Build and Deploy an Machine Learning model using AWS and APIs.

Steps:

1. Build a model on your local box (Amazon Fine Food reviews) and store the model and other key model related variables in .pkl files

2. Launch a micro instance on AWS.

3. Connect to the AWS box [ssh]

4. Move the files to an AWS EC2 instance/box [scp]

5. Install all packages needed on the AWS box.

6. Run app.py on the AWS box.

7. Check the output in the browser.

Software needed:

1. Anaconda:

a. Windows 64 bit: https://repo.continuum.io/archive/Anaconda3-5.2.0-Windows-x86_64.exe

b. Windows 32 bit: https://repo.continuum.io/archive/Anaconda3-5.2.0-Windows-x86.exe

c. Mac : https://repo.continuum.io/archive/Anaconda3-5.2.0-MacOSX-x86_64.sh

d. Linux 64 bit: https://repo.continuum.io/archive/Anaconda3-5.2.0-Linux-x86_64.sh

e. Linux 32 bit: https://repo.continuum.io/archive/Anaconda3-5.2.0-Linux-x86.sh

f. Check the previous Archives of Anaconda: https://repo.continuum.io/archive/

2. Packages needed:

1. pip3

2. pandas

3. numpy

4. sklearn

7. flask

you can copy all these packages and try like this:
https://stackoverflow.com/a/15593865/4084039

[1] Code on local box

Anaconda Prompt:

