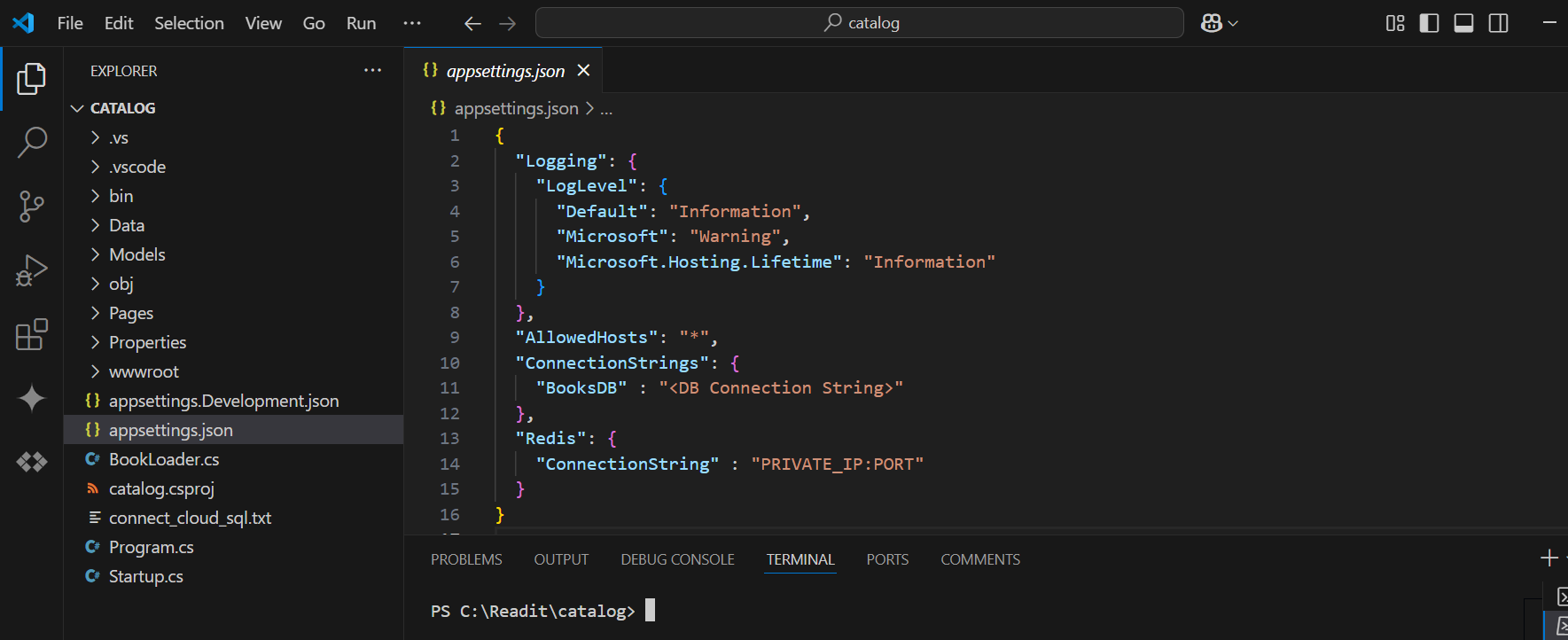
In this Lab we will publish our ReadIt app to the cat1 VM. We will also see options to login a VM using putty and ssh key.

