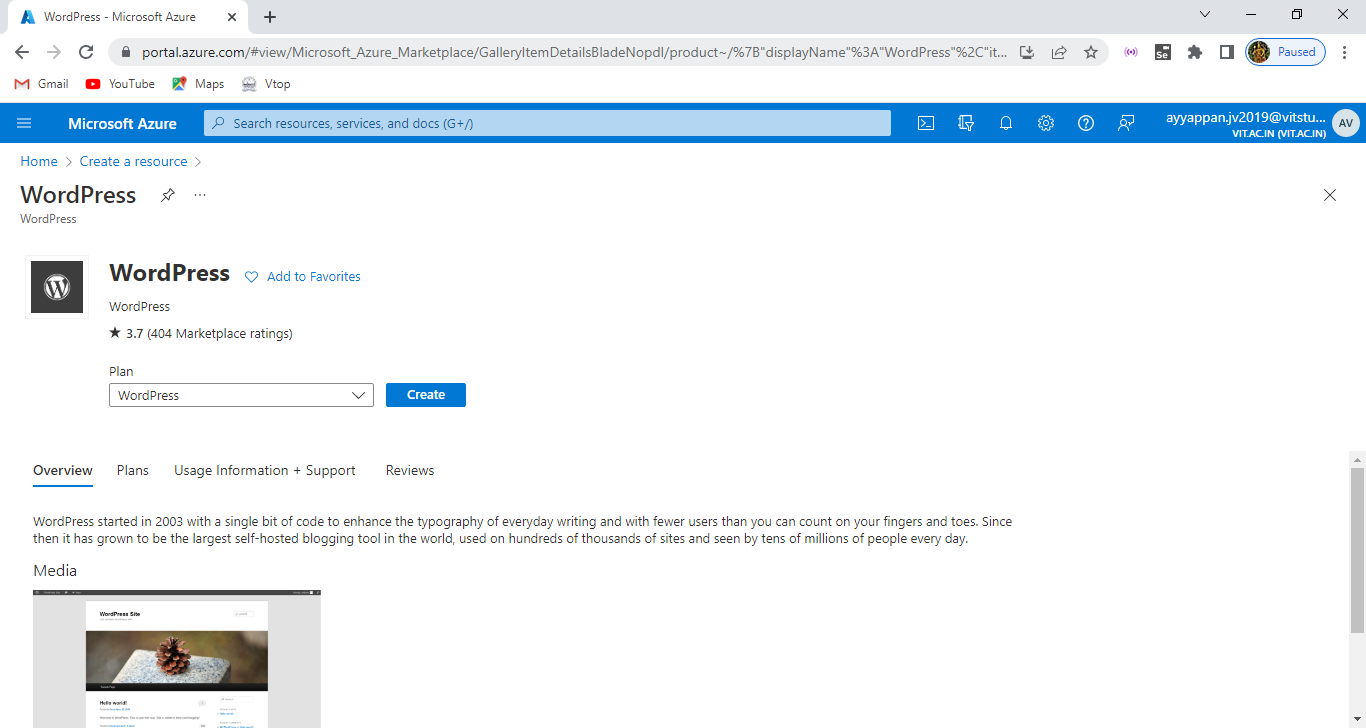
Snapshot - Main page of application ( URL should be available in the screenshot).

Screenshot of additional pages added through admin console.

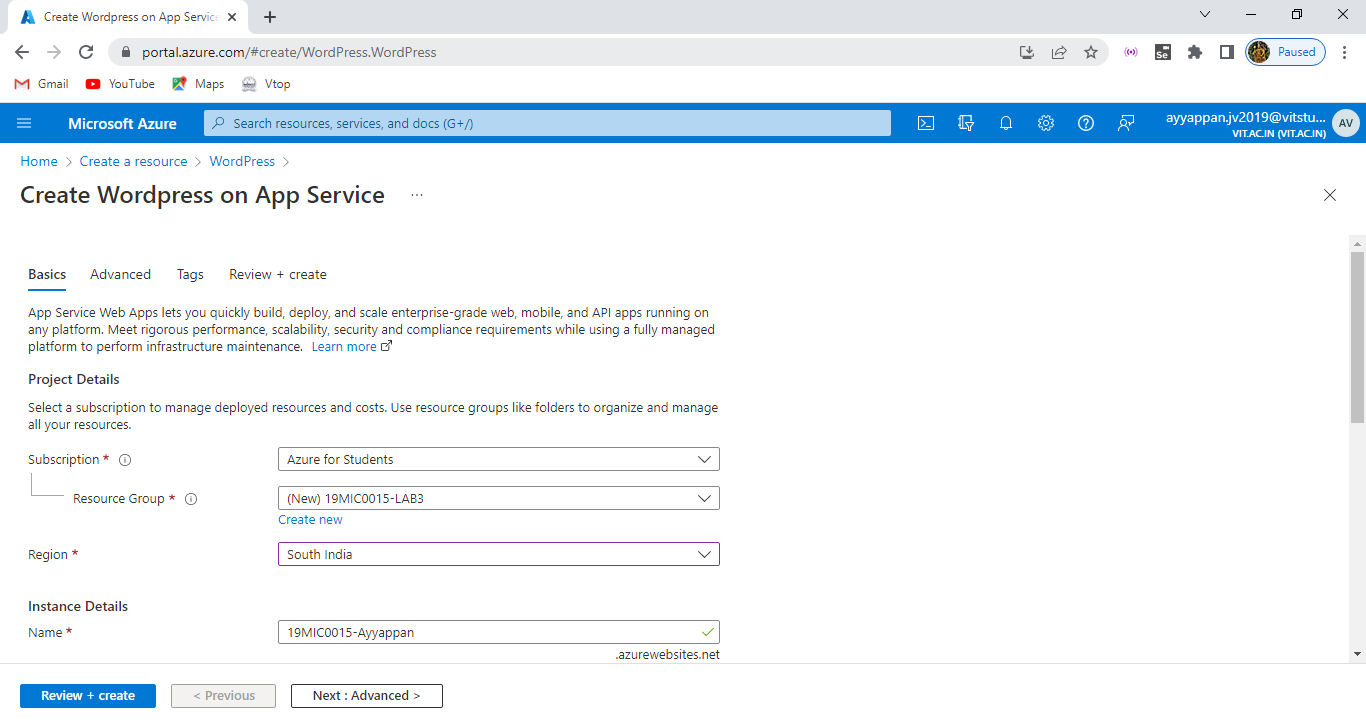
Here click on create resource and type in search box as word press for creating the word press from the Microsoft azure :



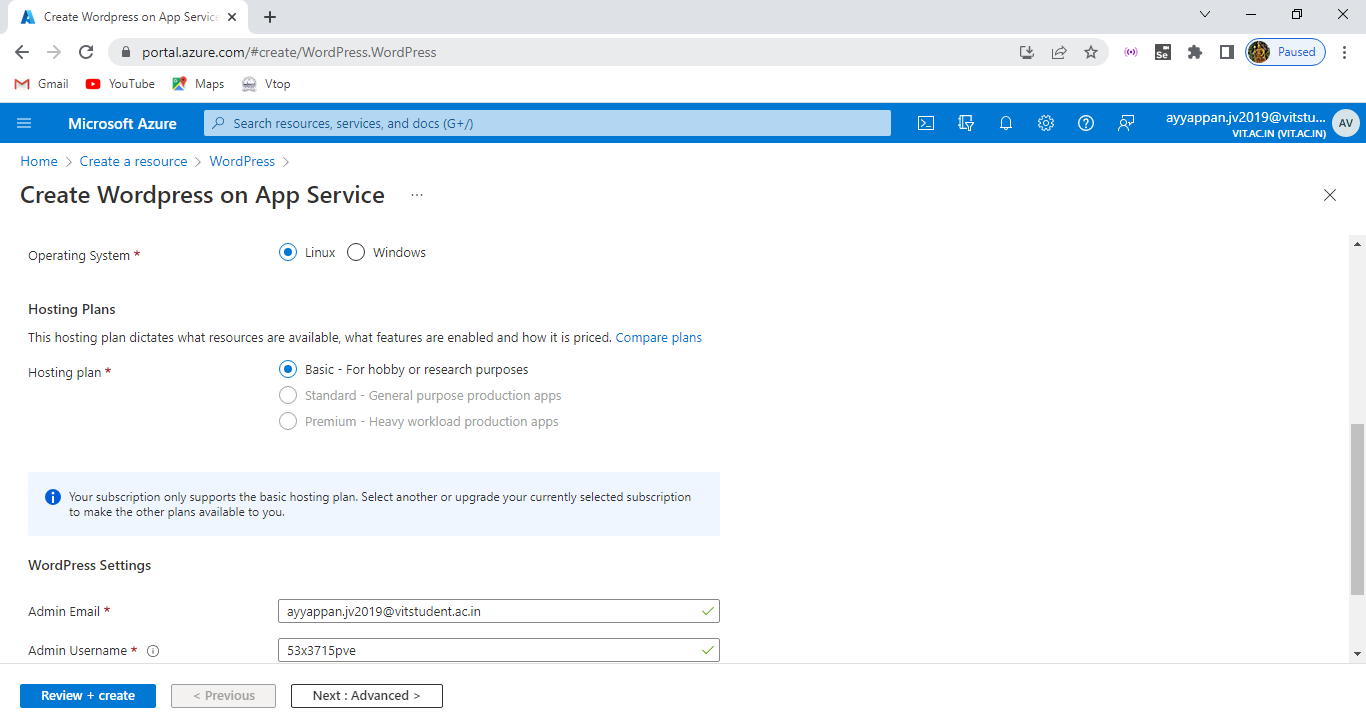
After that word press overview page appears and select the plan as word press and click create:

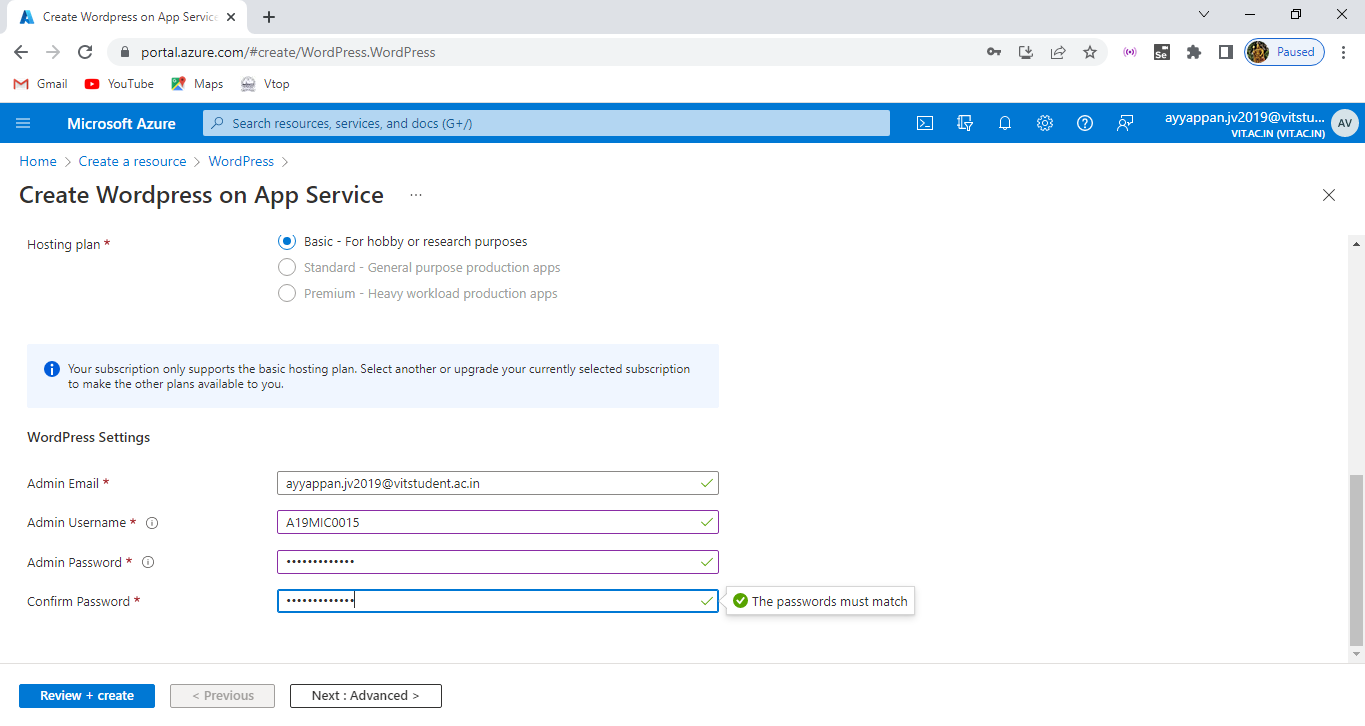


Here the word press app service is displayed enter the resource group and region on given below as follows:

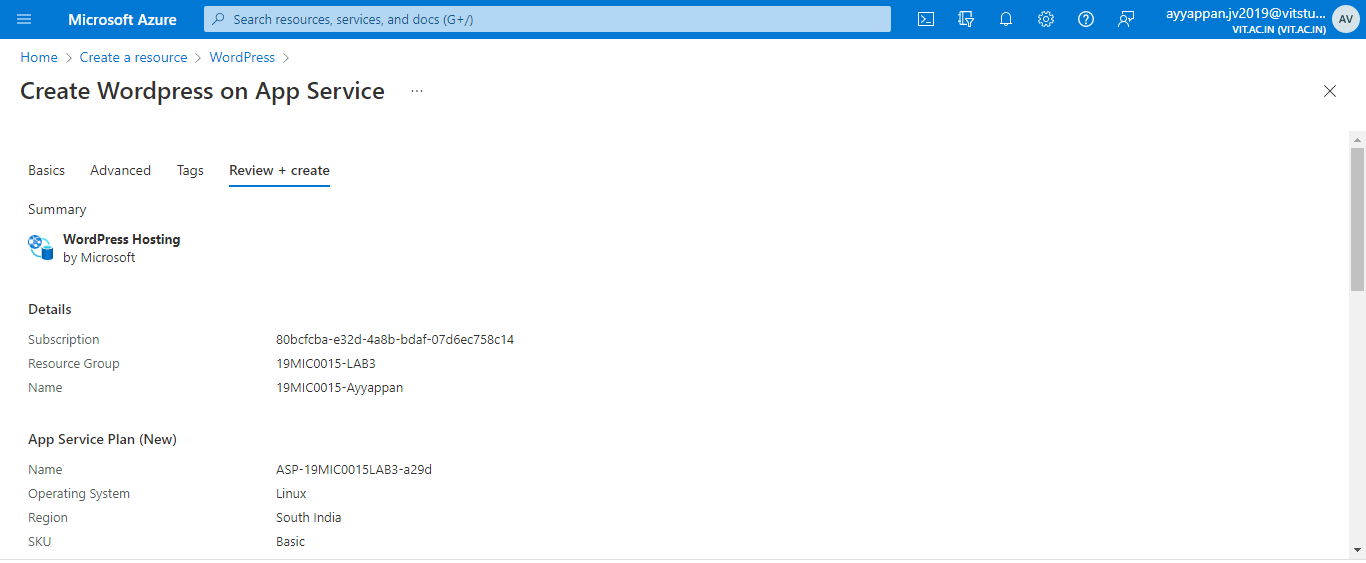


Here by default OS as linux we have choosen and choose the basic plan for the subscription and enter the username and password and click on review + create option:

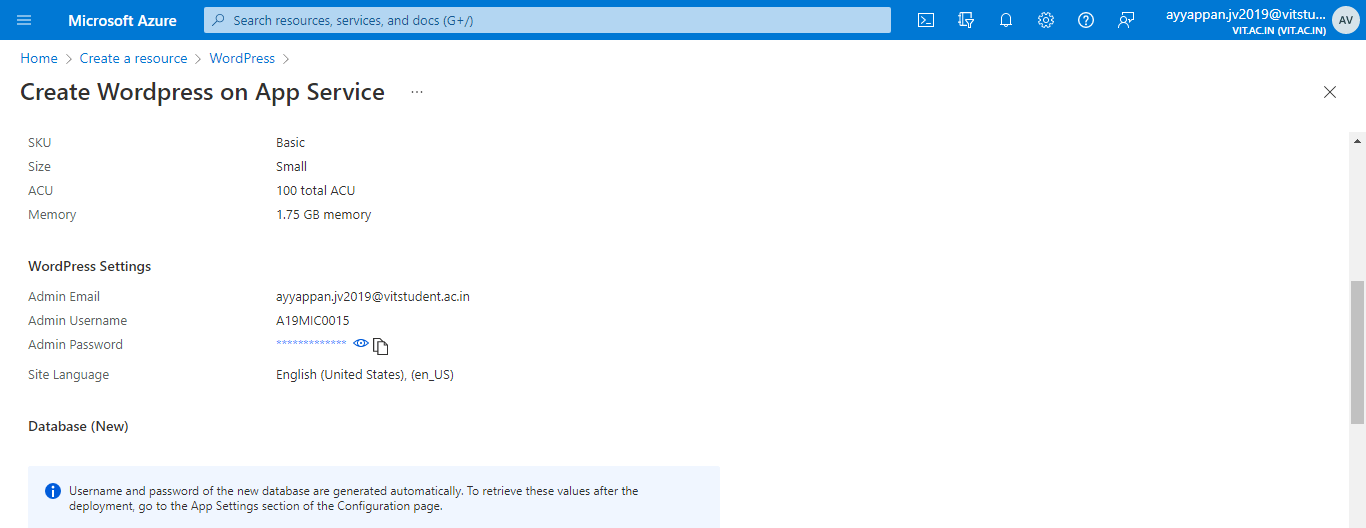


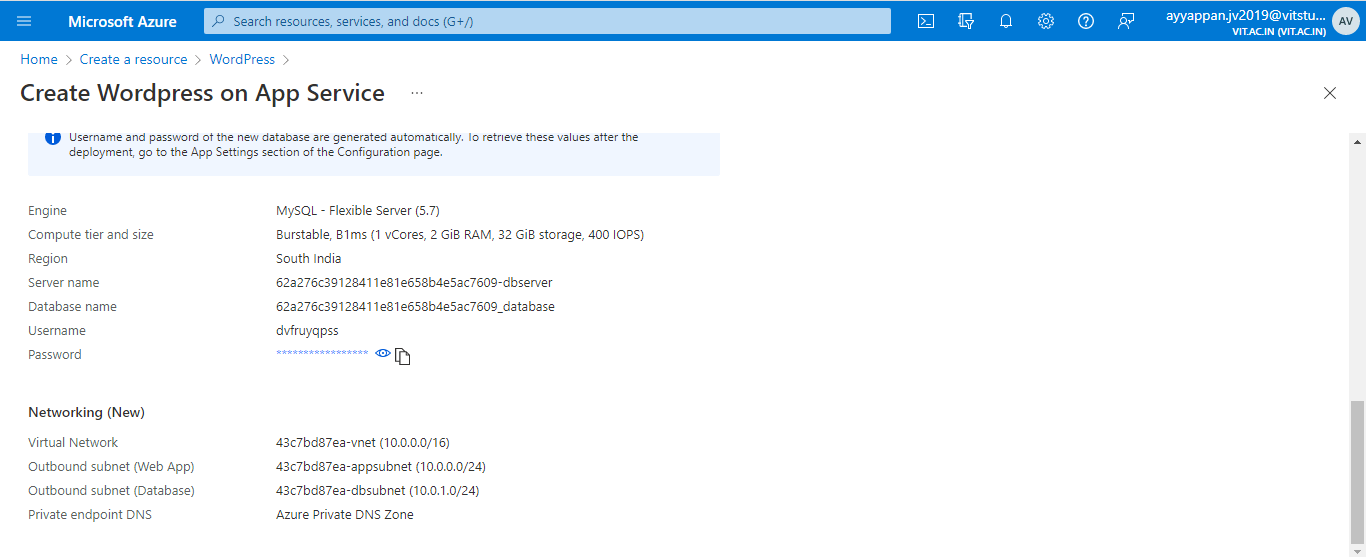


After clicking the review + create option it will display the details , app service plan whatever we given in the creation of wordpress:

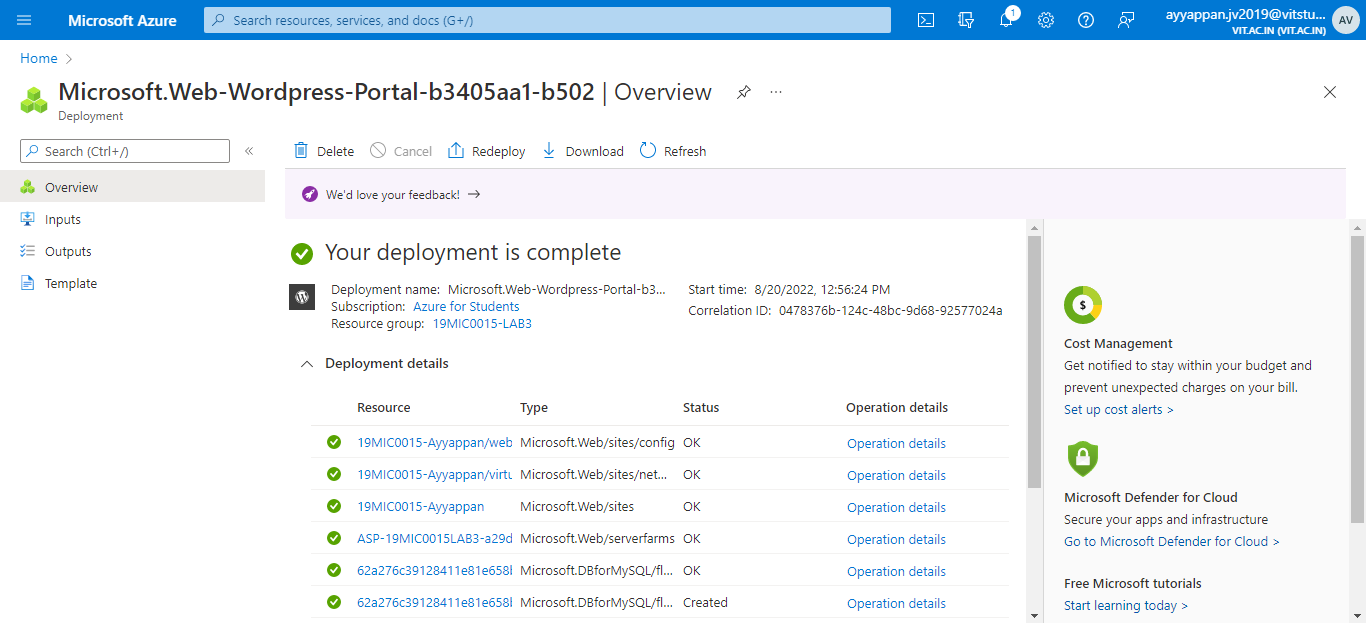


Then it shows the word press settings , networking details of the word press and finally for data storage section it creates default mysql account of it :

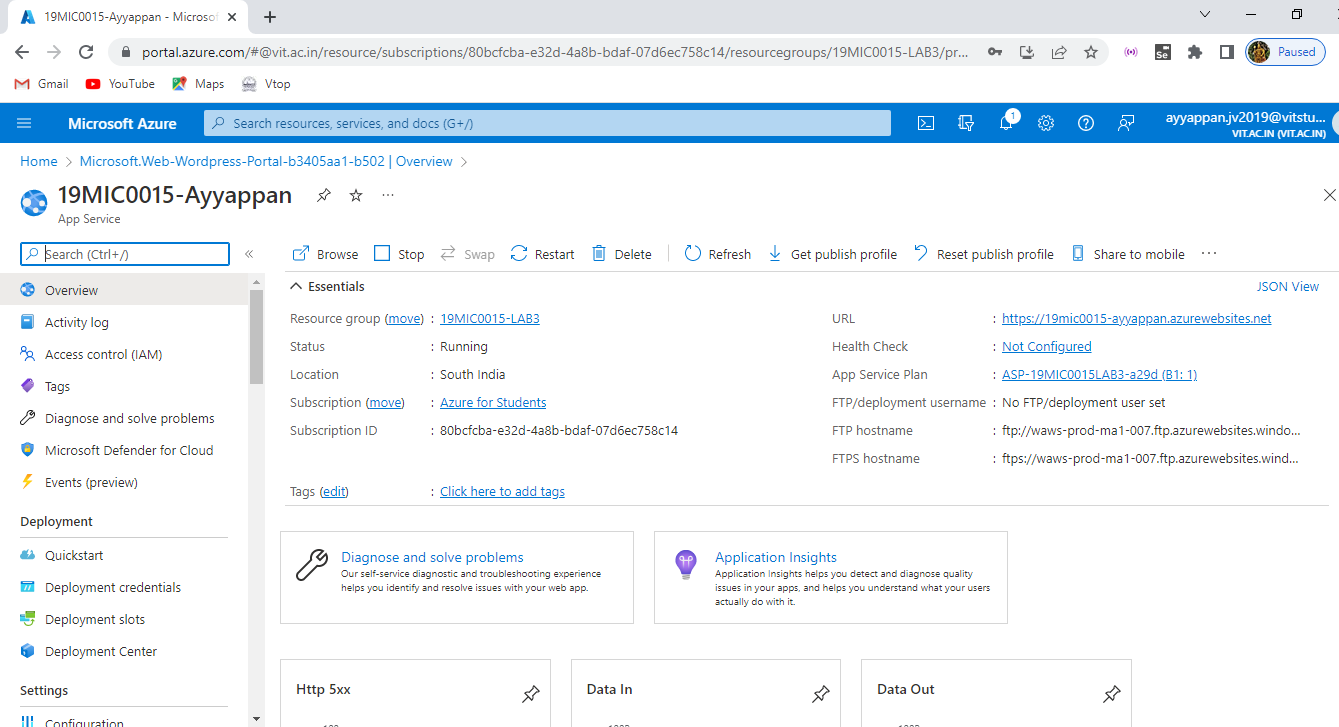




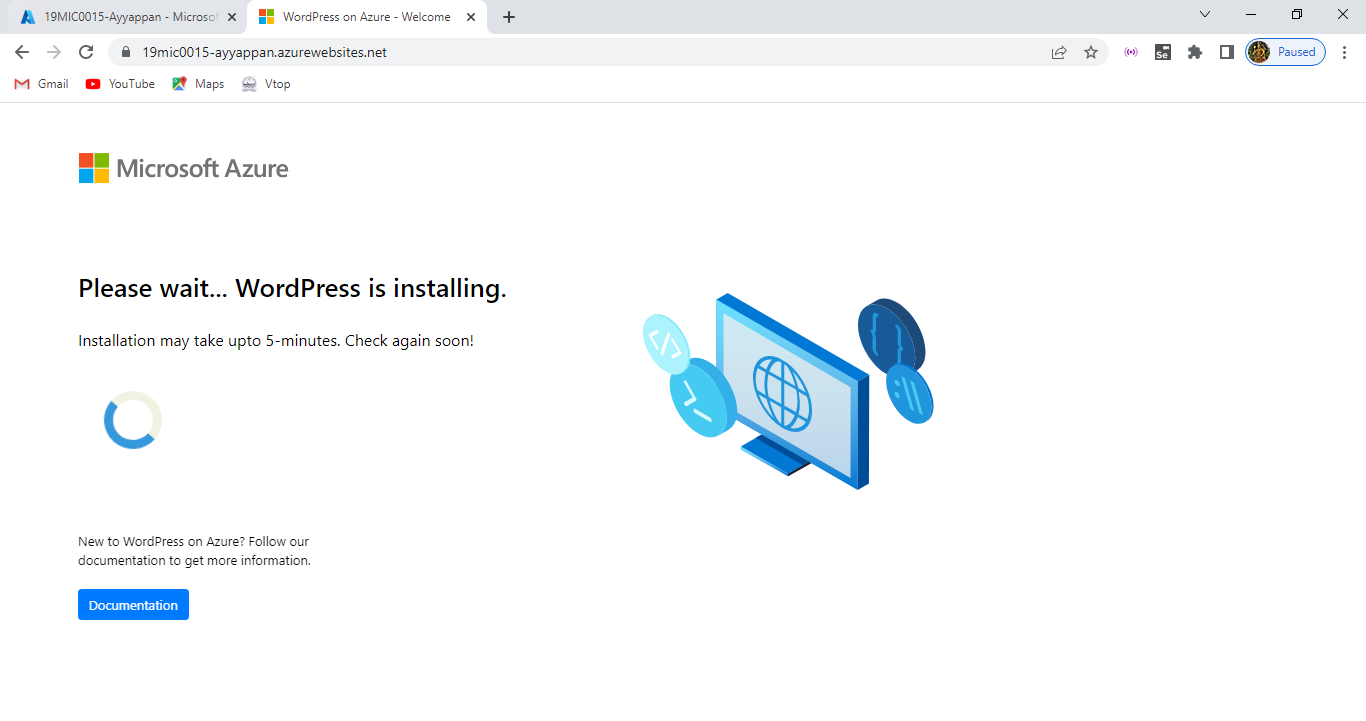
After clicking the create option it take some time to deploy the resource into the azure and finally it is completed:



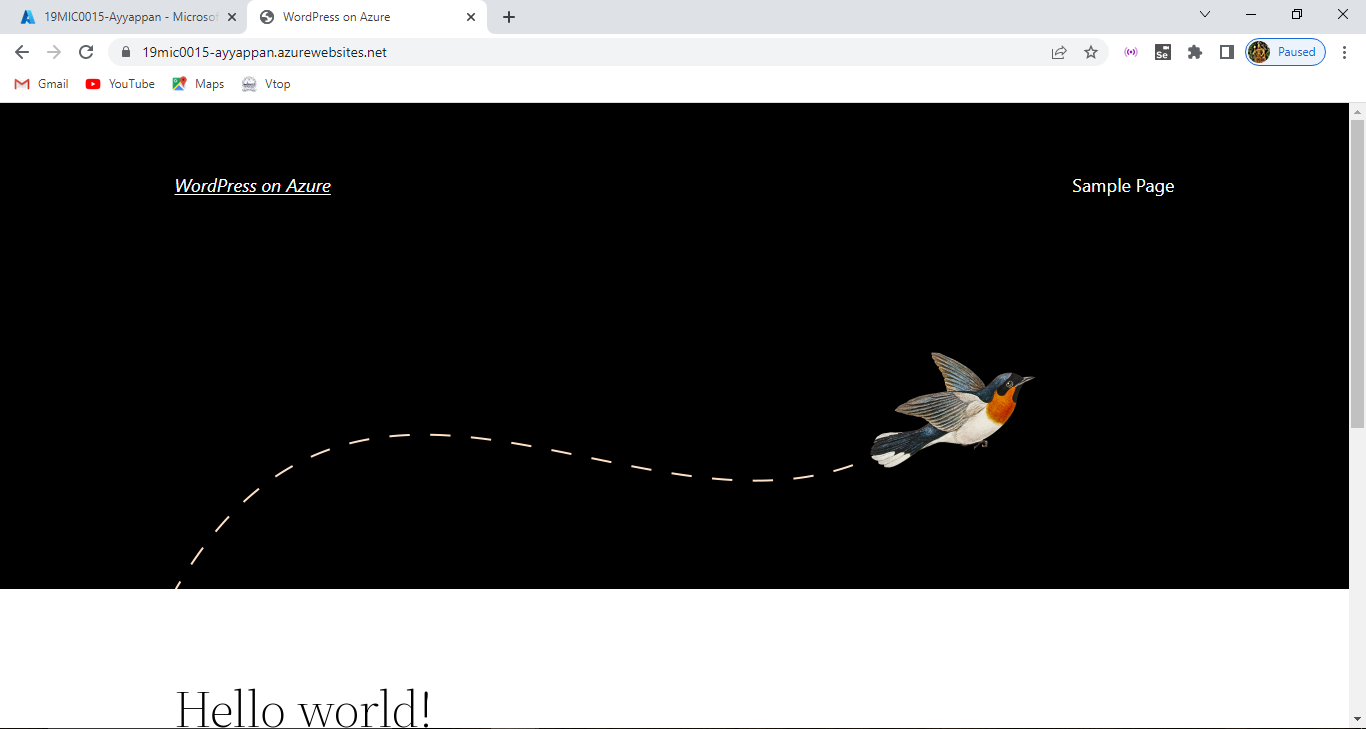
Then click on overview whatever the details we are given while creating the word press that details will be appears in the overview and including the URL of our web page also diaplyed to us:

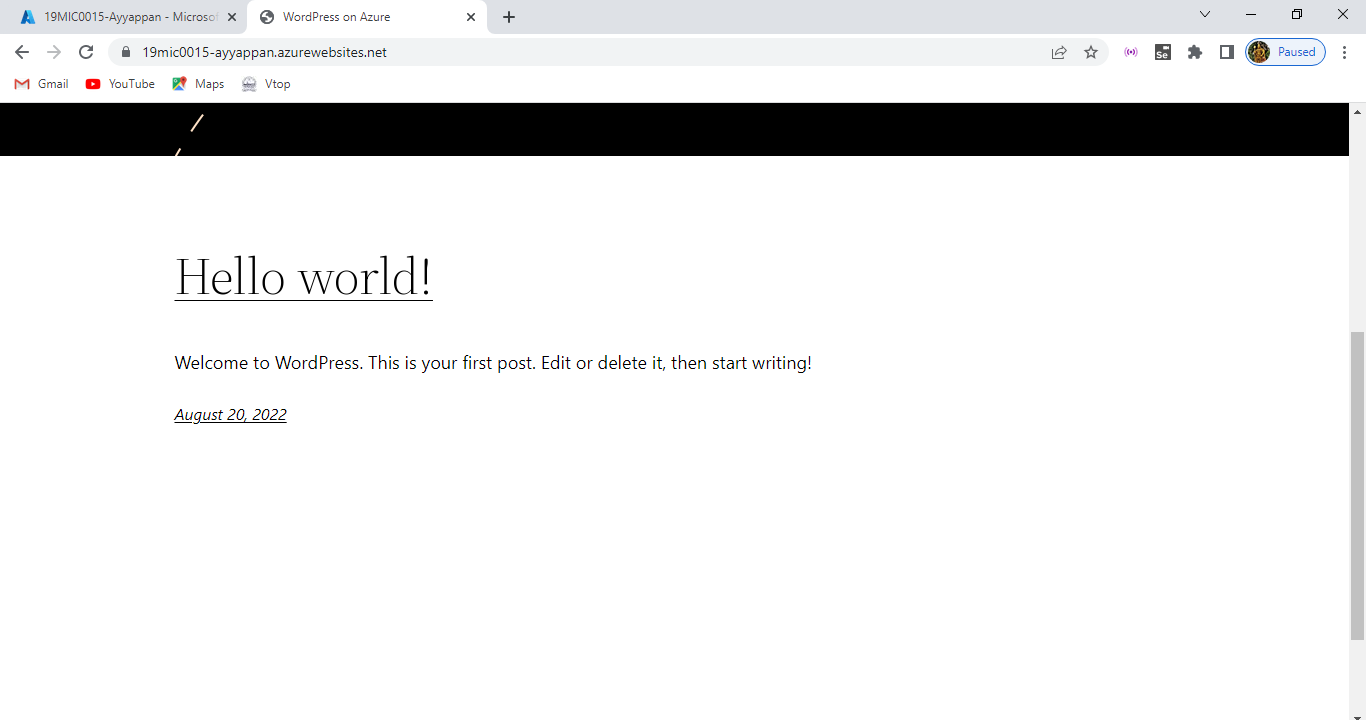


And this is the link for our webpage copy paste in new tab of any browser and wait for some time it get load ad display the sample page: <https://19mic0015-ayyappan.azurewebsites.net/>

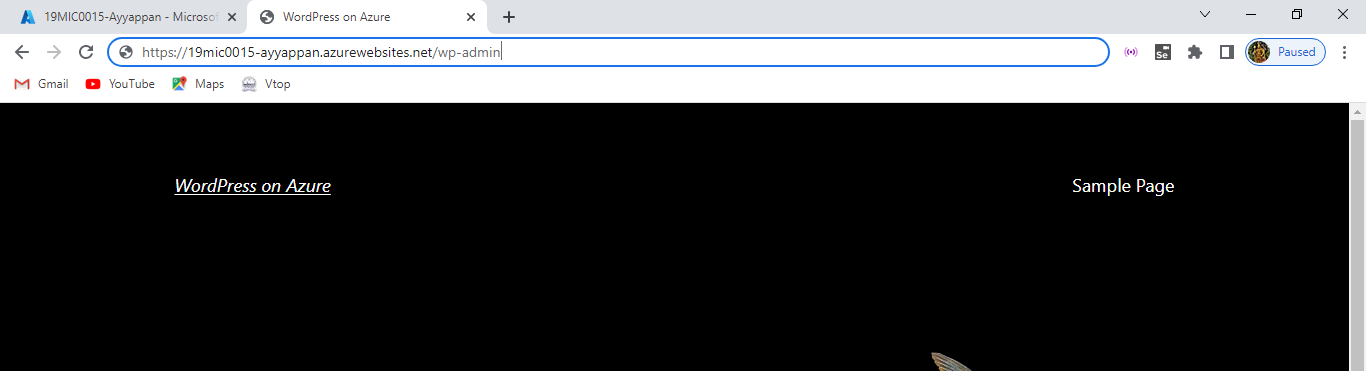


After installation of word press our webpage contains of sample page of azure get displayed here with some welcoming details it display our sample page of it :