1. Change to the code directory.

2. Run “python3 app.py”

3. Browser: http://localhost:8080/index

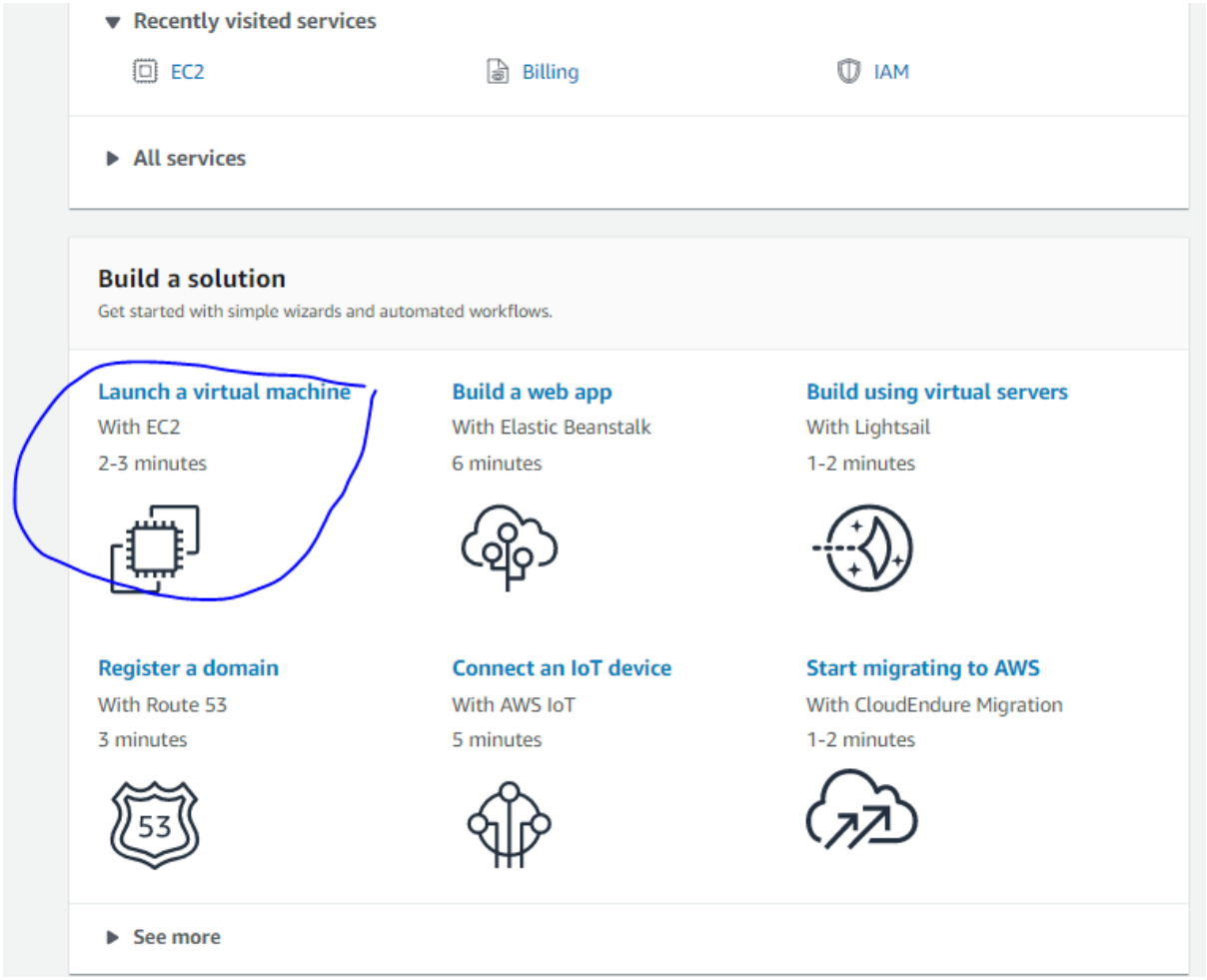
[2] Launch a micro instance on AWS

Creating an instance:

1. Create an AWS account https://aws.amazon.com,
https://portal.aws.amazon.com/billing/signup#/start

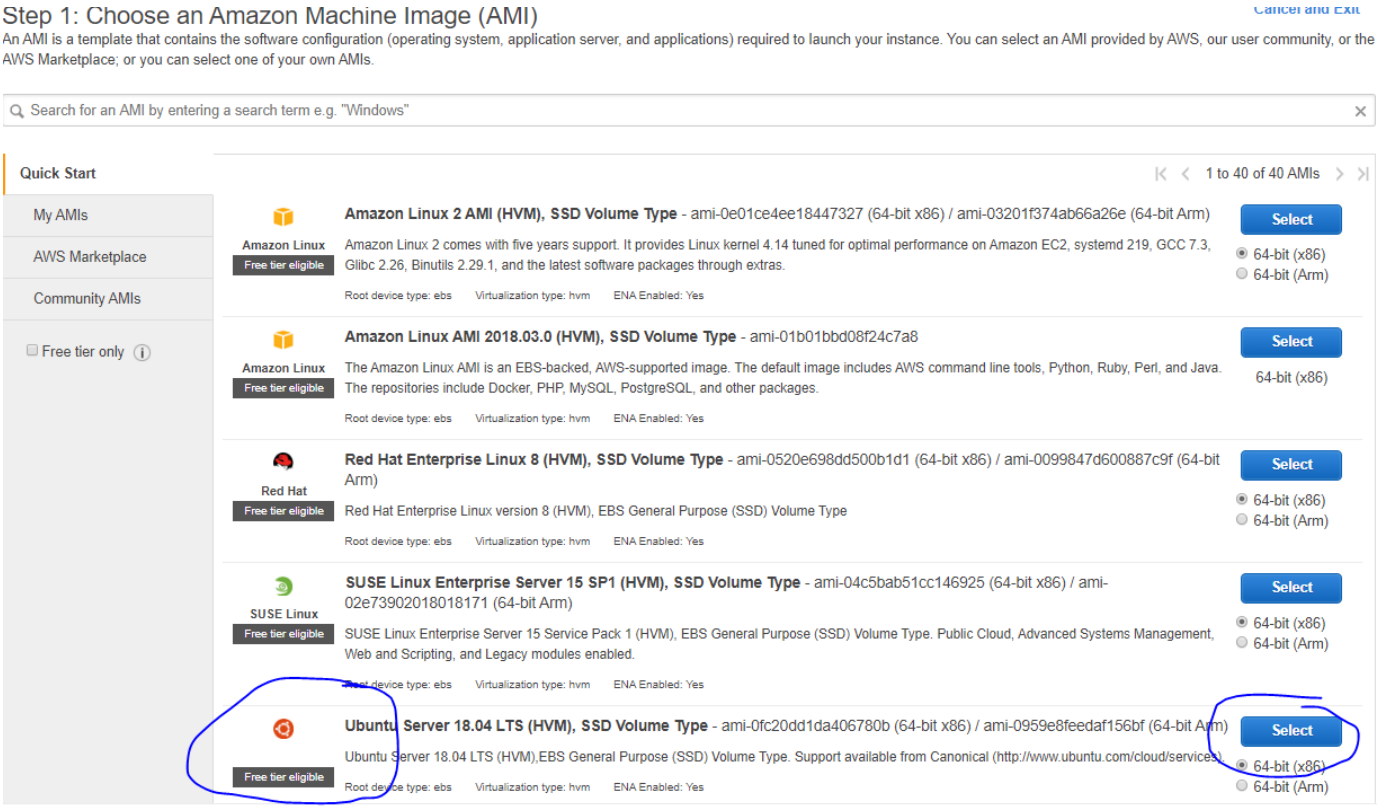
2. Login: https://console.aws.amazon.com