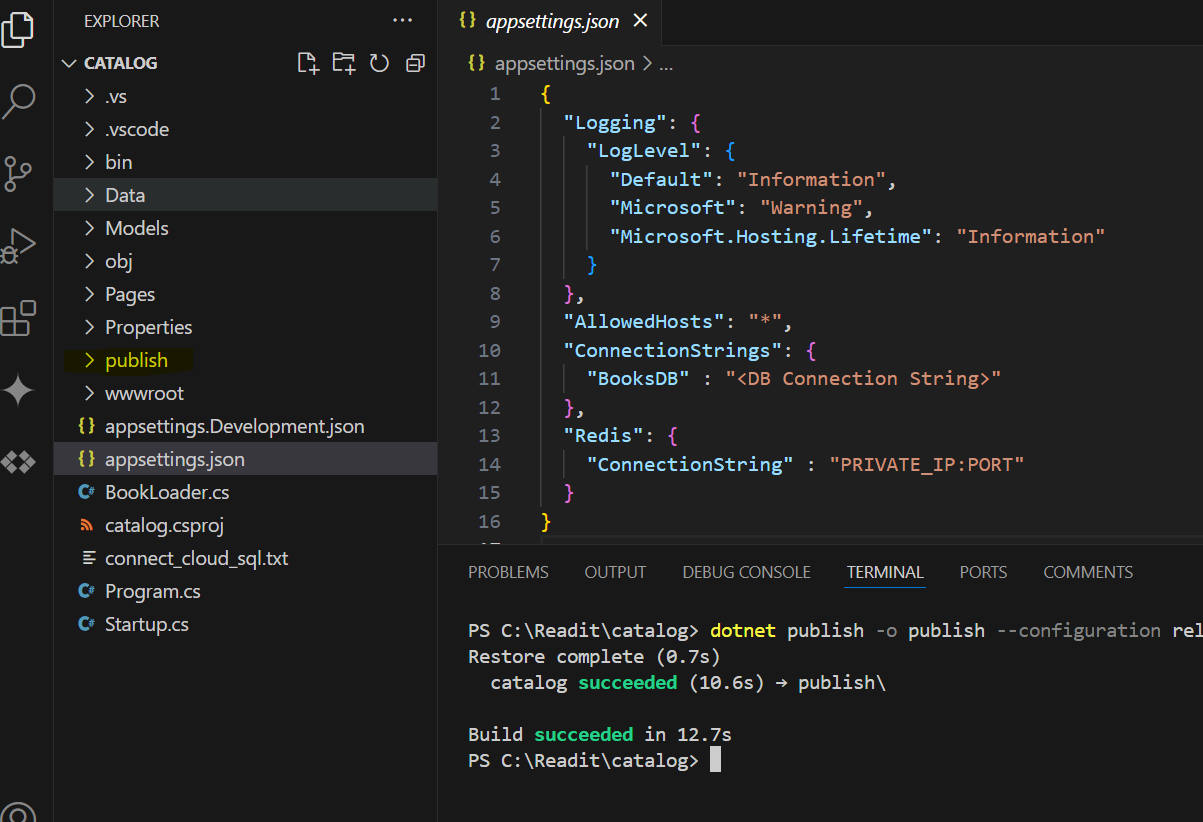
1. Start cat1 vm we created earlier from gcp console
2. Open VScode and open catalog folder there



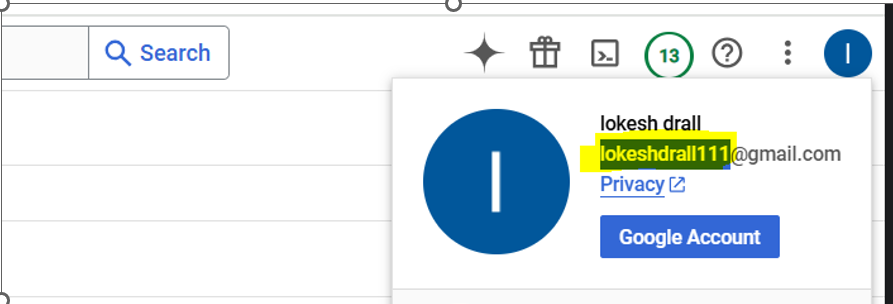
1. Go to Terminal>New terminal and run below command

dotnet publish -o publish --configuration release

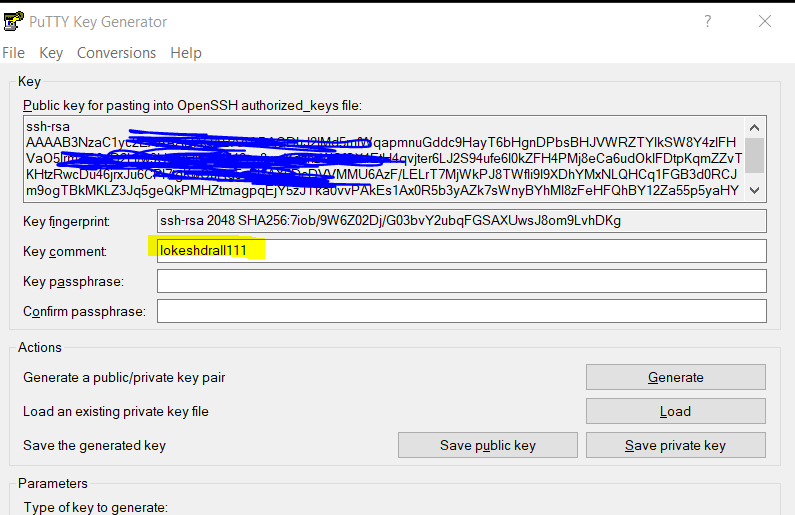
Once completed, must be able to see new folder “Publish”