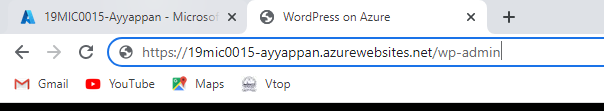




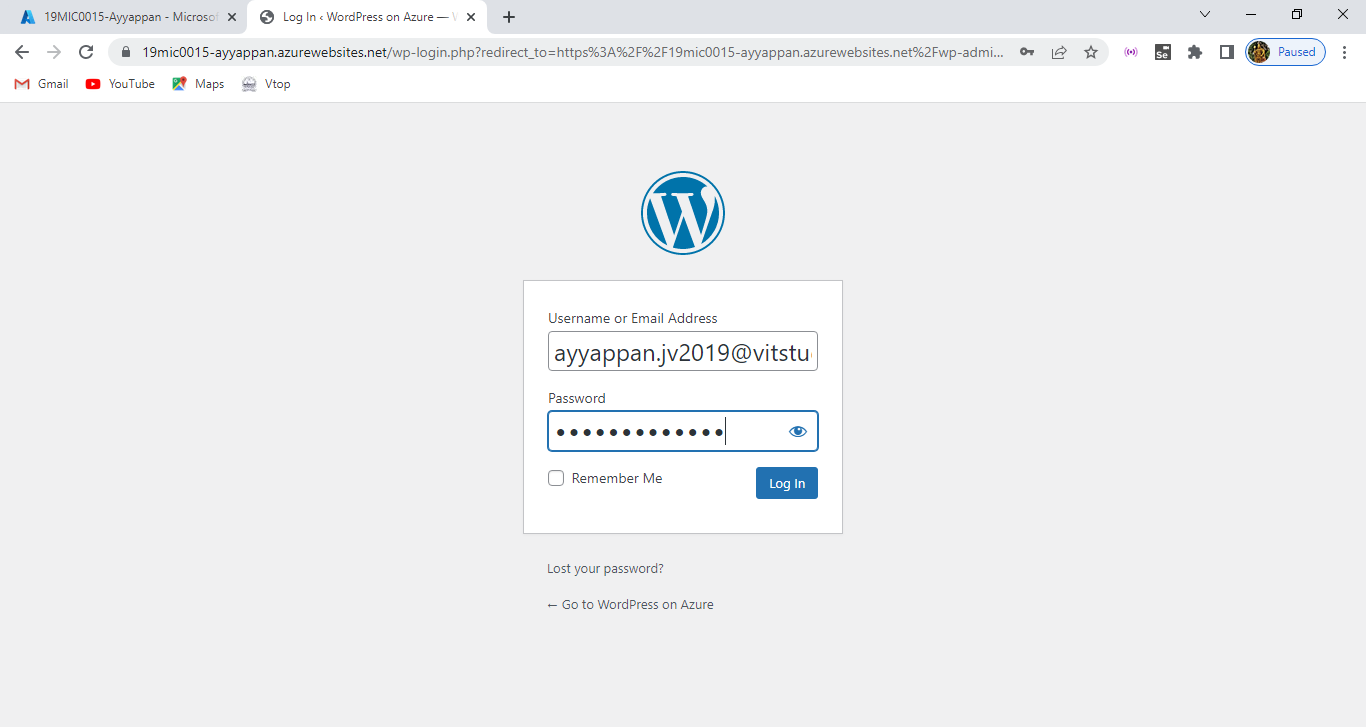
To edit the webpage type after URL /wp-admin it will take us to admin page i.e. editing dashboard login page :



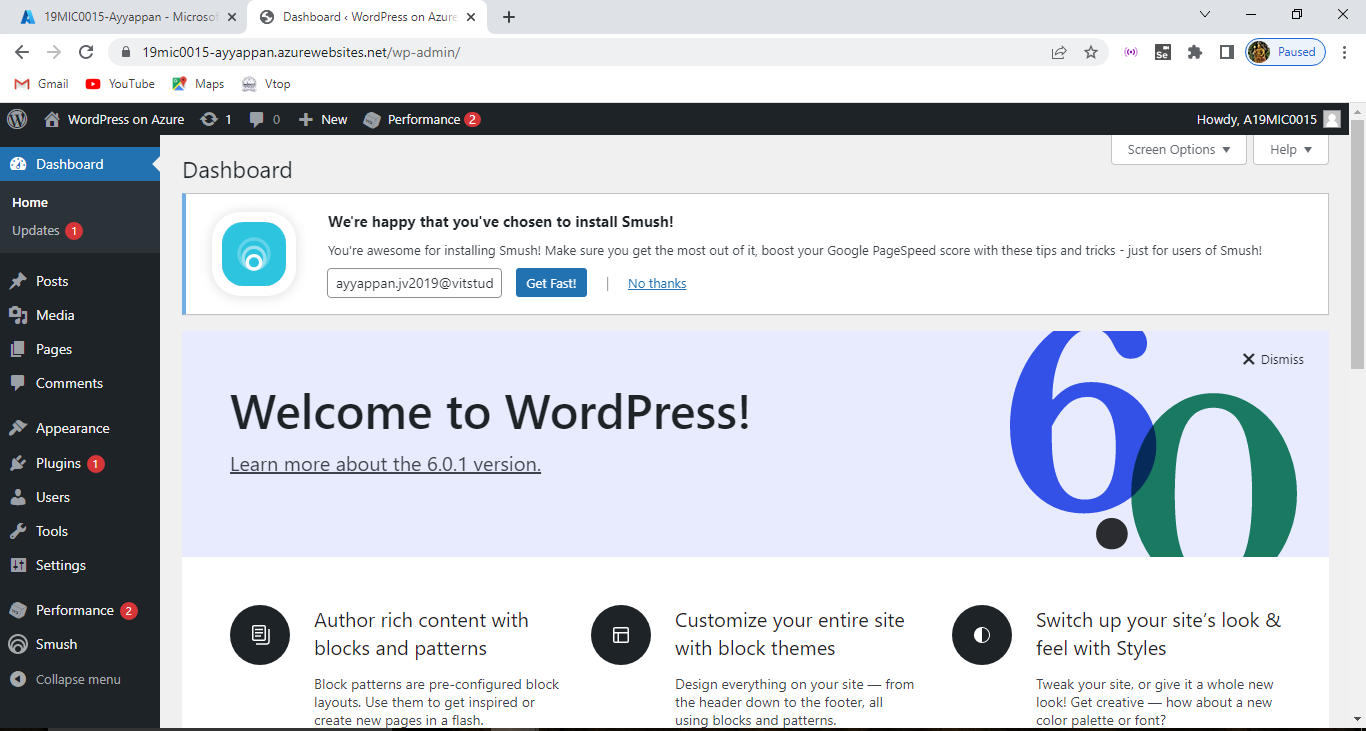
Like this we need to add the command at last of the URL of our webpage of it:



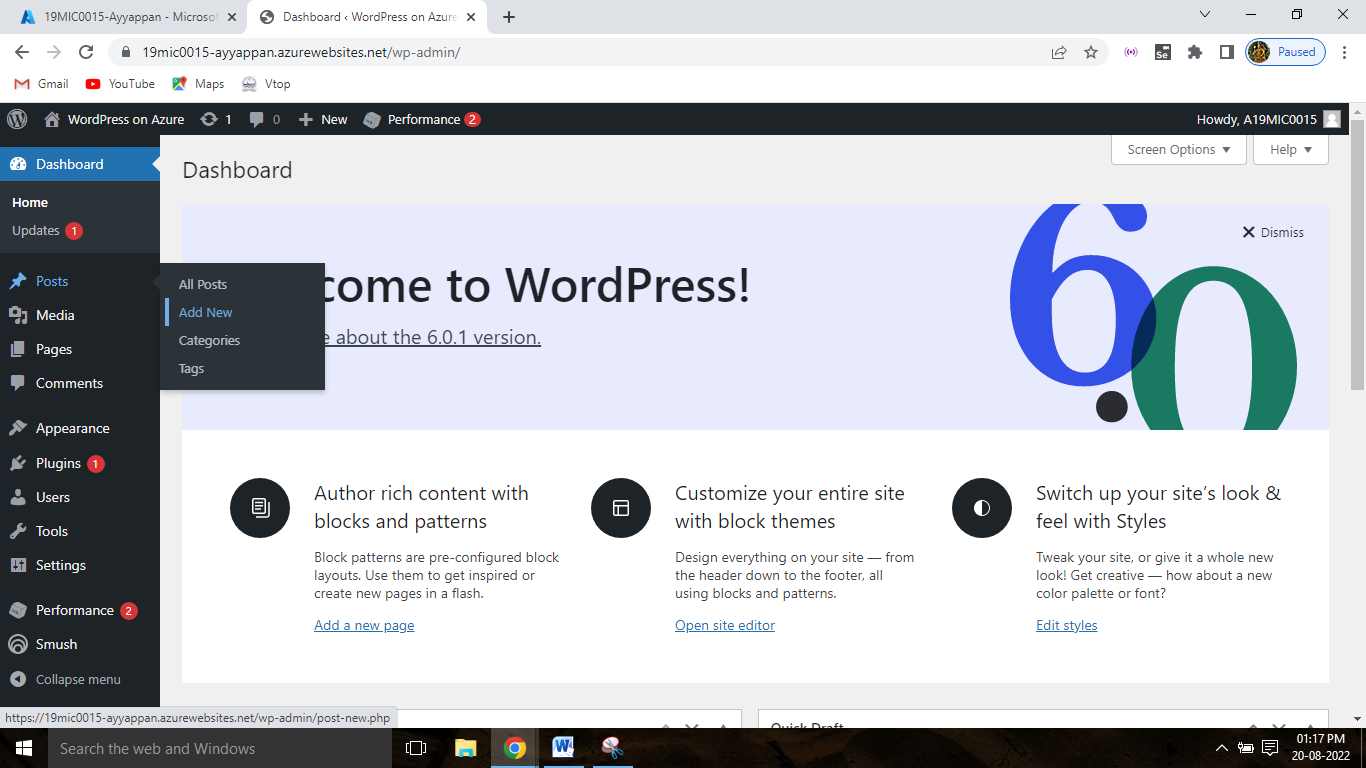
Then it reaches the login page enter the valid username or email and password to login into the editing dashboard page of it:



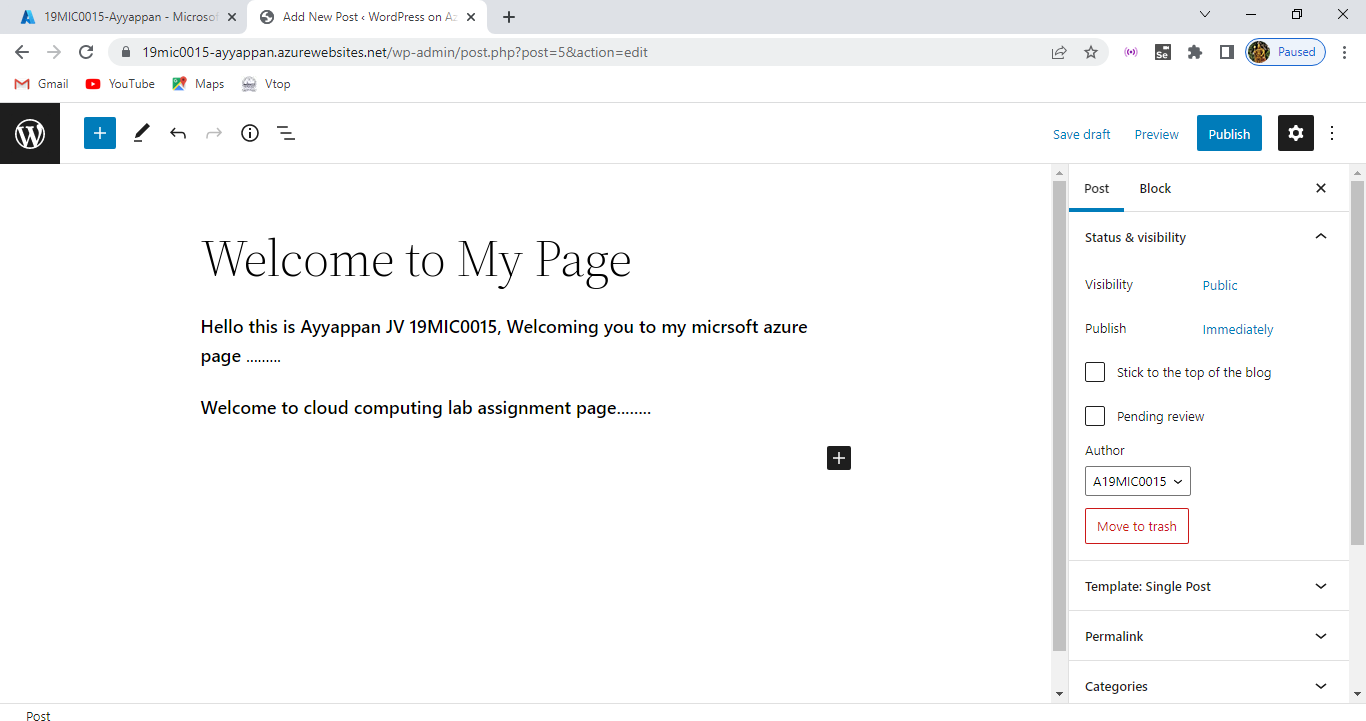
And finally it takes us to the word press dashboard page:



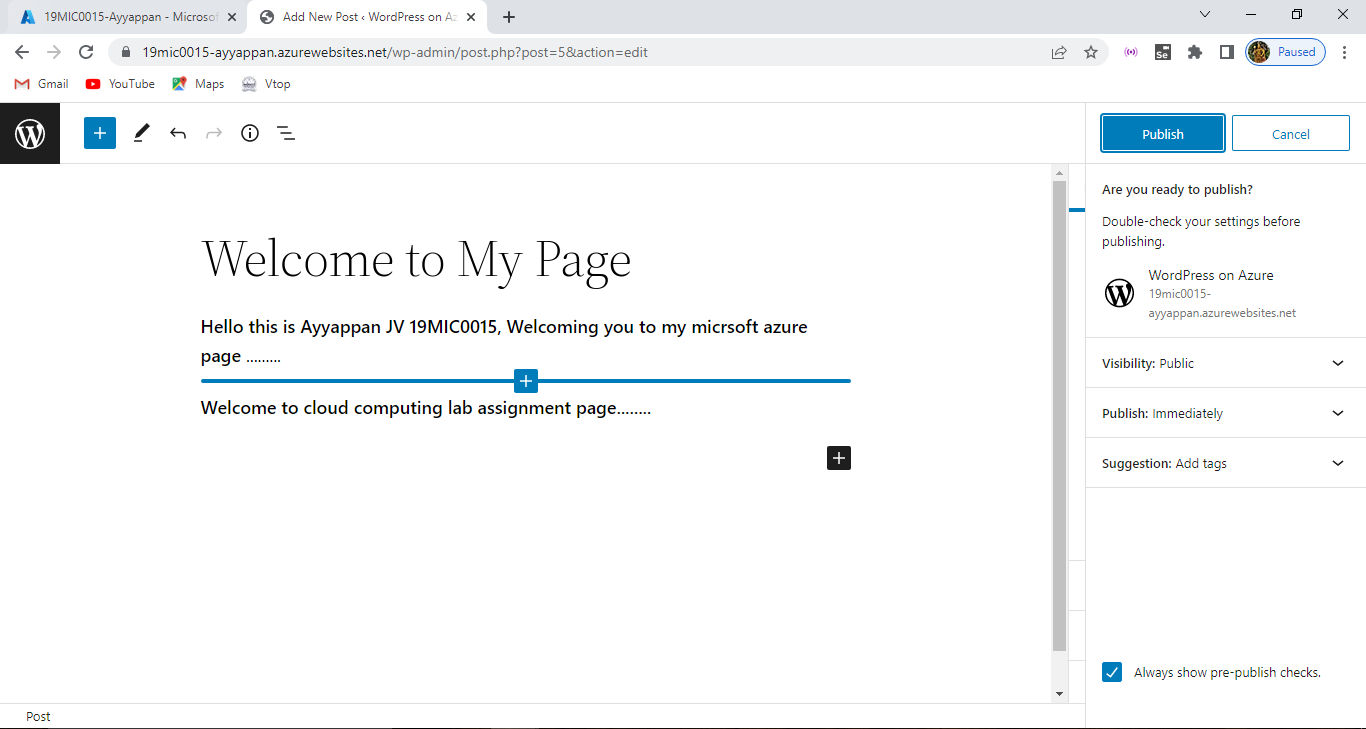
Here at left side click on post then click on add new it will take us to the editing page of our webpage whatever changes or editing that can be added over there:



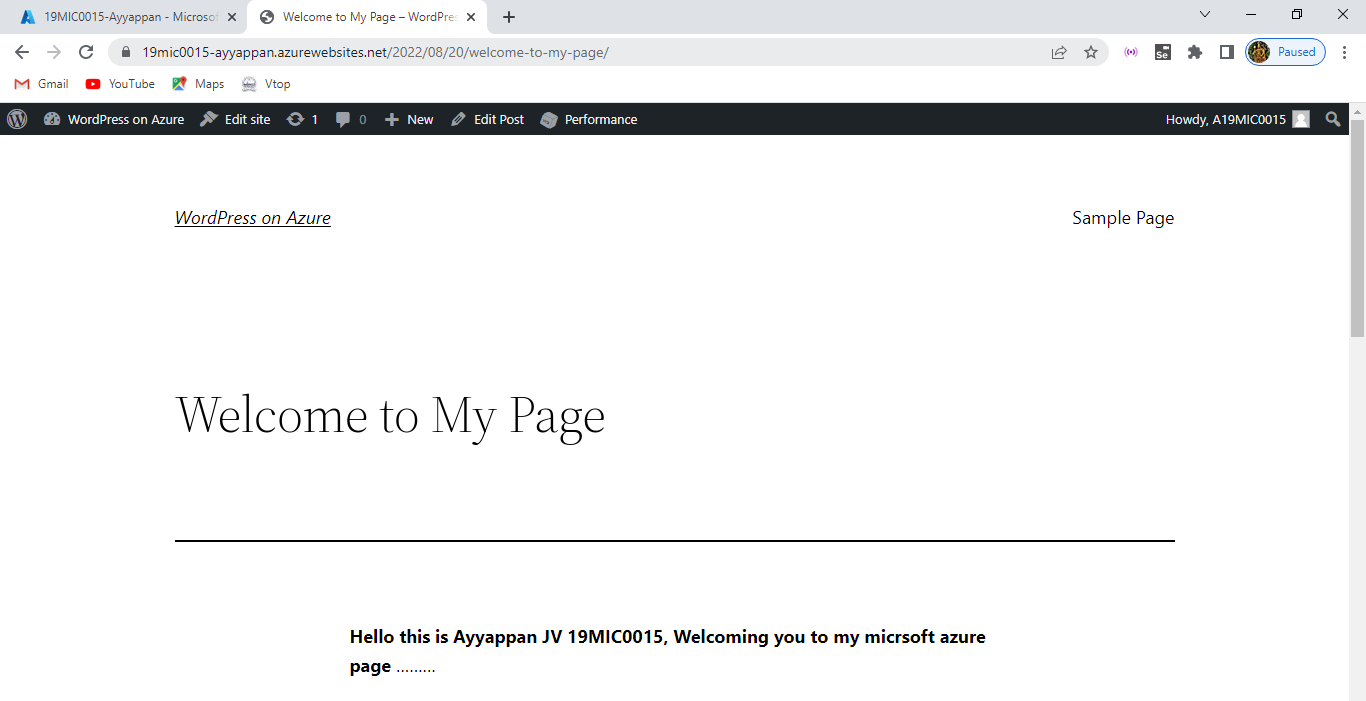
Here we can the editing or paragraph that we are adding into the block and we have done some modification for our web page:



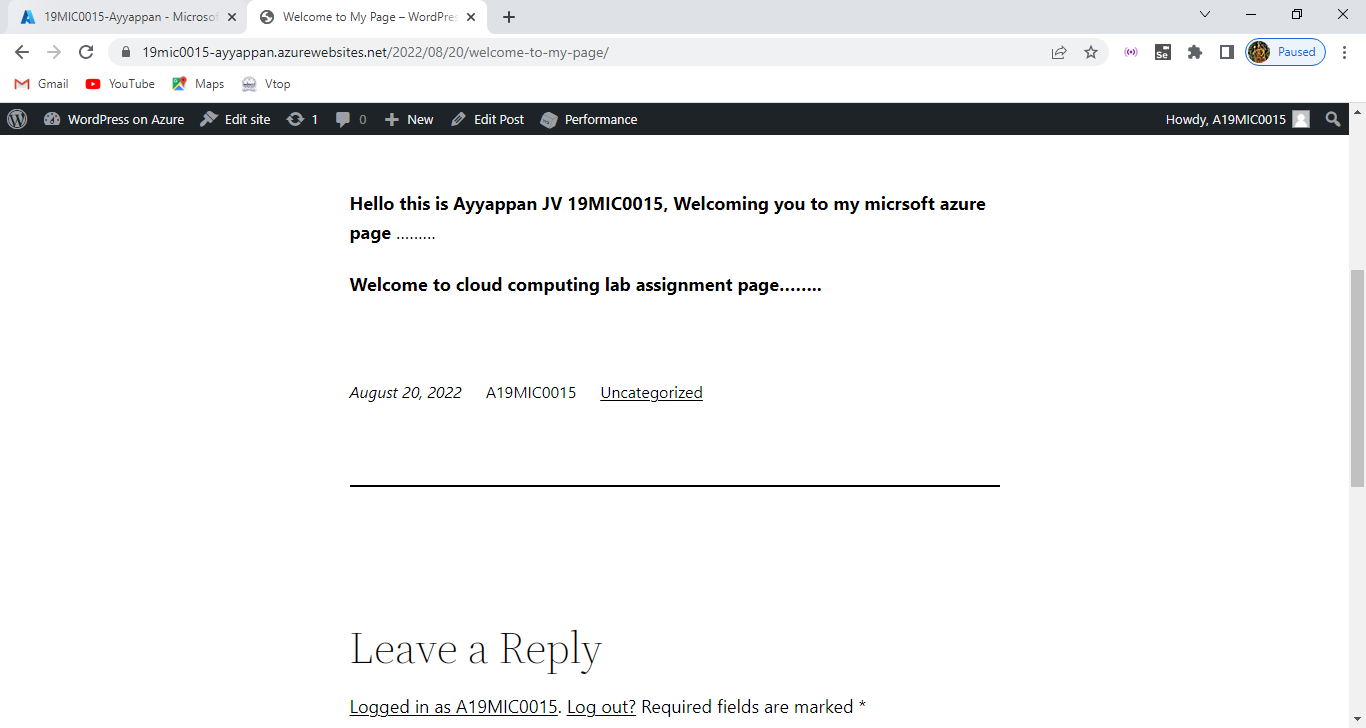
After editing over right top corner there is publish option for publishing those editing page into current page where after publishing whatever changes made yours that’s get reflected over there:



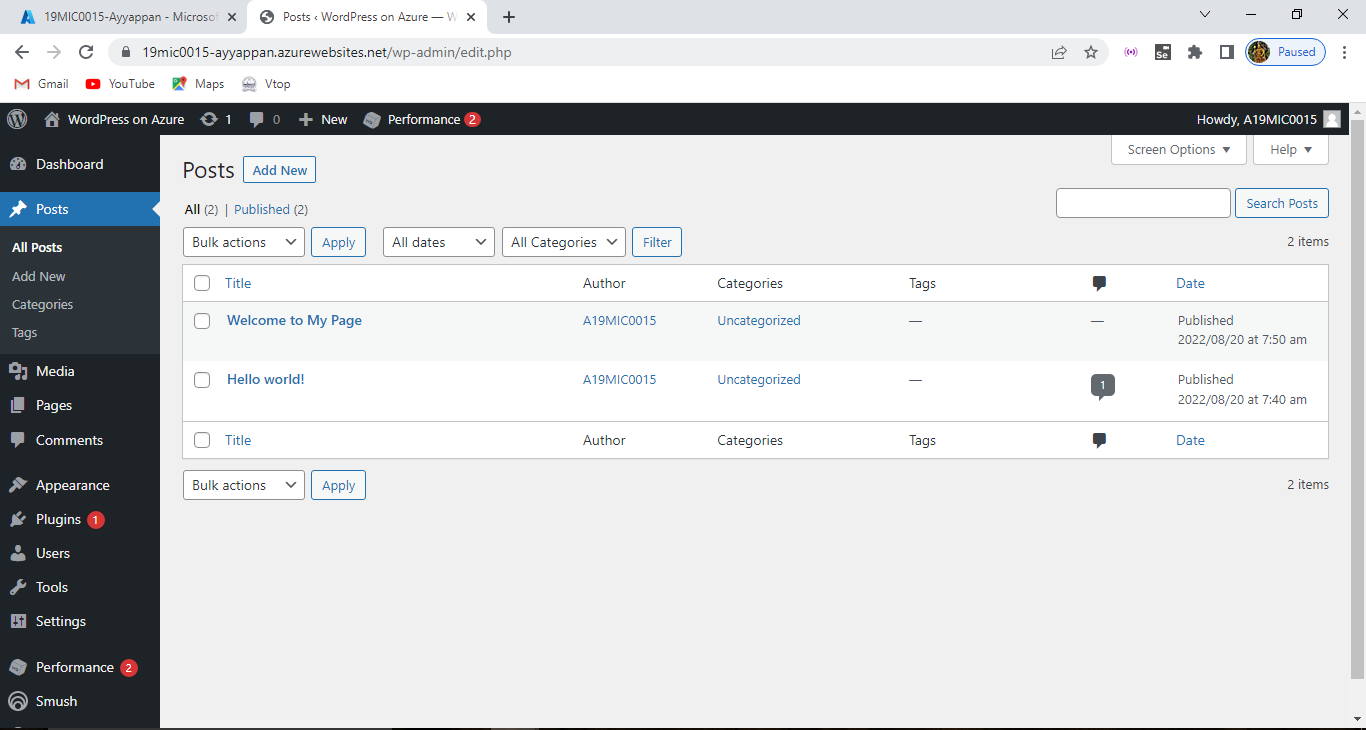
After it gets published the output of the web page is displayed here:



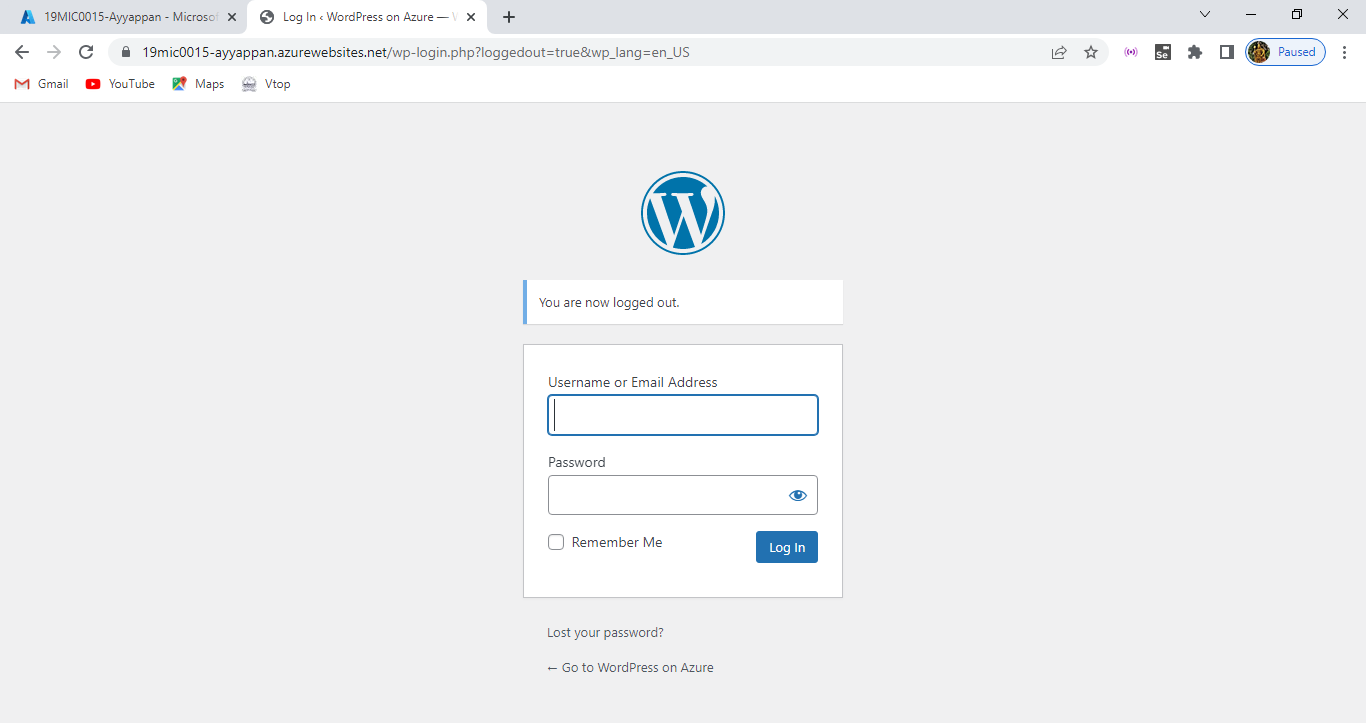
So whatever changes or modifications done, thats get reflected over here and this is how we can edit our webpage by using the wp-admin page of it:



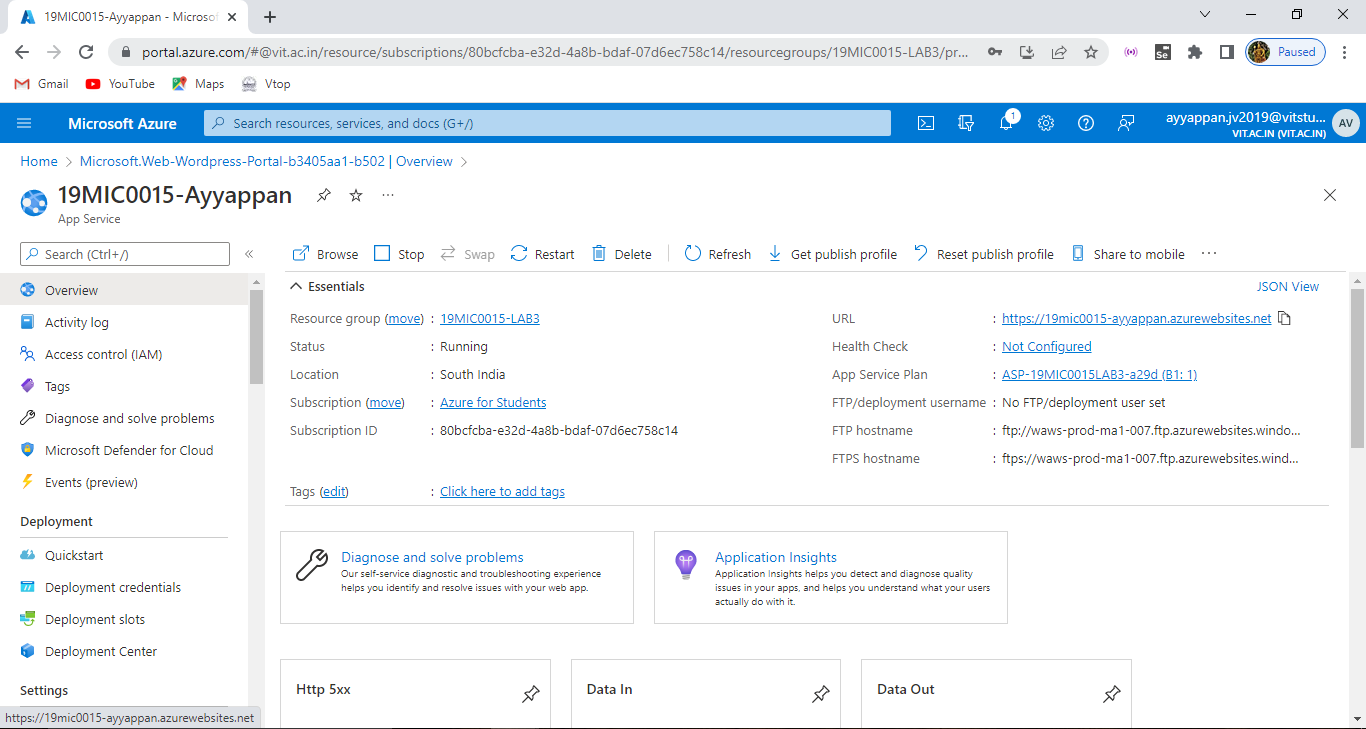
Now we can see the posts whatever editing or any other modification is happened that can be displayed over posts area and we can also add new posts also simantenously:



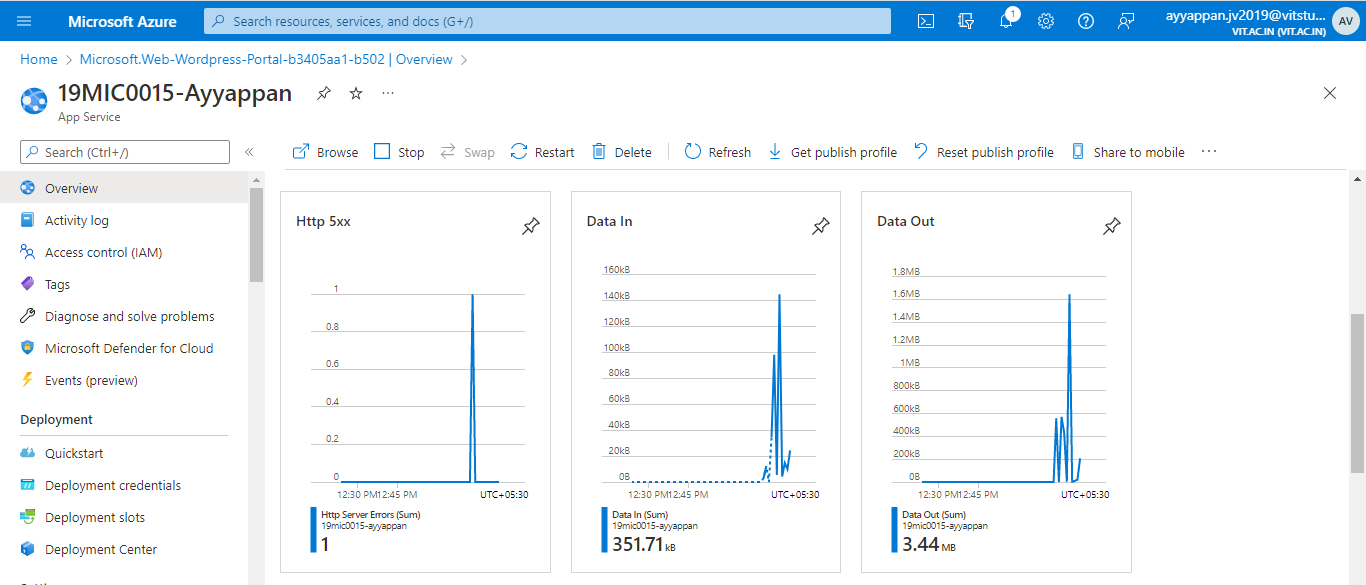
After the editing process over try to logout from there:

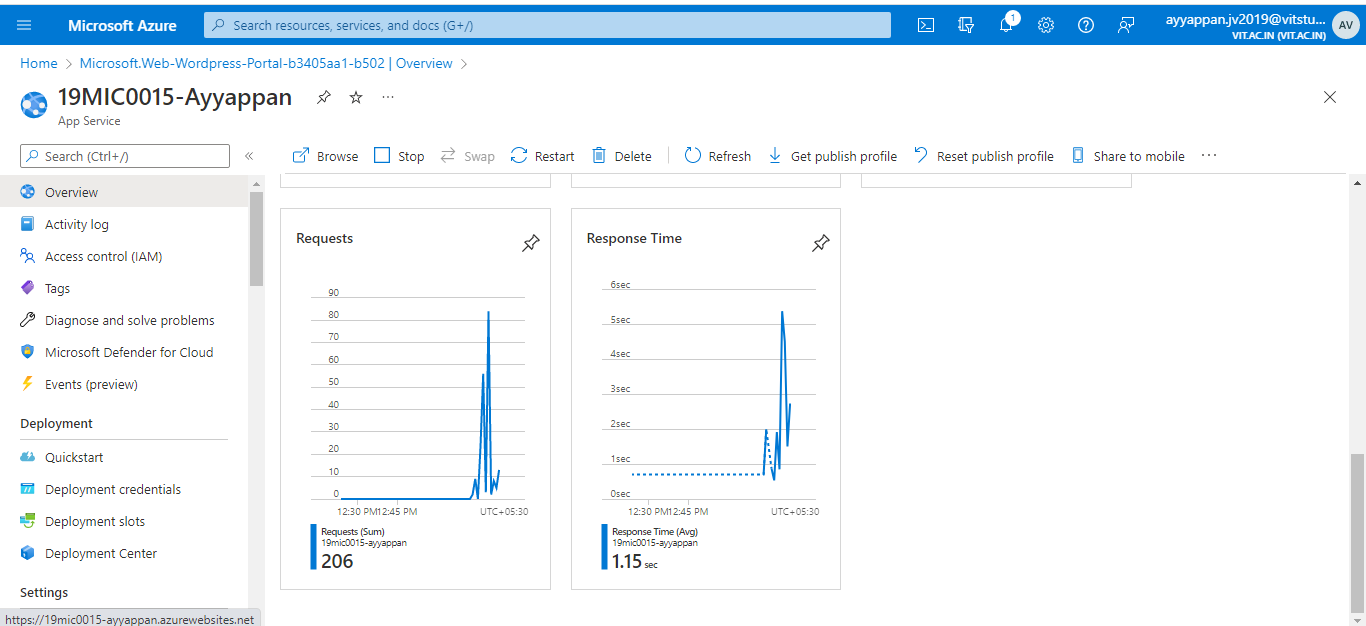


And here you can check the web page is running successfully whenever you click the link it will take you your web page of it and http sxx details , how much data is in and out response time and the number of requests are displayed here in the overview:

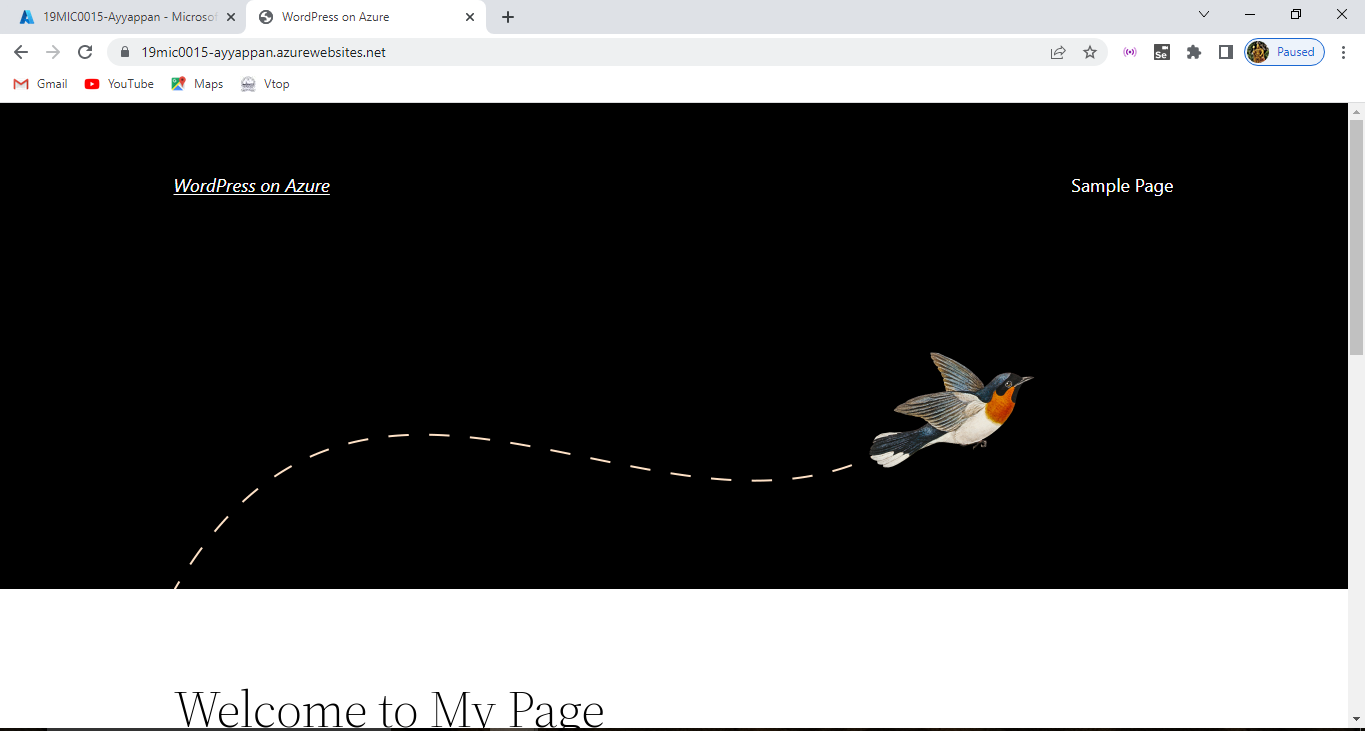


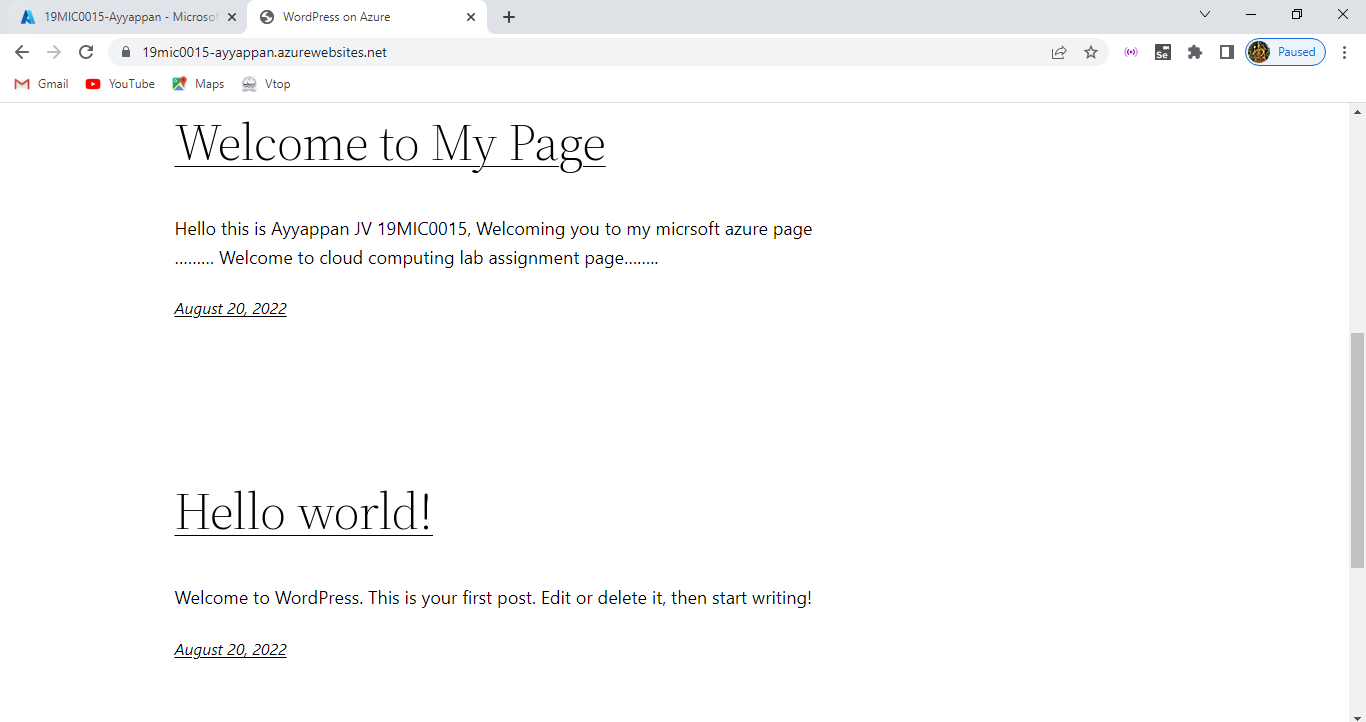
Here the data in , data out , response time and number requests are shown below:



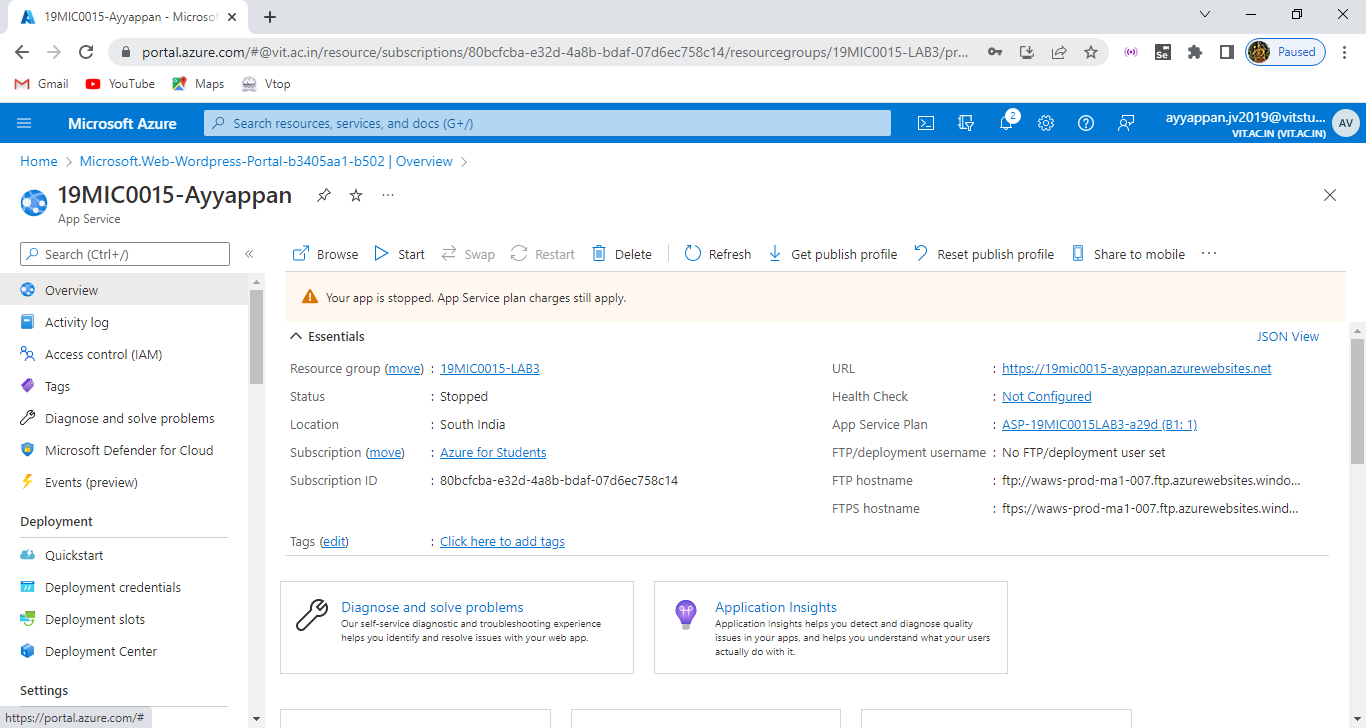


And then whenever you click the webpage URL it will take you to that web page which you have created of it and this page can be accessed by anyone else:

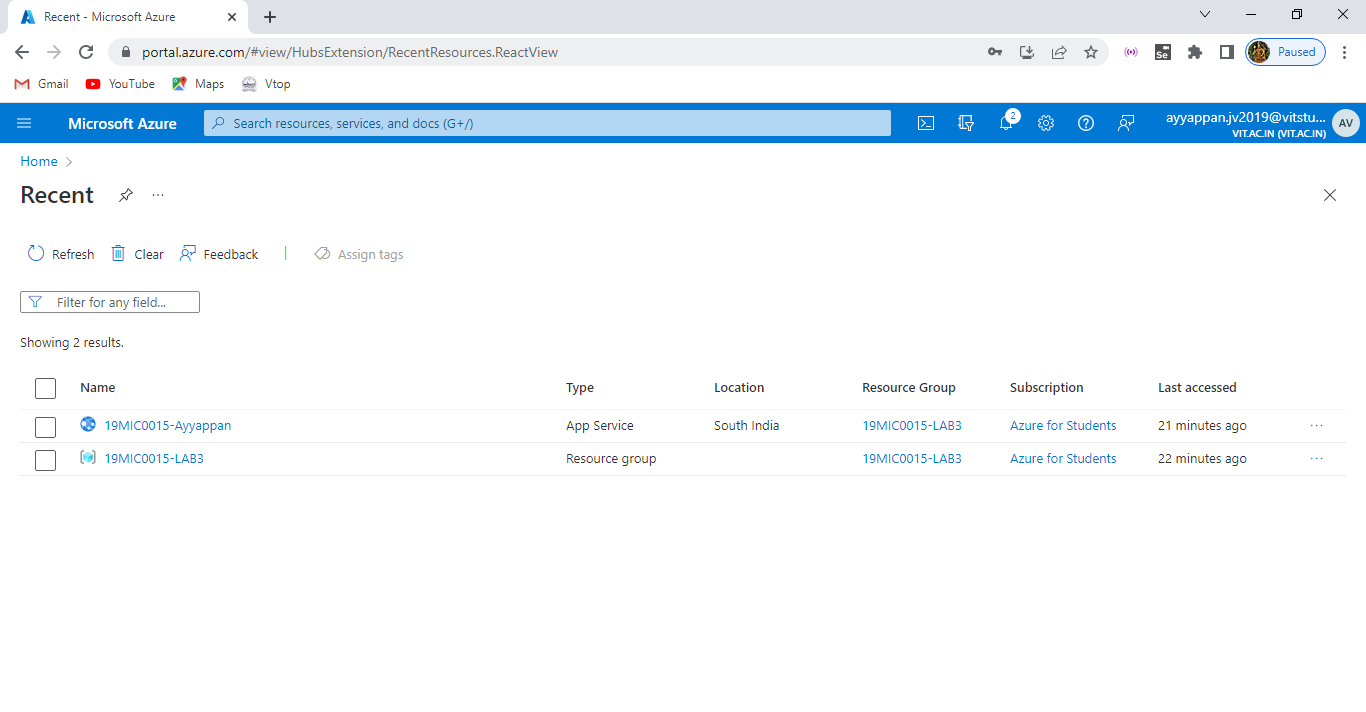




After the process over we need to stop the configuration and after stopping that webpage will be 404 error or site can be reached:



And this the overall view of the Word press that we have created into the resource files get displayed here:

And finally this is the procedure of creating and deploying our own Word Press Site or webpage on Azure Cloud Environment.

Phase – 2:

Create/ clone any sample Web application ( Wordpress/ HTML not allowed).

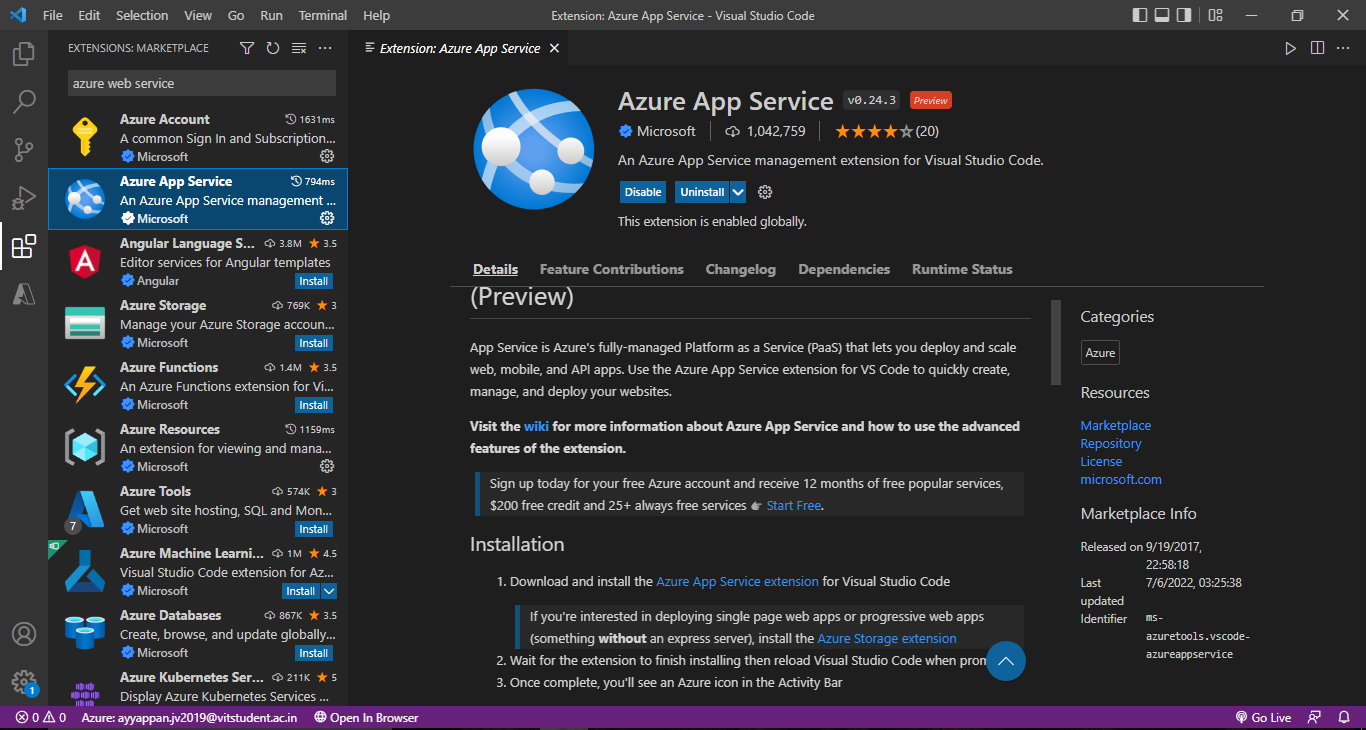
Run the application Locally ( Screenshot).

Deploy and run application in the Cloud environment. ( Detail the steps followed and attach screenshots of each stages, instance name should be your name and register number).