After login:



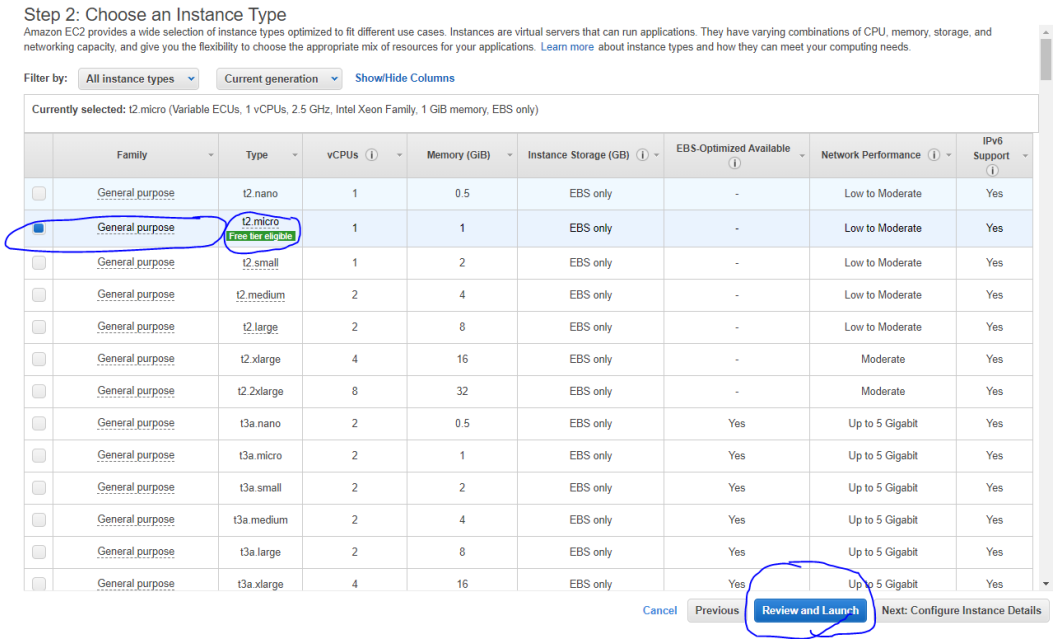
Launch the EC2 instance

3. Choose the ubuntu free tire

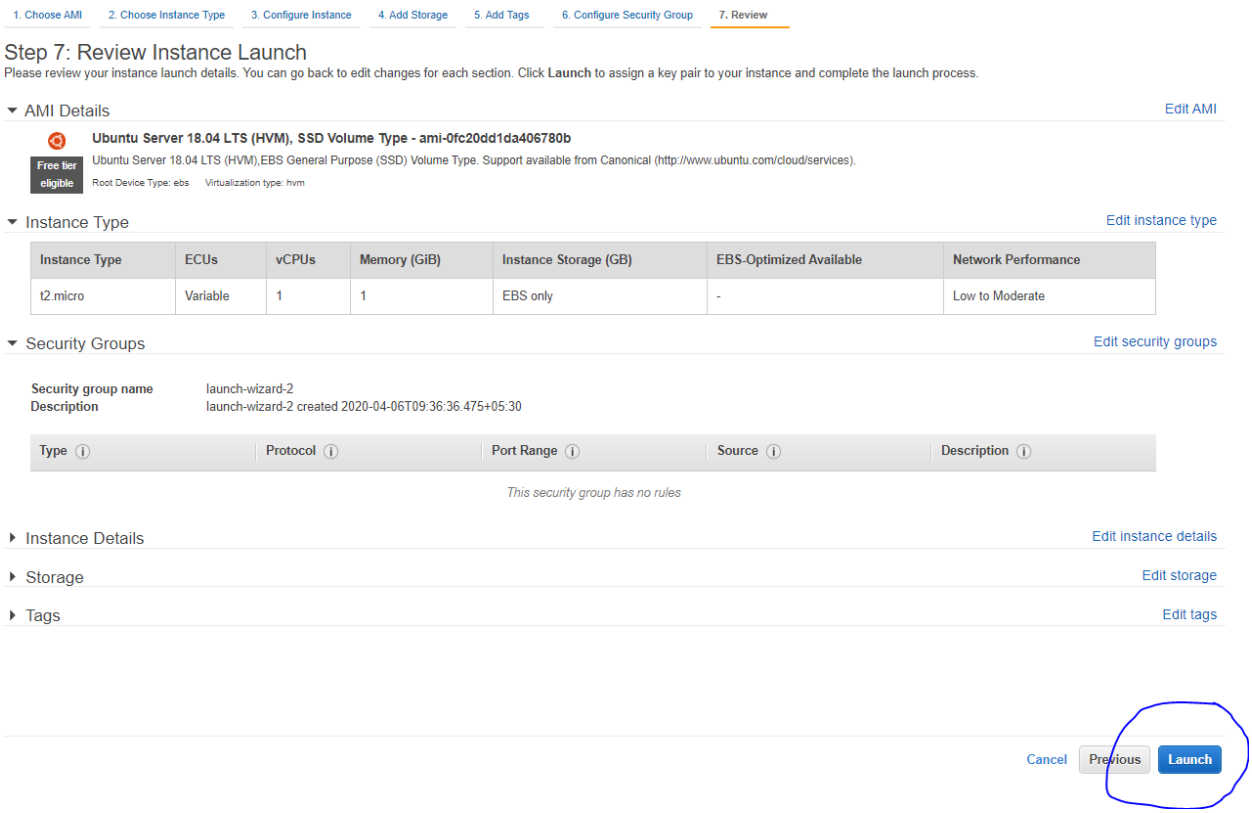


Click on select