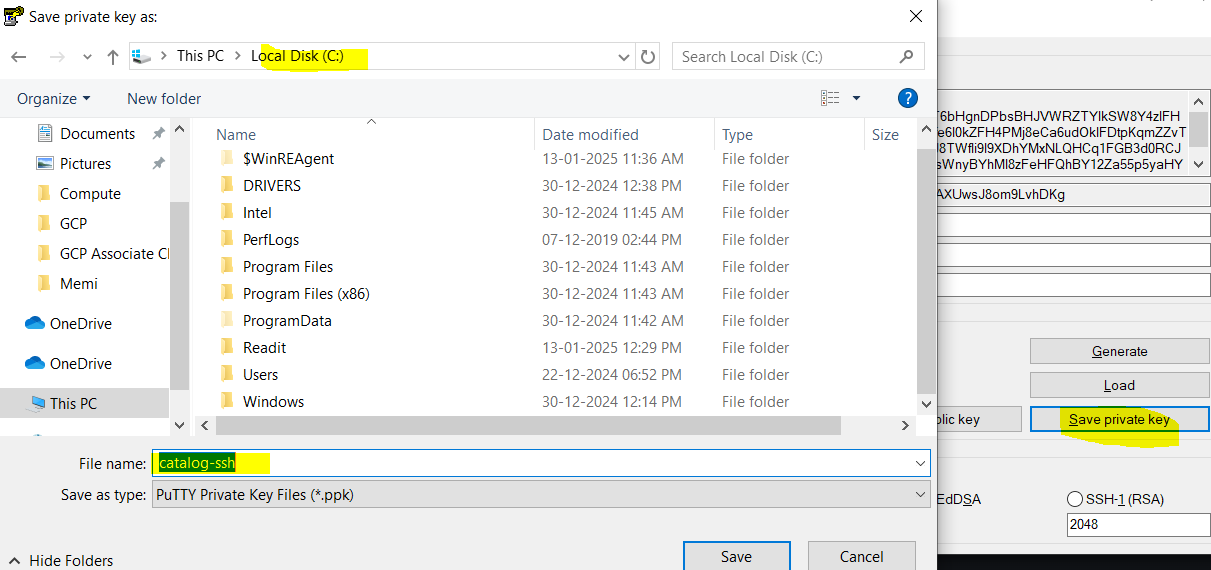
1. Download and install Putty tool <https://www.putty.org/>
2. Open Puttygen and click generate and keep the mouse moving until it completes generating the key
3. Now go to GCP console and copy your username (without email suffix)



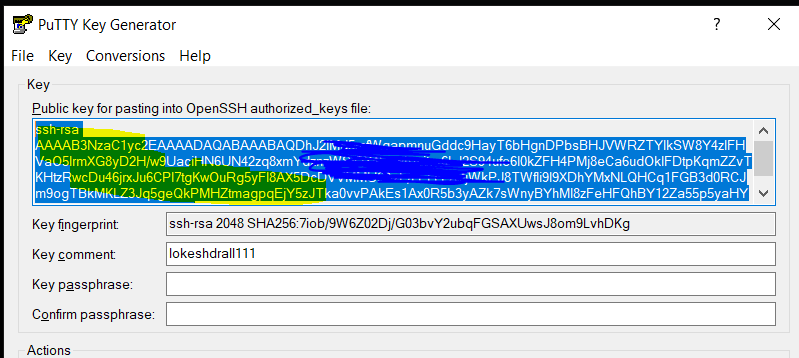
1. Go to PuttyGen, replace “key comment” with your username



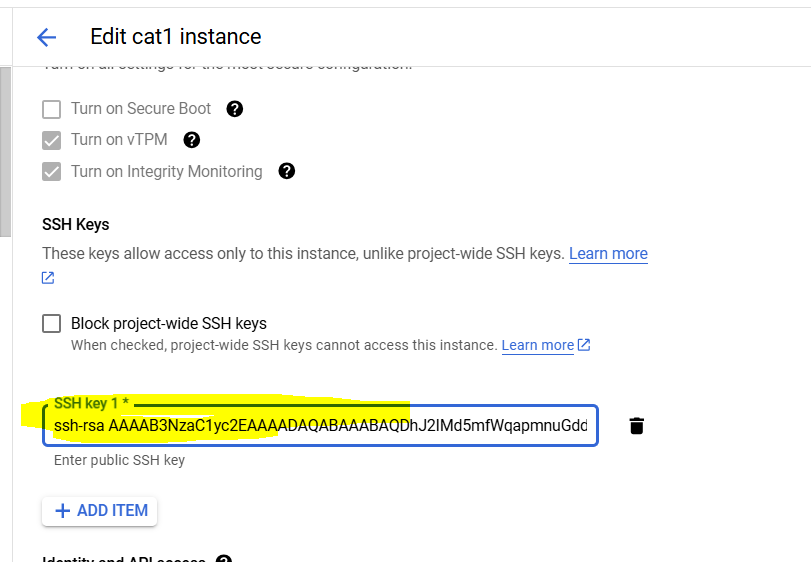
1. Click Save private key and save it to C drive with name “catalog-ssh” (If see any error sving it to c drive, then first save it to somewhere else and then copy to c drive)