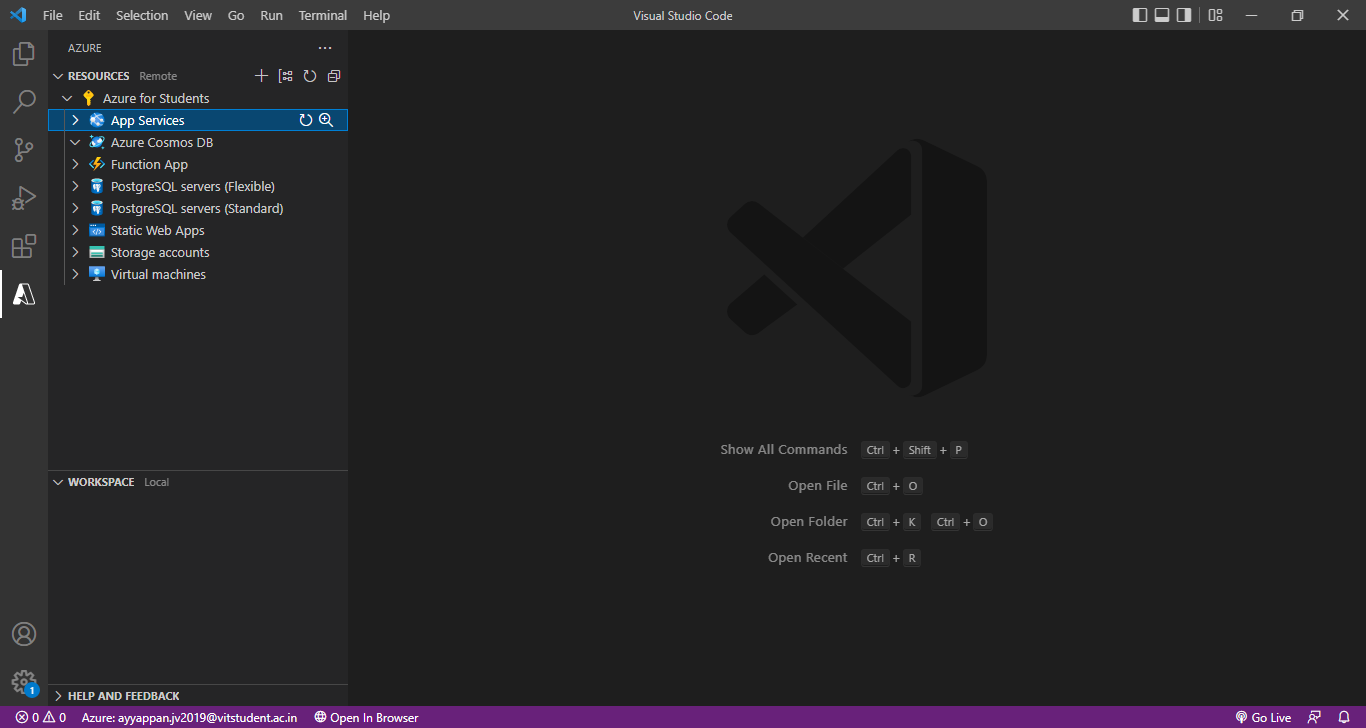
Finally add the  screenshot of running application( Index/Main) Page with the URL( register number/name should be available in the URL).

Solution:

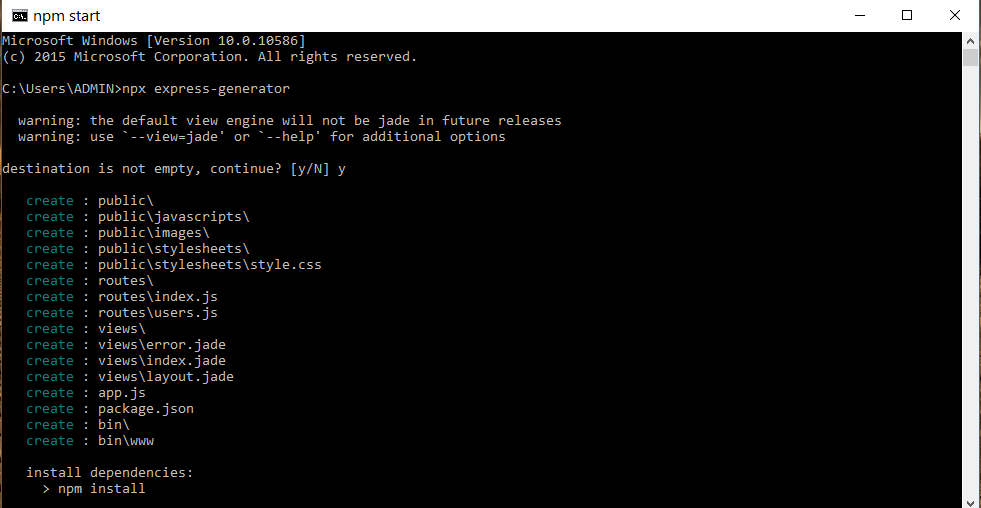
Here we are creating the sample web application using node JS deploy on azure web services. First set up our environment node js and npm, download visual studio code and then azure app service extension using cmd locally should be done. Then open visual studio code in extensions type azure app service and download that app extensions.



After downloaded try to login azure account through this and list of services will be available.

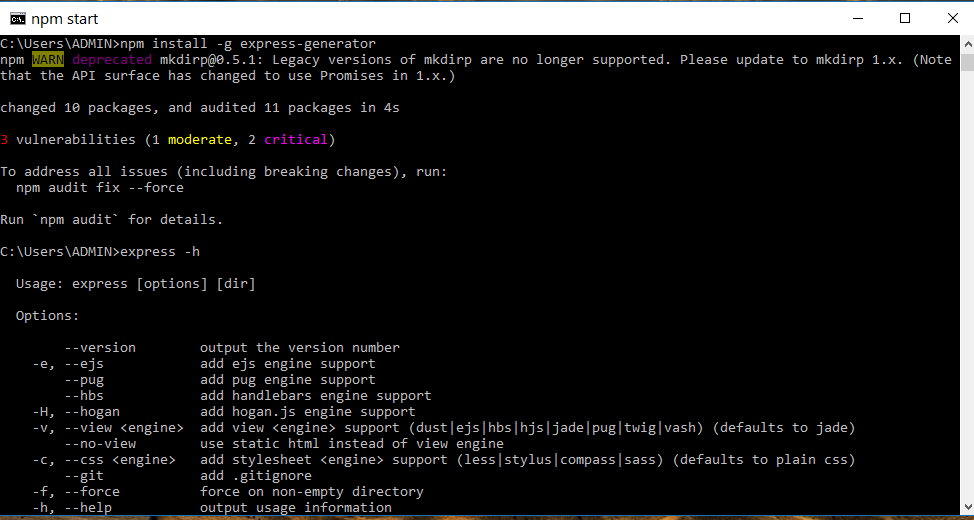


Go to local terminal and use all these commands on our local pc in order to install node js application and to ensure runs on our computer.



Use npm command to install those express files and folders in your local system.

Then type the command as express –h which gives the different directory files available.

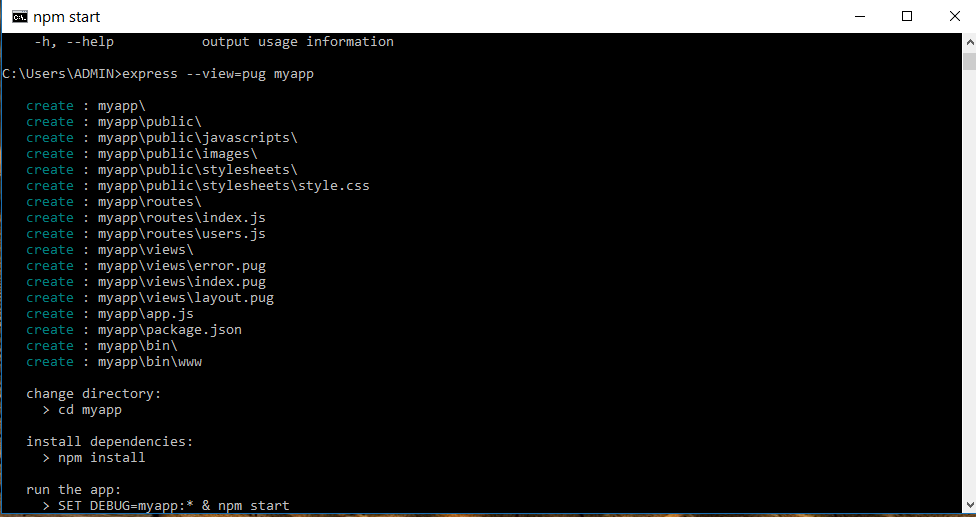


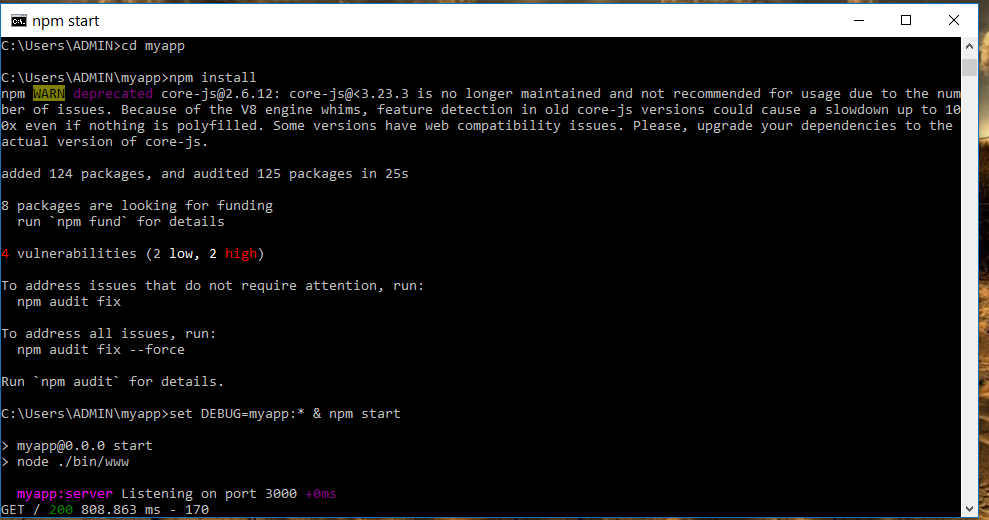
npx express-generator myExpressApp --view ejs

cd myExpressApp && npm install

DEBUG=myexpressapp:\* npm start

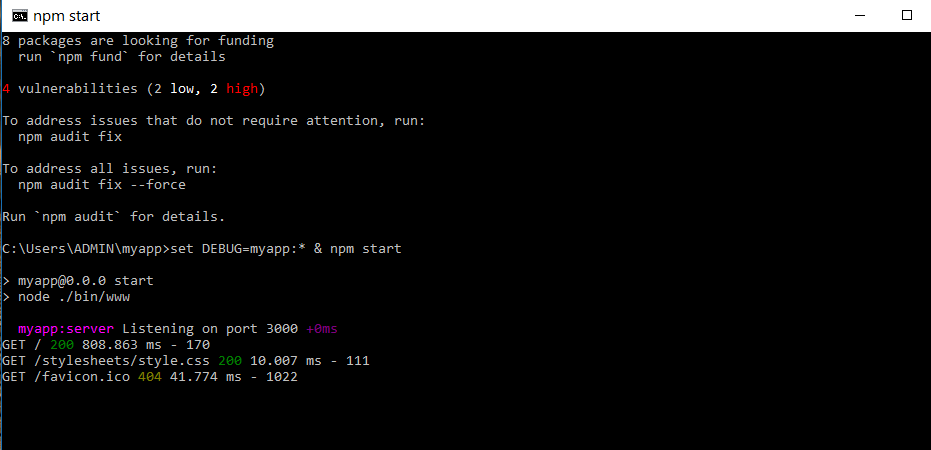
Use this command express the generator file on VSC.





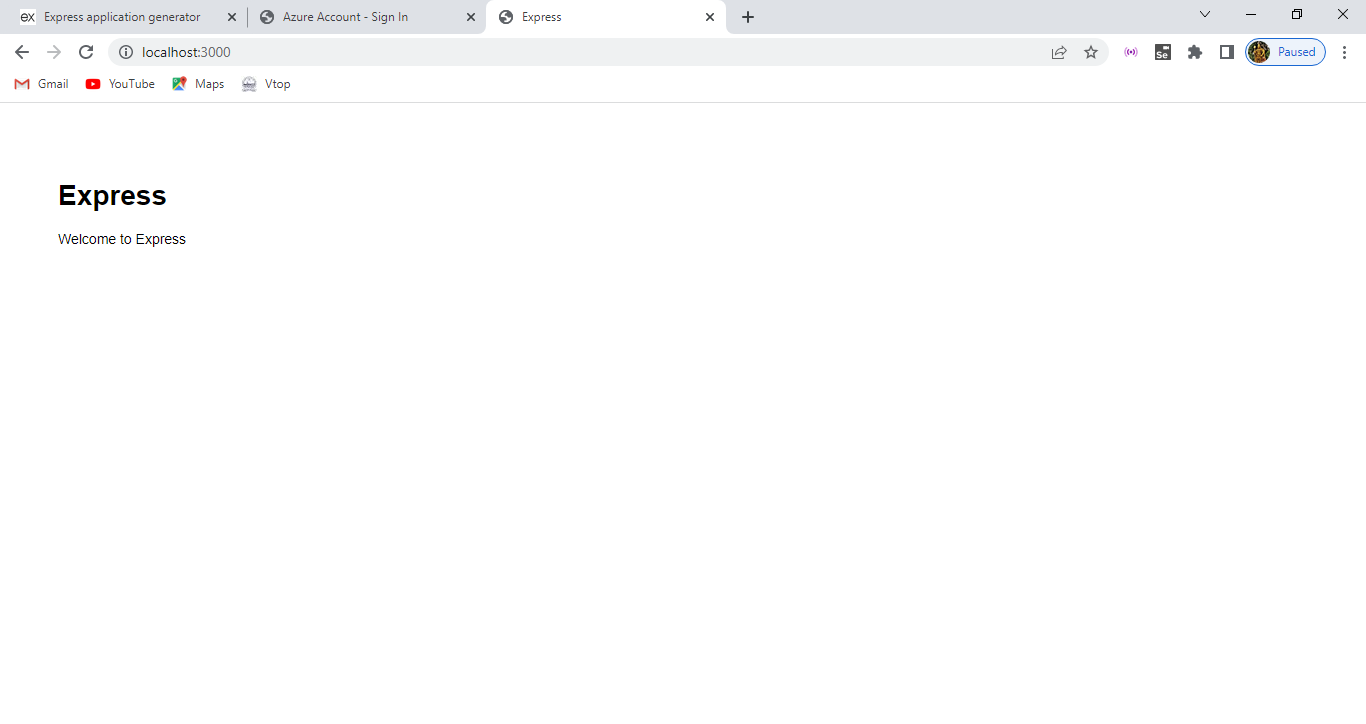
Give this command to start the local host page i.e sample page of it.

DEBUG=myexpressapp:\* npm start

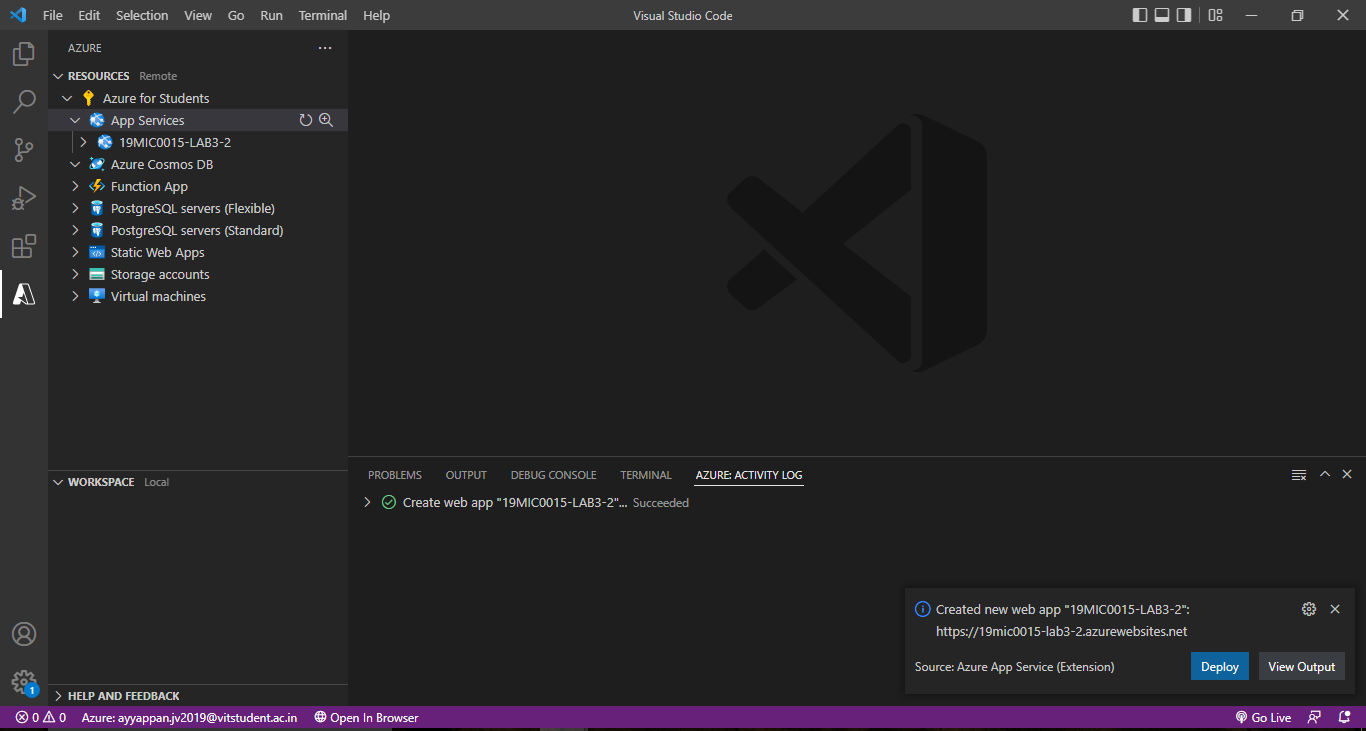


After installing those commands in the terminal don’t close, open any web browser and type this URL we get sample page of the web application using node js.