4. Choose t2.micro free tier eligible

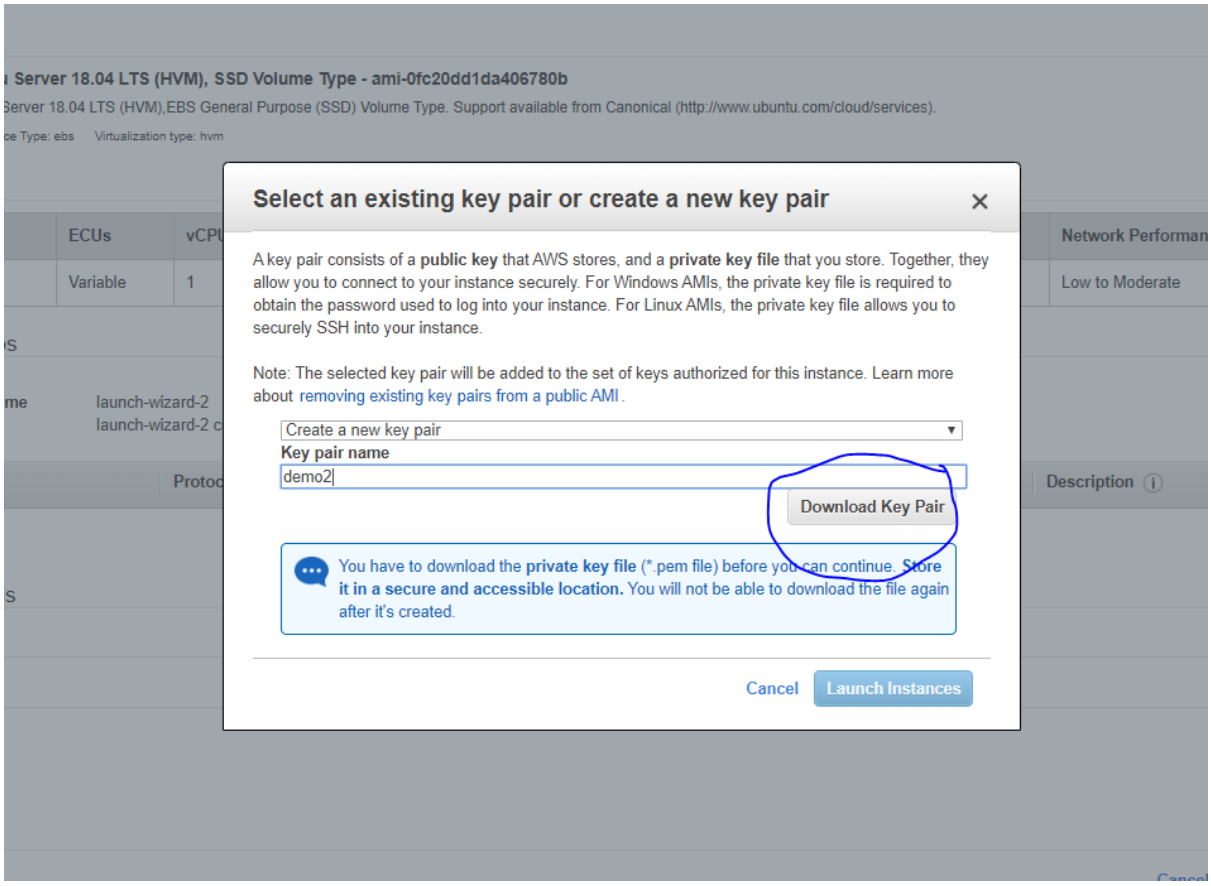


Click on review and launch



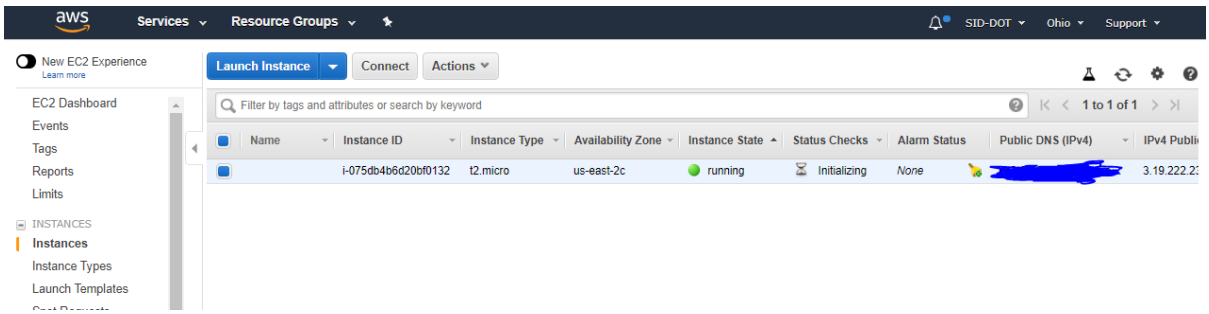
Click on launch

6.