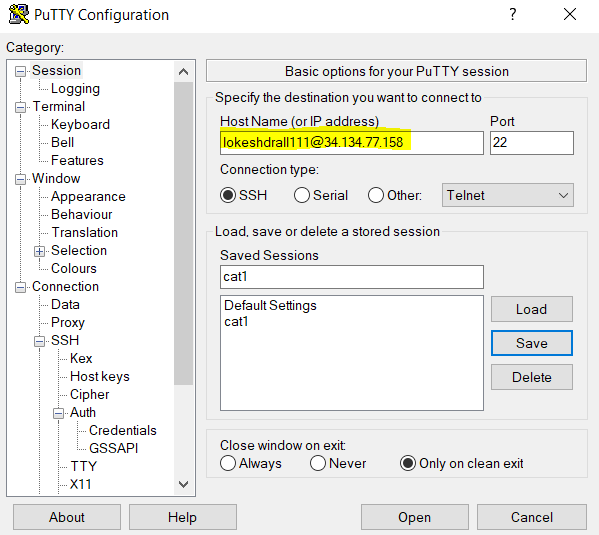
1. Copy the key from puttygen



1. Go to GCP console, open the cat1 vm, click edit, scroll down to SSH Keys, click add item and paste the key

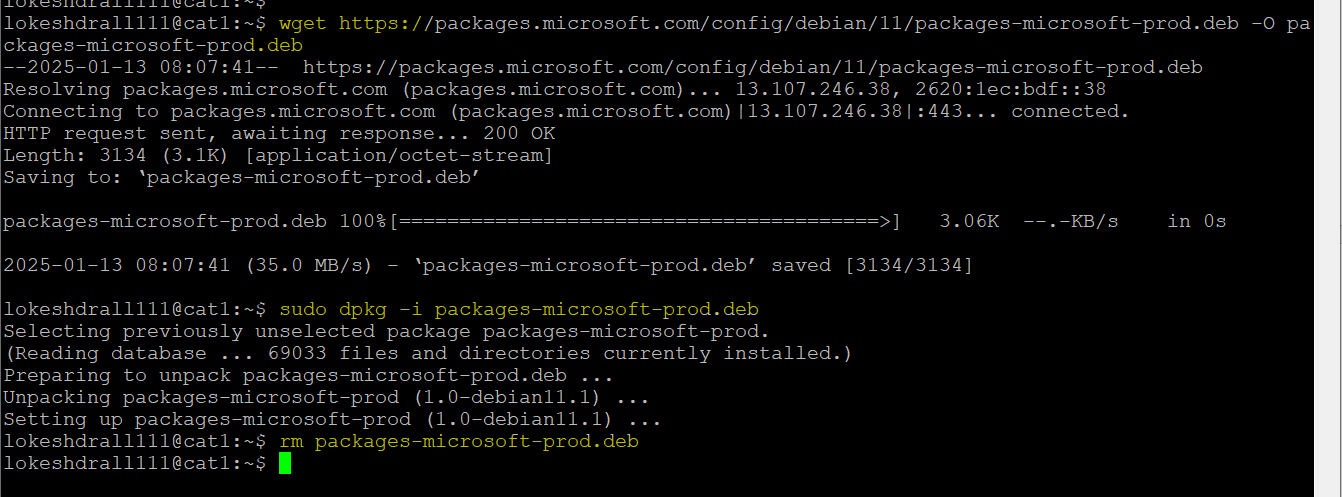


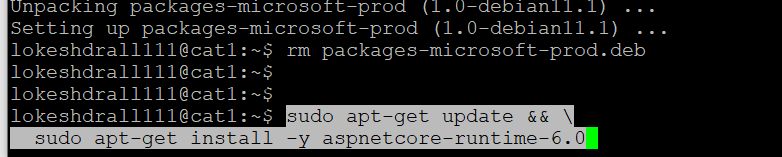
1. Connect to cat1 vm via putty, open putty, enter your username@externalip as hostname, add the private key location under ssh>auth>credentials, also optionally can save it for future use and click open



1. Run below commands one by one on the cat1 vm:-
   1. wget [https://packages.microsoft.com/config/debian/11/packages-microsoft-prod.deb -O packages-microsoft-prod.deb](https://packages.microsoft.com/config/debian/11/packages-microsoft-prod.deb%20-O%20packages-microsoft-prod.deb)
   2. sudo dpkg -i packages-microsoft-prod.deb
   3. sudo dpkg -i packages-microsoft-prod.deb
   4. sudo apt-get update && \