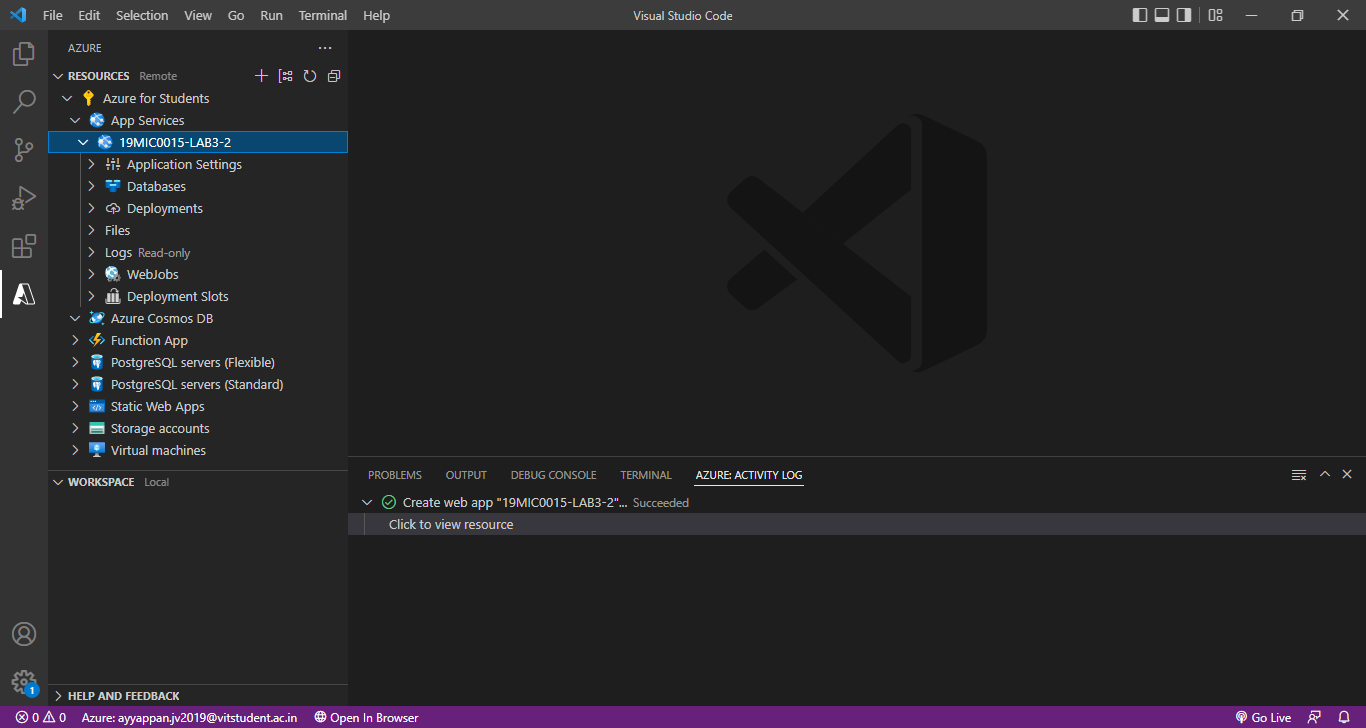
http://localhost:3000/



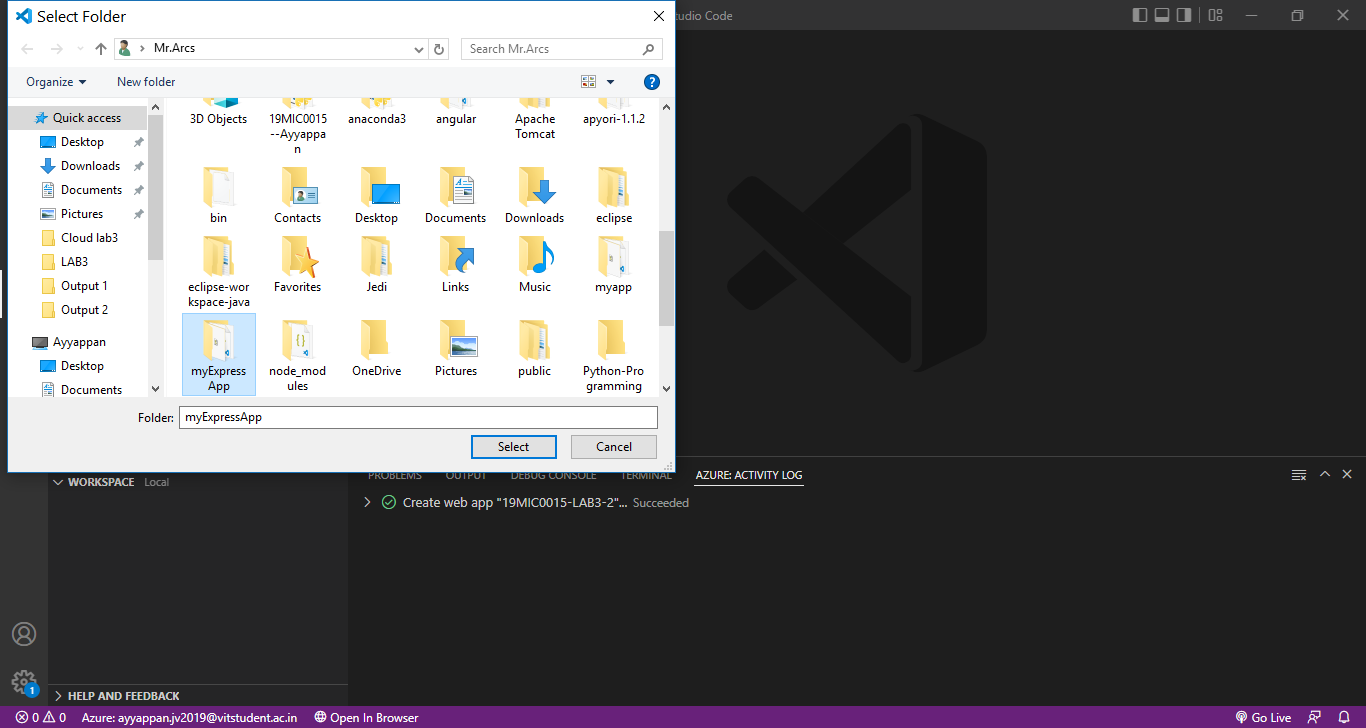
After this open VSC, we need to deploy the azure in node js so right click on app services enter those respective details and after that it create some deployment process to our local PC.



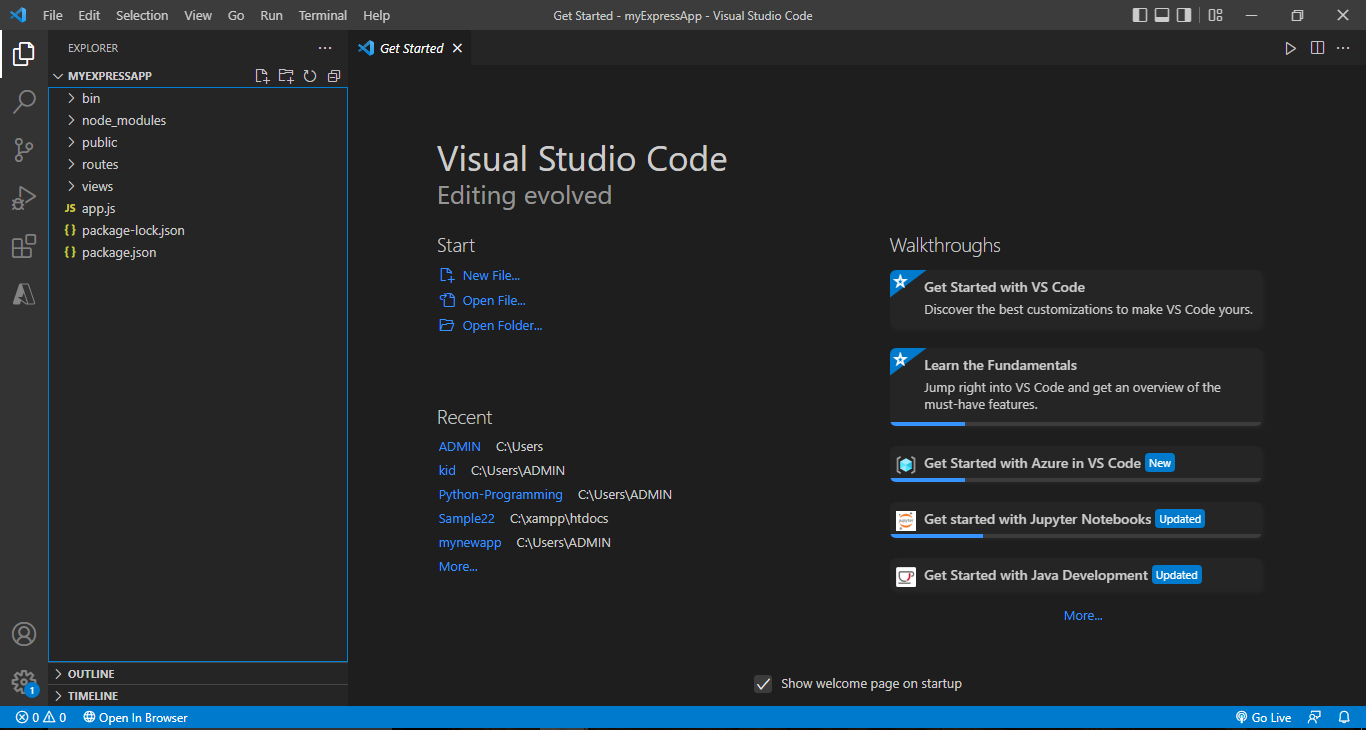
Here we can see successfully the deployment is created and at left corner our resource is created with 19MIC0015-LAB3-2 which is an azure web service for deploying the node js application.



Select those respective folder is created on command prompt that web application folder.



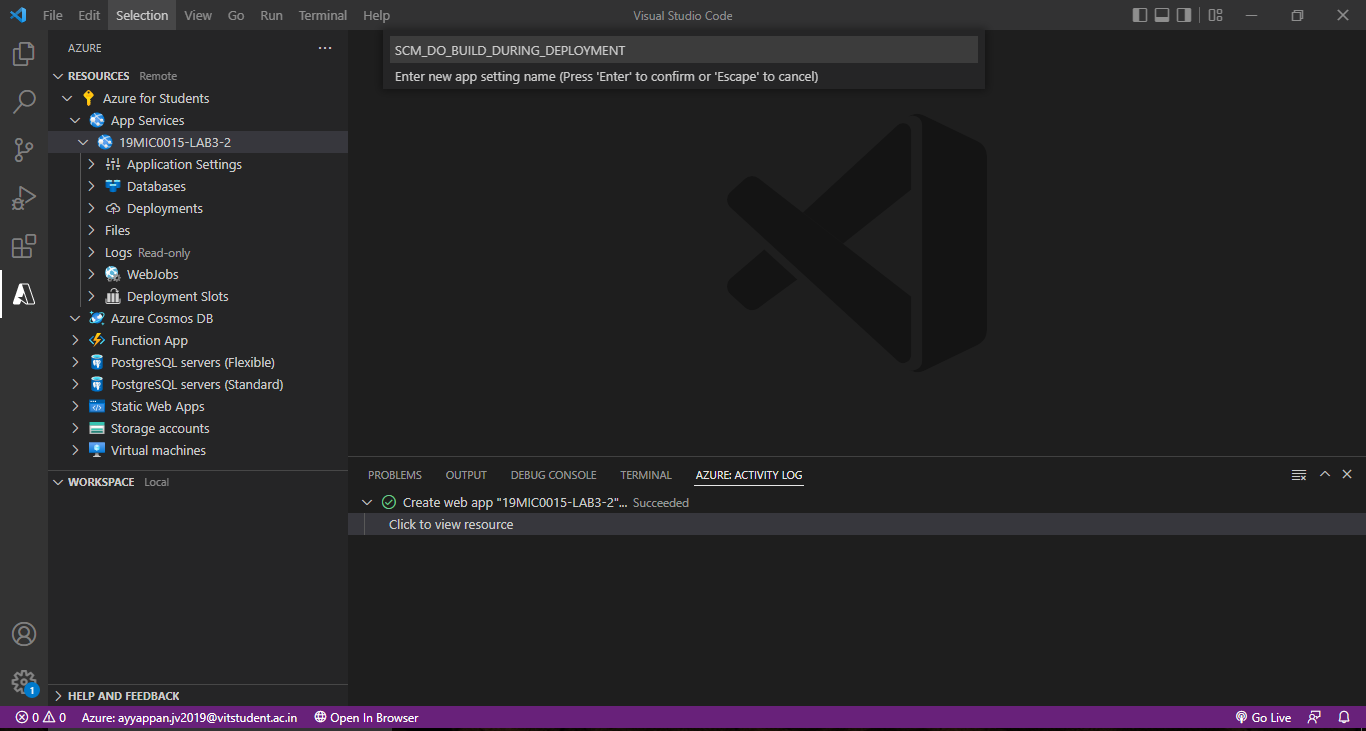
Here we got that folder of files which has node file of some sample webpage application.



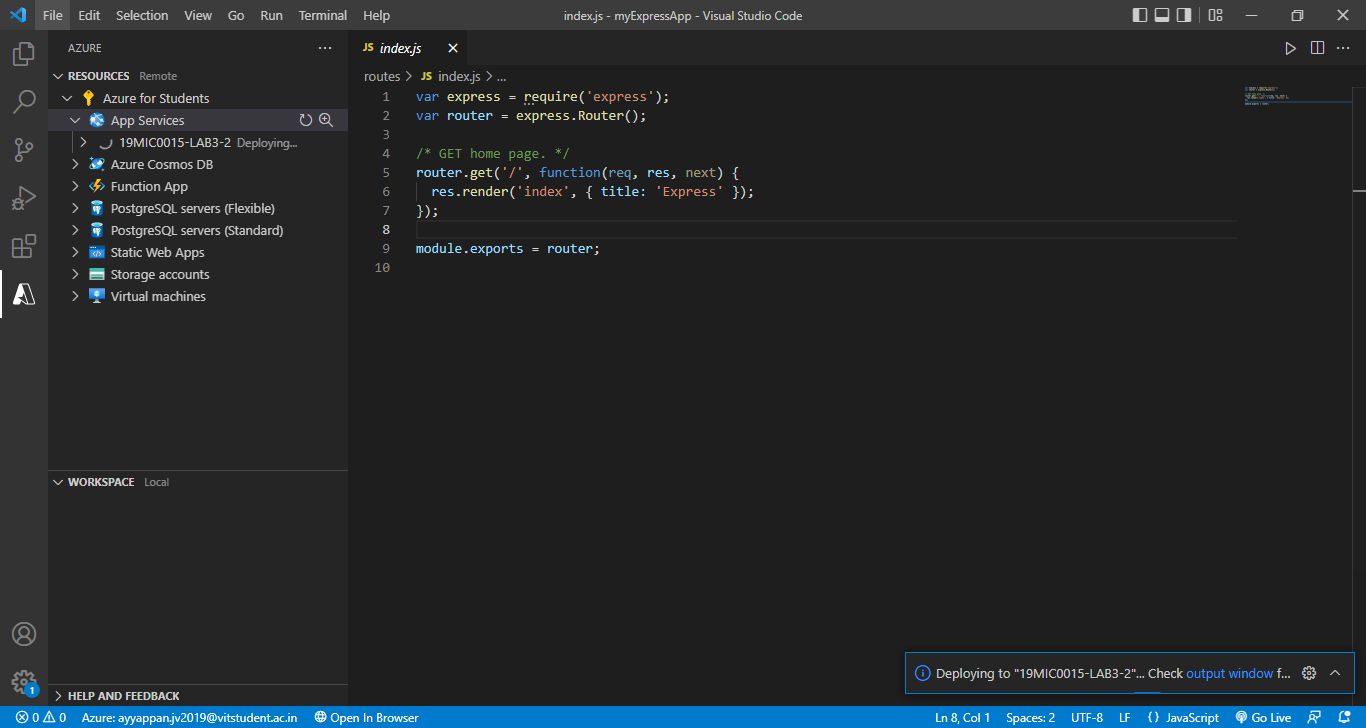
Now after this go to azure for student page in VSC, then in app service we have option called application settings select that

Type the command as SCM\_DO\_BUILD\_DURING\_DEPLOYMENT

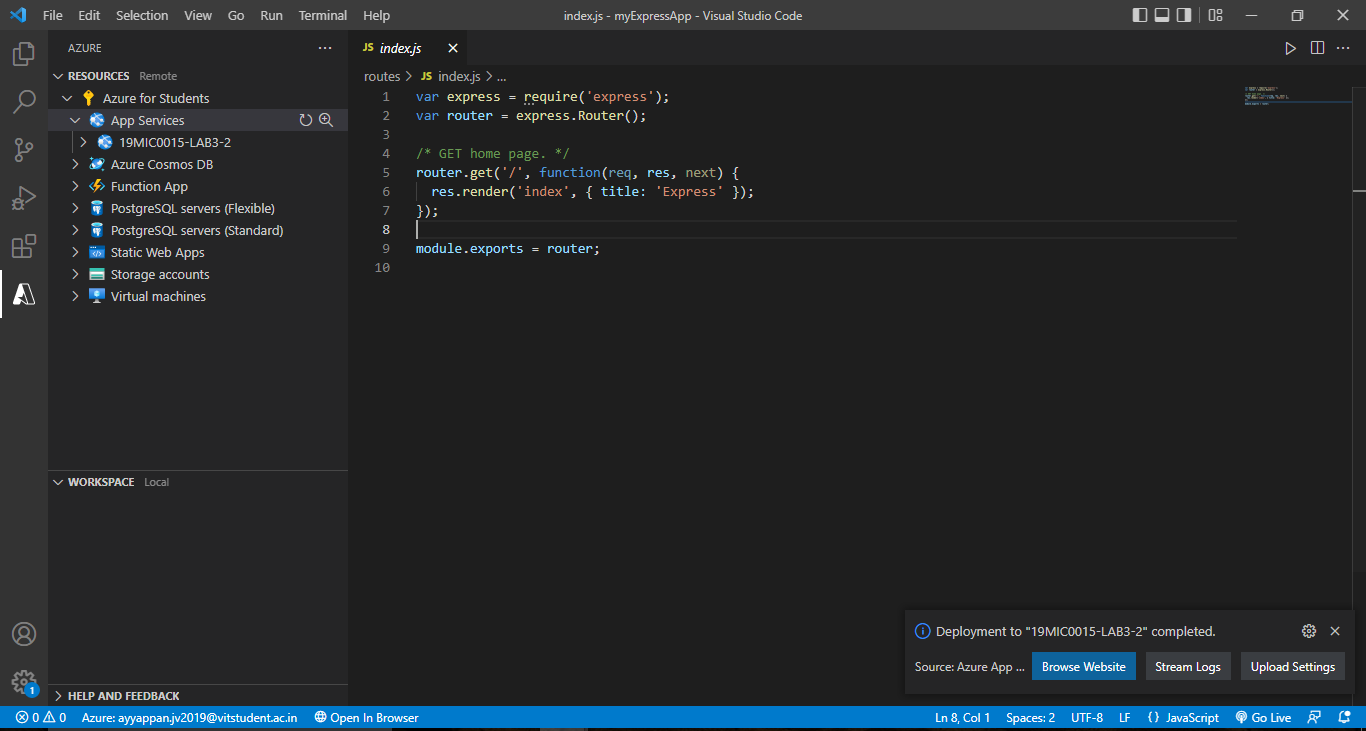
And Type true so this will enable build automation at deploy time which automatically detects the scripts and configure web.config with it.