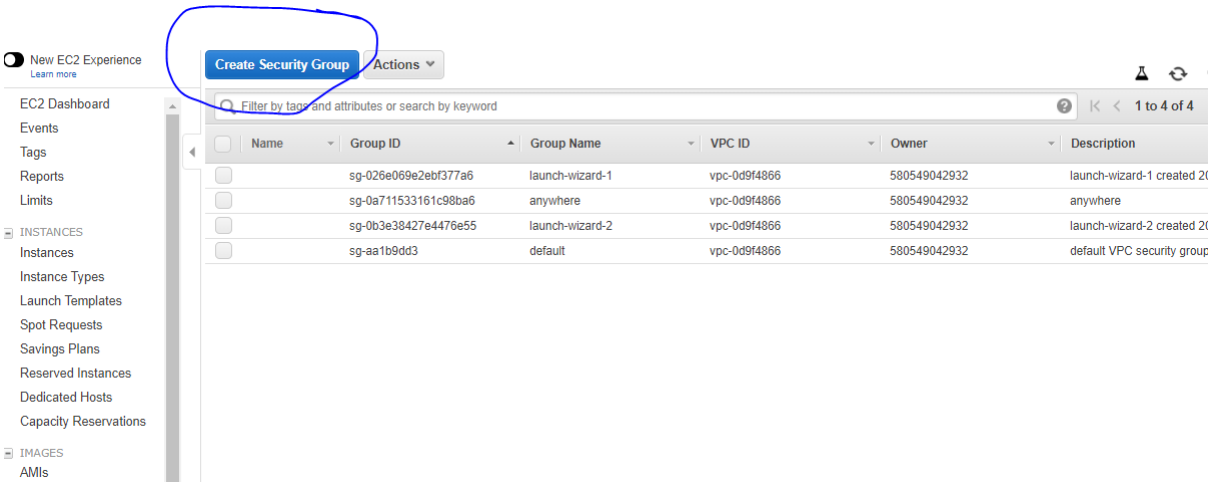
Click on “Download Key Pair” and save the .pem file then click on “Launch Instance”

7.

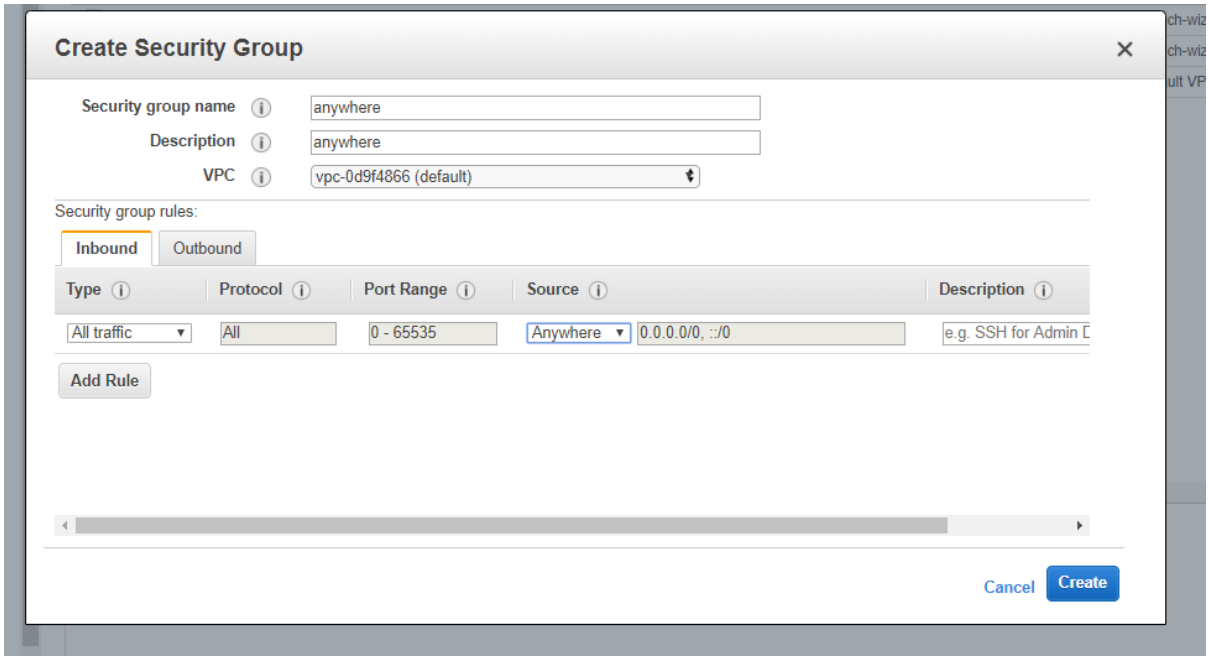


You will see this screen, you have successfully launched the an EC2 instance, now we need to launch an flask api in it