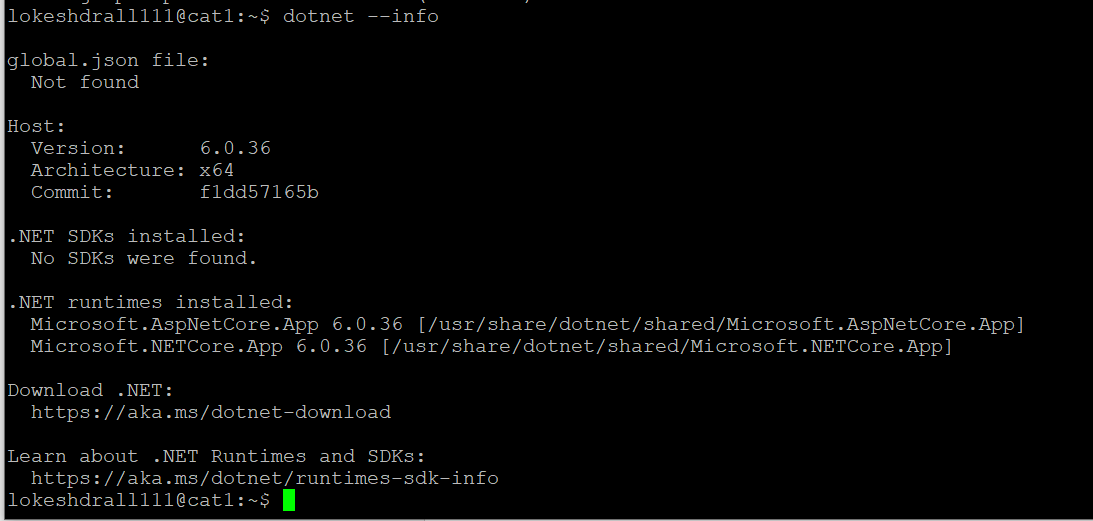
sudo apt-get install -y aspnetcore-runtime-8.0



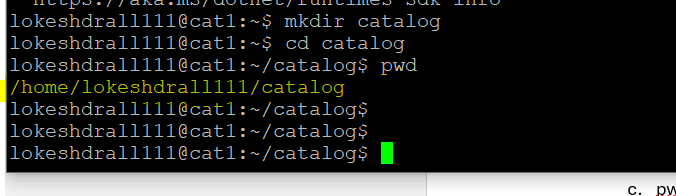


1. To ensure dotnet is installed run below, it should give dotnet details

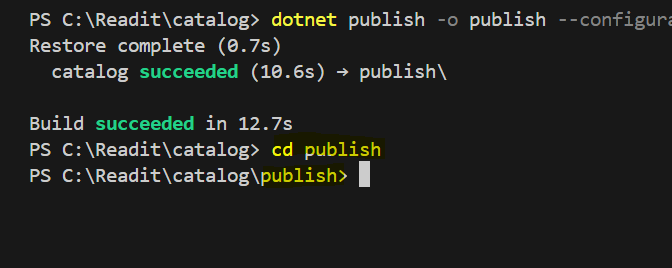
dotnet –info



1. Run below commands
   1. mkdir catalog
   2. cd catalog
   3. pwd
2. copy the whole path “/home/lokeshdrall111/catalog”