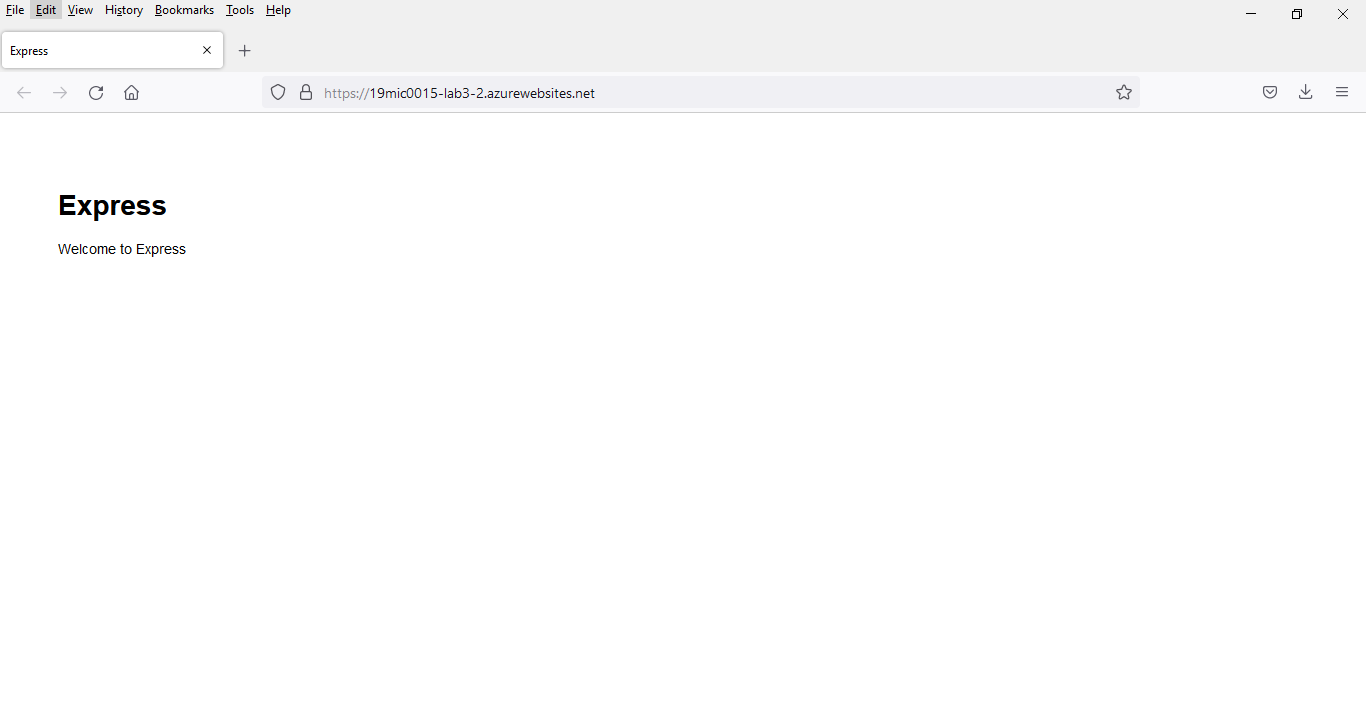
Then the deployment will setup and configure there and finally right click on app service which is created by us and select deploy the application.



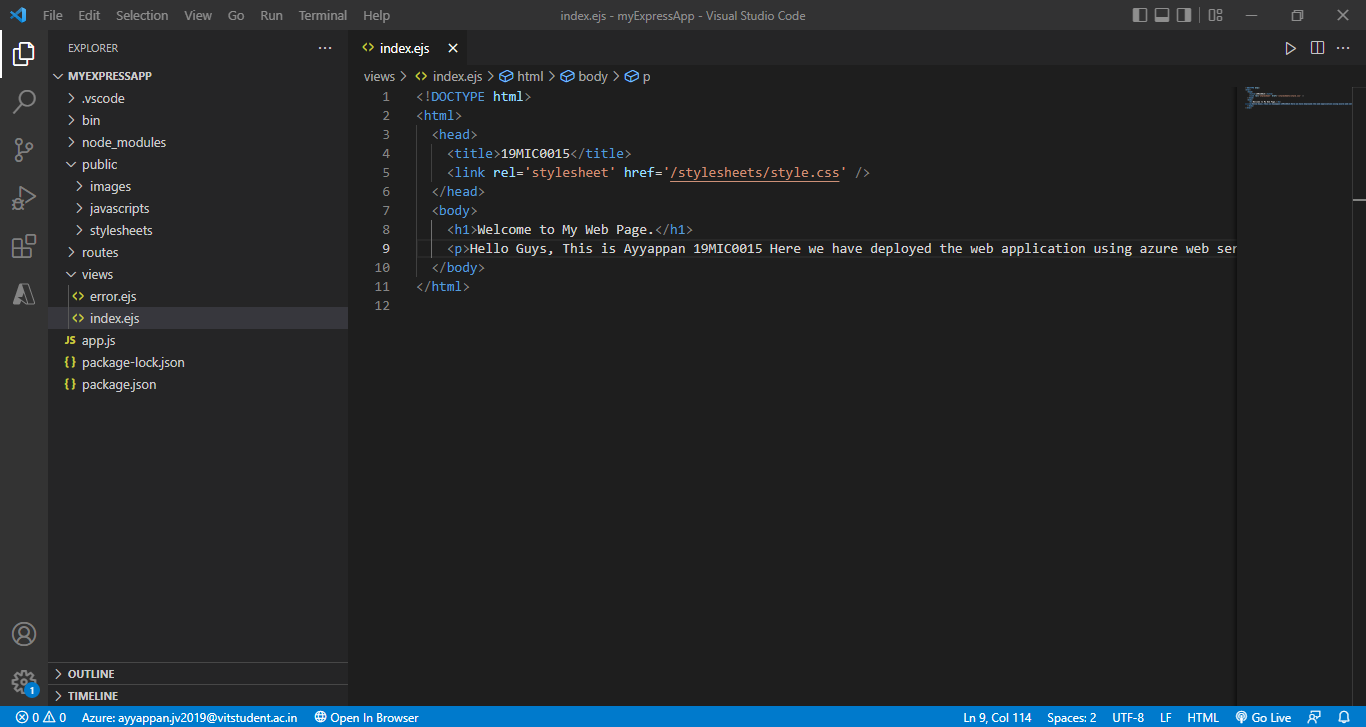
Then after some time if we open browser site and the sample page of node js web application deployed on azure web service in it.



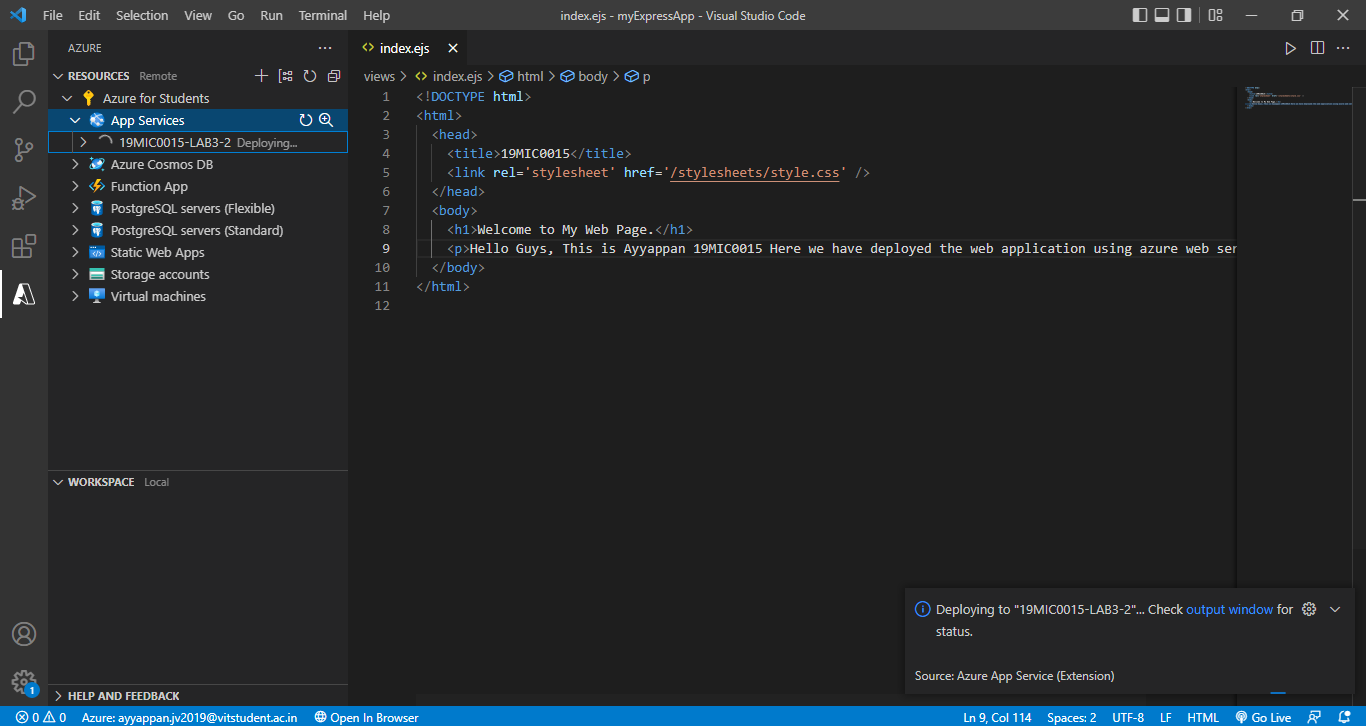
Here we go sample page of our web application which is node js file and it is deployed on the azure web service.



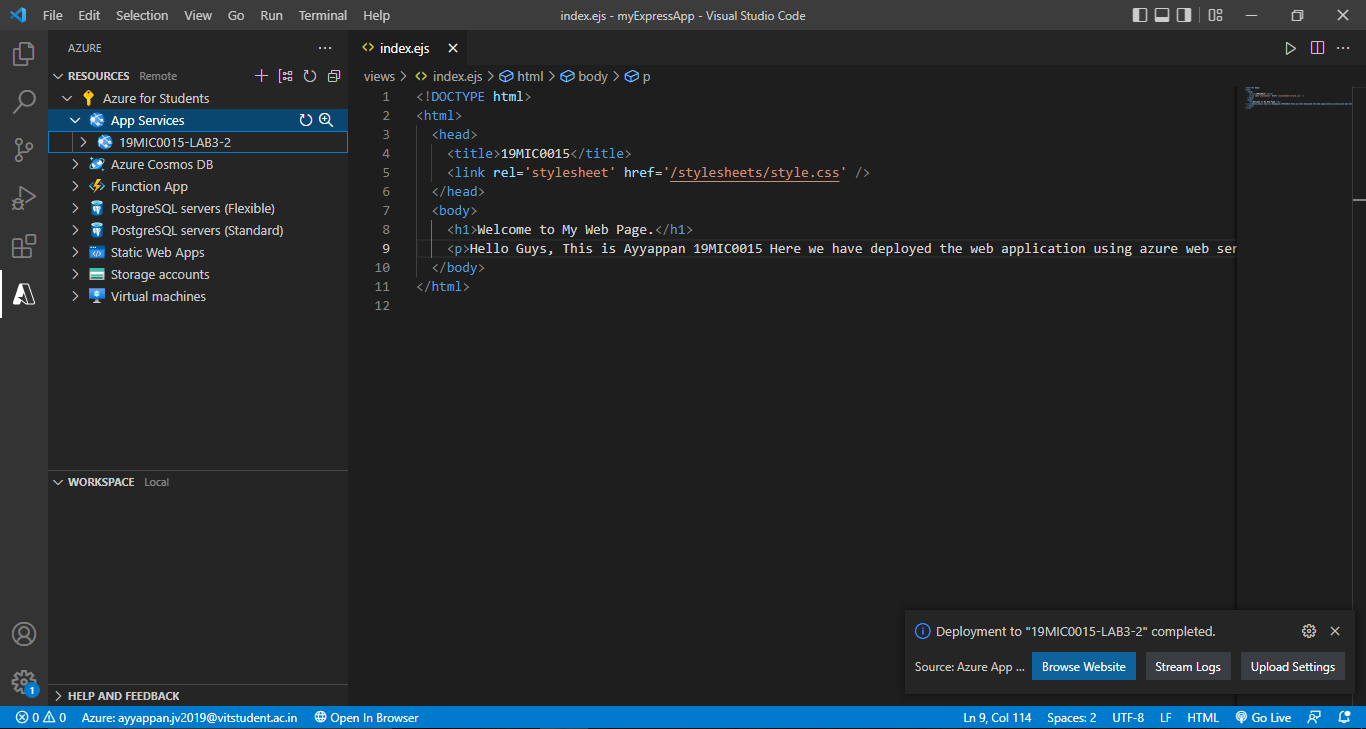
We have done redeploy here



We have edited the code with our details in index.ejs file After editing save it and again go to app service in azure content then select deploy application.

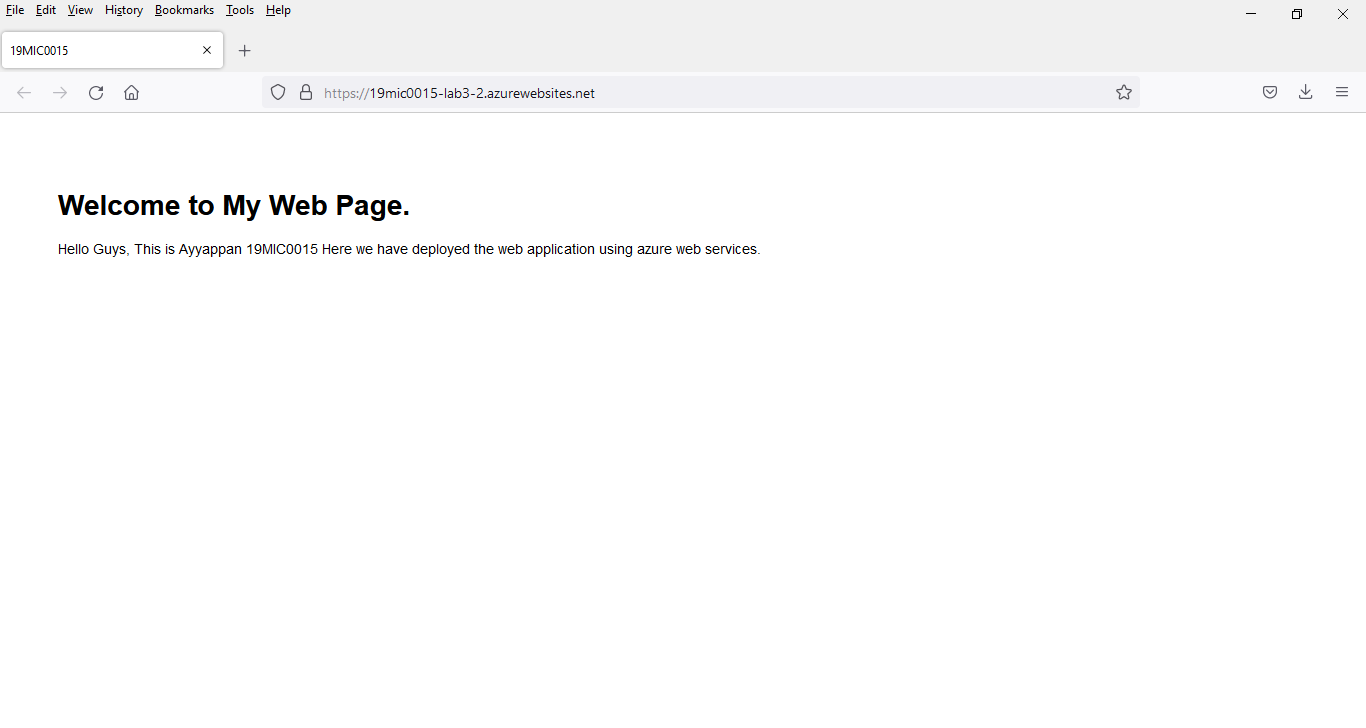


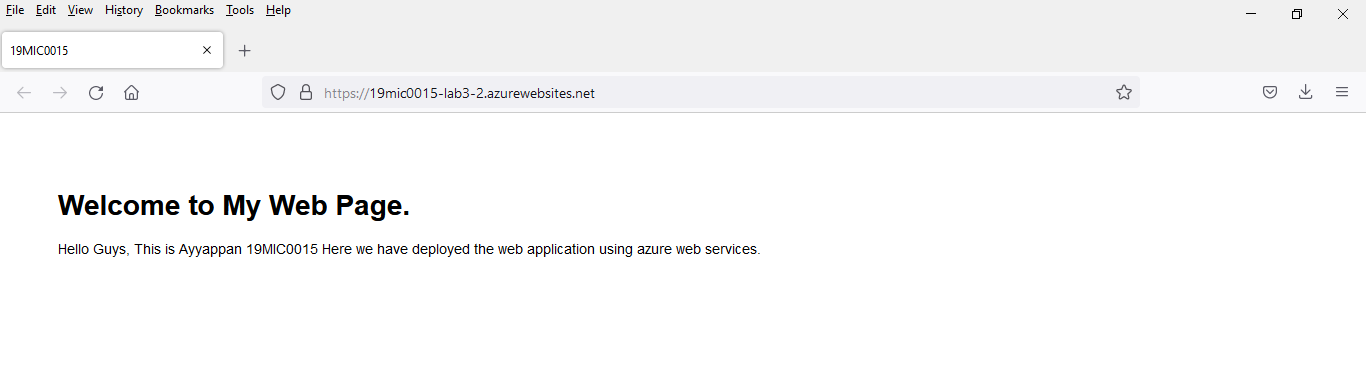
After some it get successfully deplaoyed.



Here finally we have created web application or web page using node JS on deploying on the azure web service portal.

And finally this our deployed web application.





Here this is stream logs of output where the message is indicating for that we are connected to the log streaming device in it.