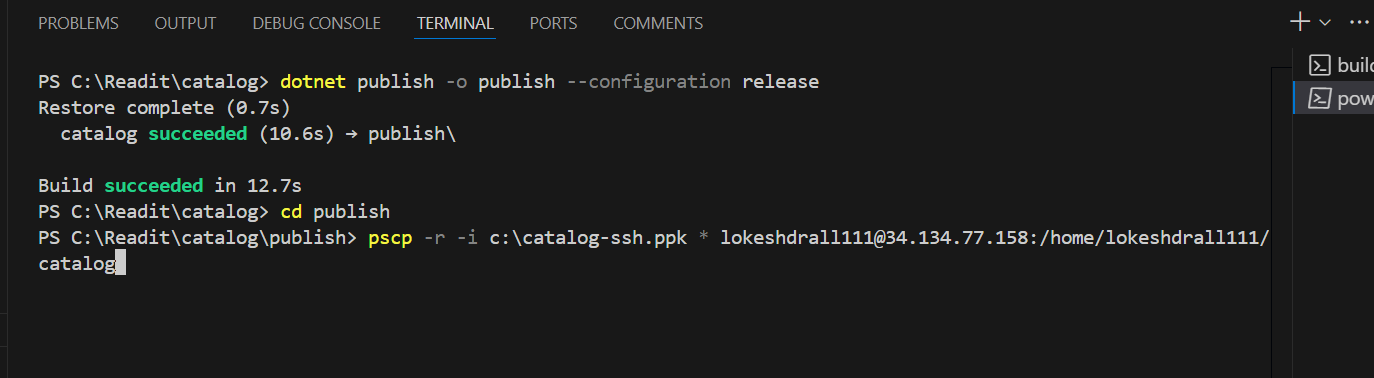


1. Go to vscode, go to publish folder by typing cd publish



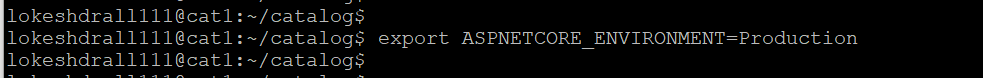
1. Now to upload our code files to vm, run below command in VS (Note if see error like pscp not recognized… then simply close and open vscode again)

pscp -r -i c:\catalog-ssh.ppk \* {USERNAME}@{IP\_ADDRESS}:/home/{USERNAME}/catalog

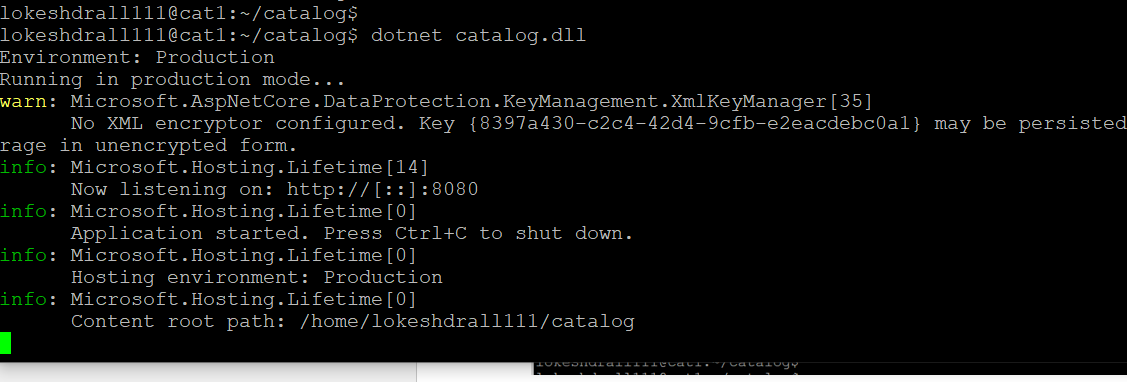


1. Run below command in VM (putty)

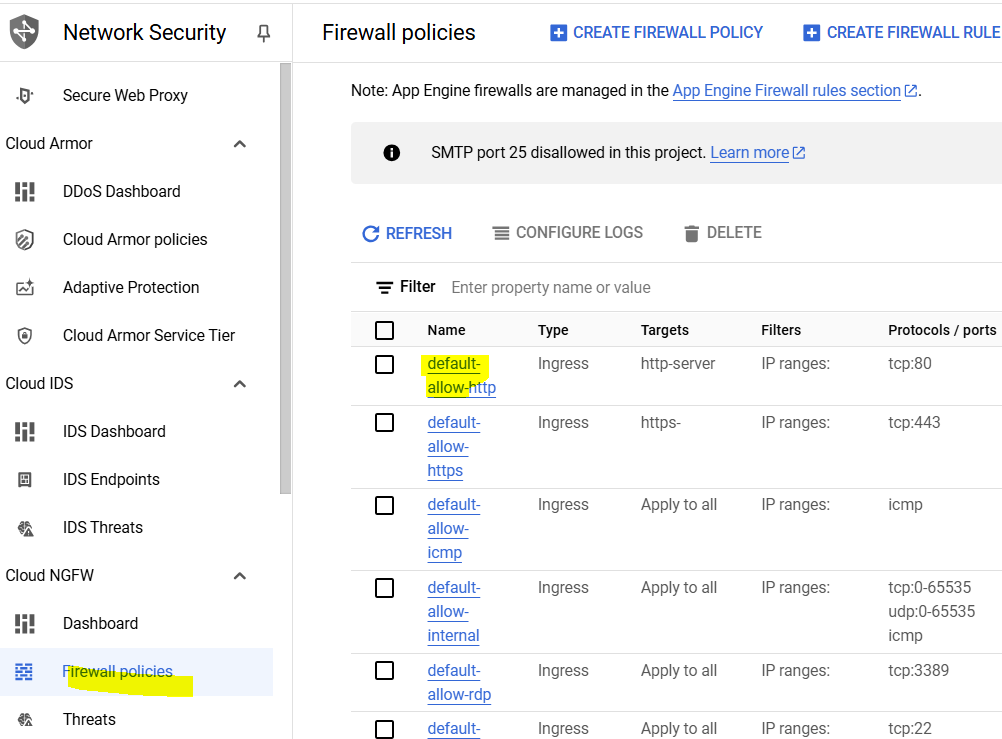
export ASPNETCORE\_ENVIRONMENT=Production



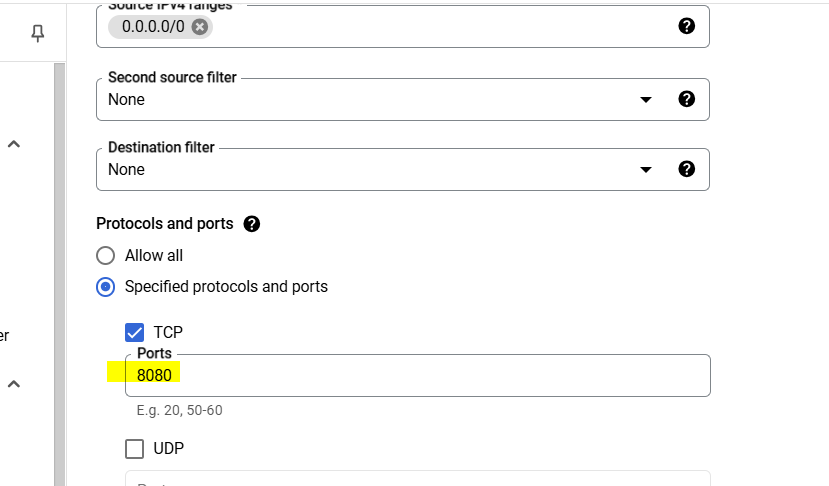
1. Run dotnet catalog.dll it should show app running



1. Now we ned to open port 8080 at GCP firewall to access this via web, for that go to gcp, search vpc network and open default-allow-http rule

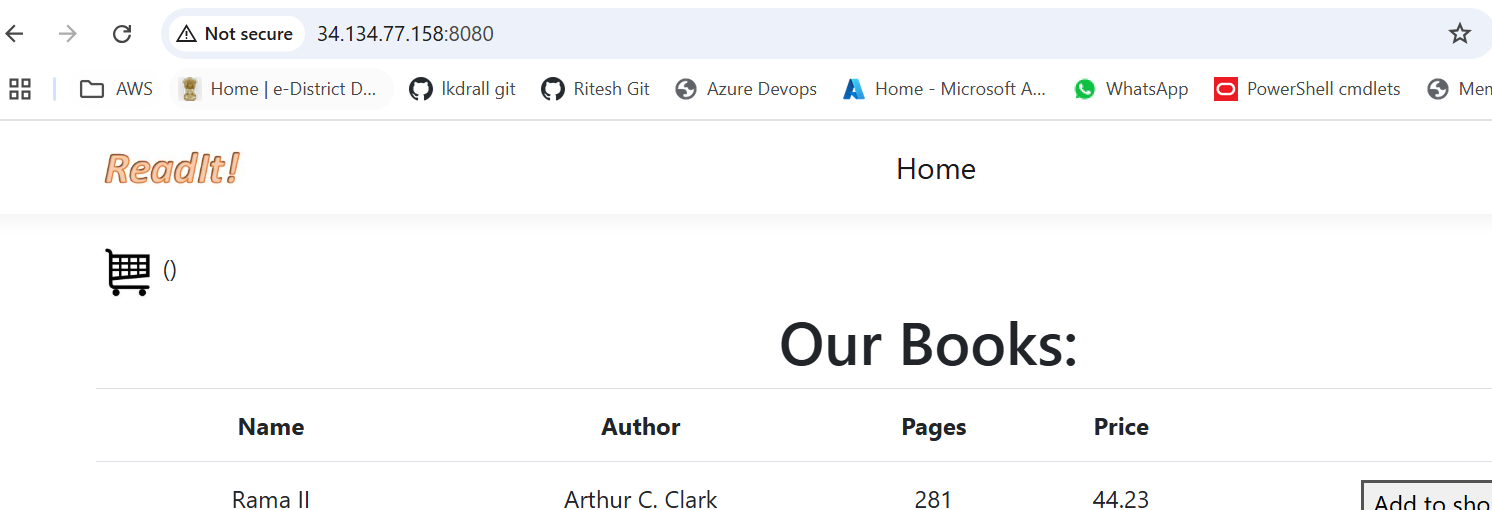


1. Click edit, and change the port number to 8080 (initially it was 80), then click save

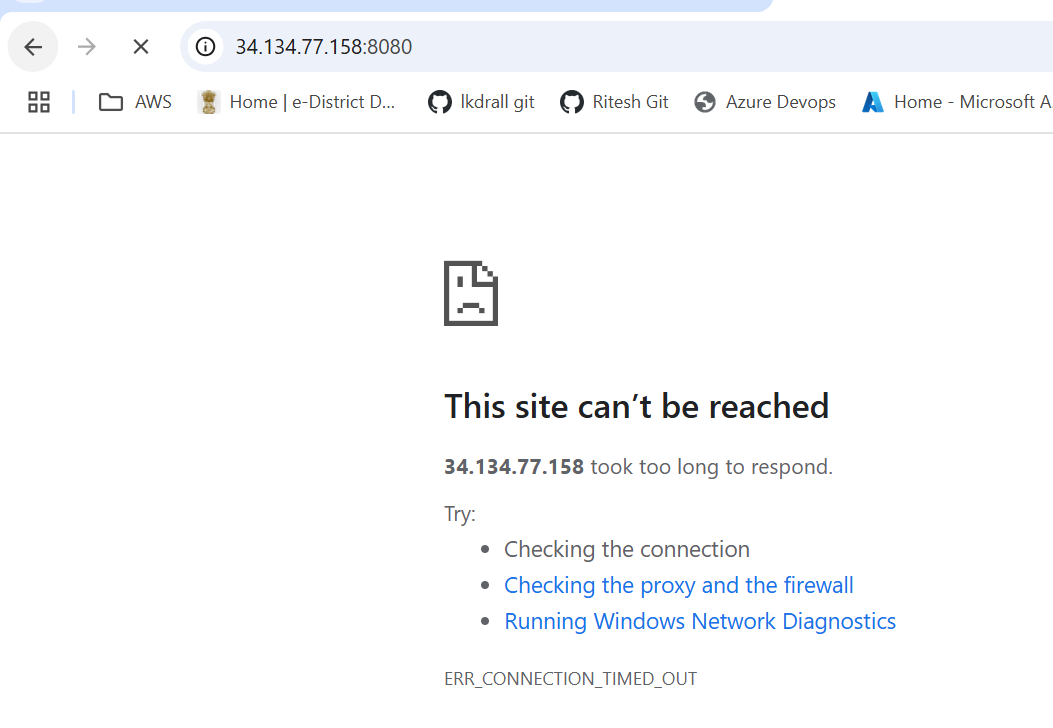


1. Copy external IP of vm, paste it in new tab of browser and add port 8080, 34.134.77.158:8080

And open it, this should show the readit app.



1. Now if we close the putty screen, and then reload page then it will not work



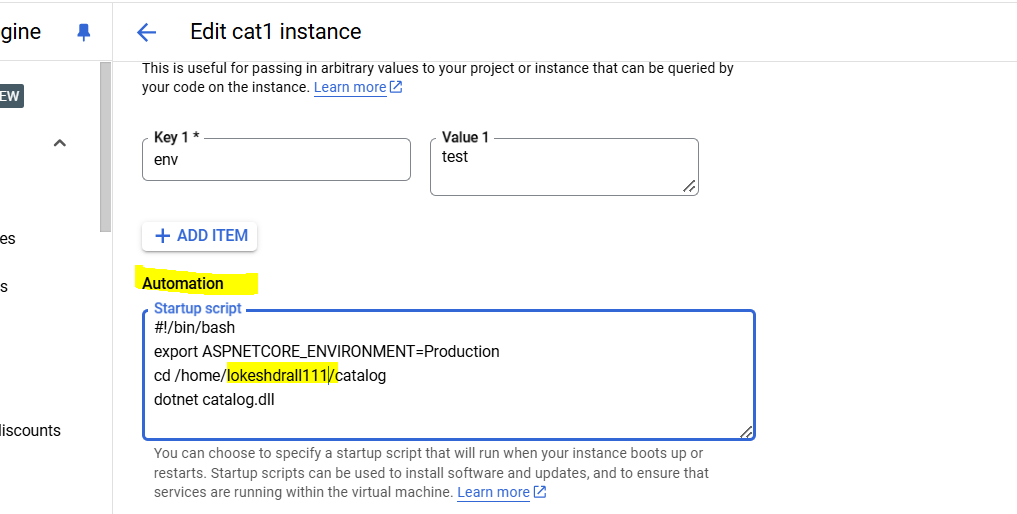
1. Now go back to GCP console, open vm and click edit, scroll down to Automation and paste below script there, make sure to update the username, click save

#!/bin/bash

export ASPNETCORE\_ENVIRONMENT=Production

cd /home/{USERNAME}/catalog

dotnet catalog.dll



1. Now stop and start the vm, after starting try accessing web app again, it should work now. (After restart external Ip may change, so make sure to check that)