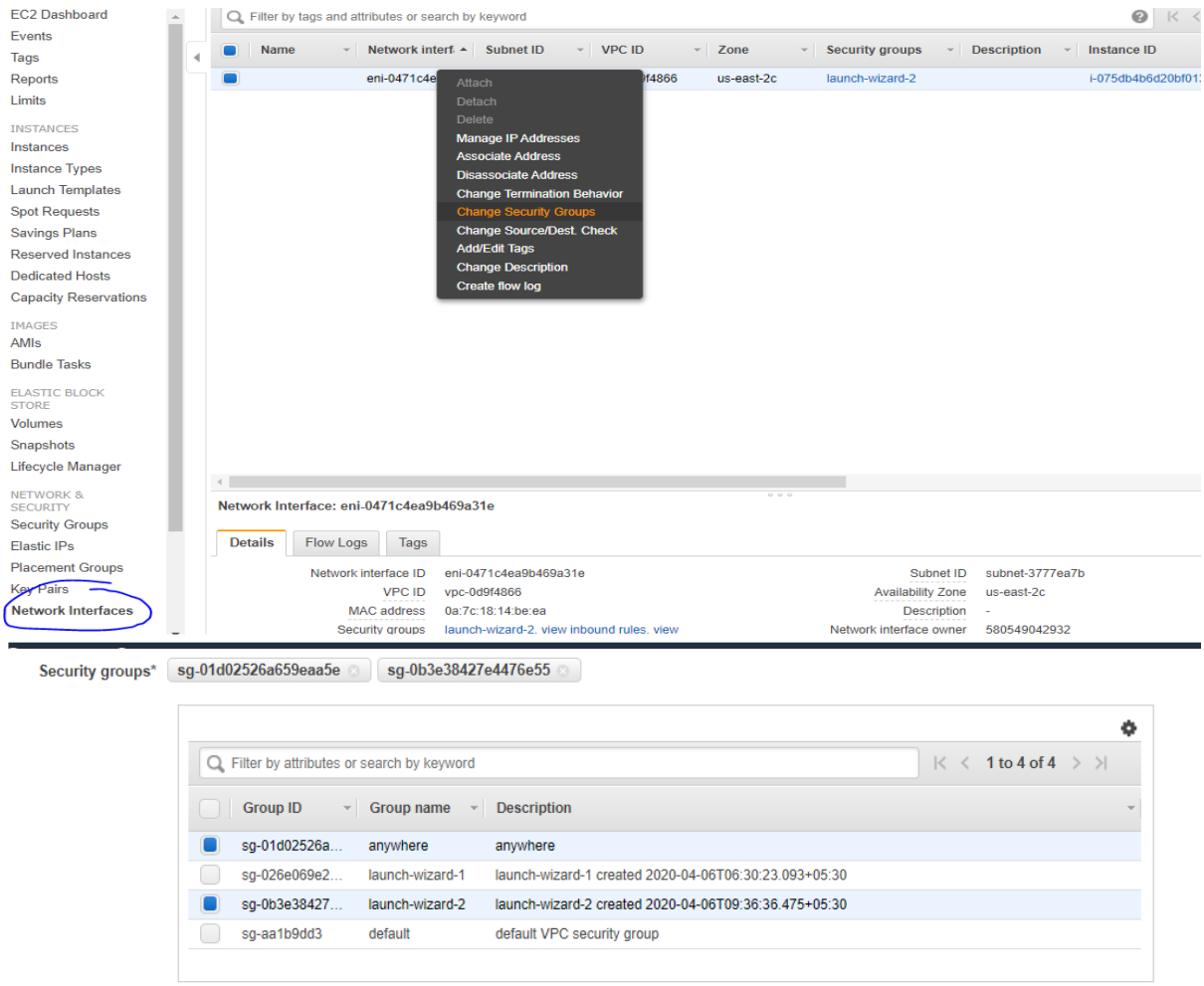
8. Final step:



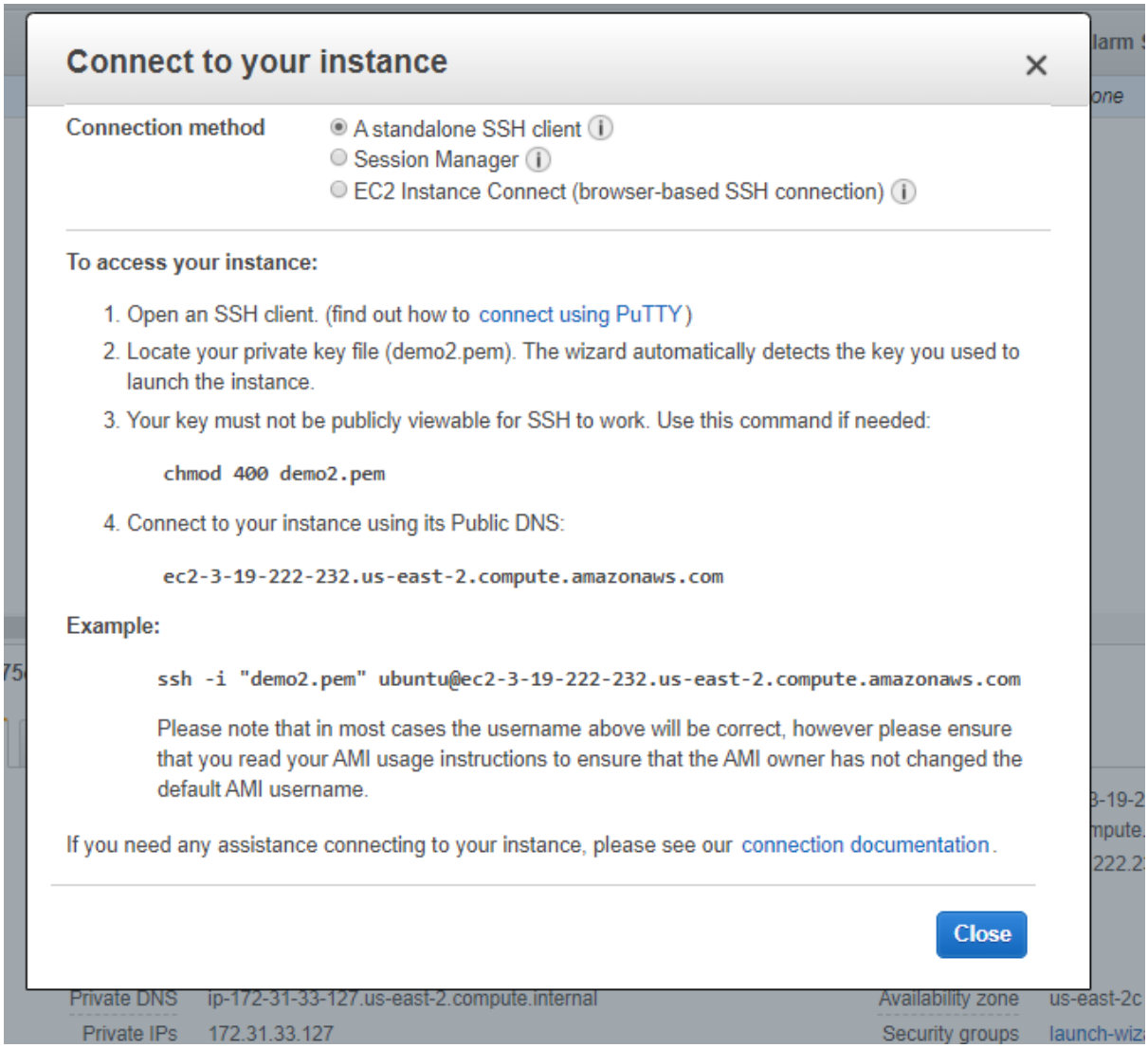
Select the “Network & security” -> Security groups and then click “Create Security Group”



Then add the specific security group to network interface

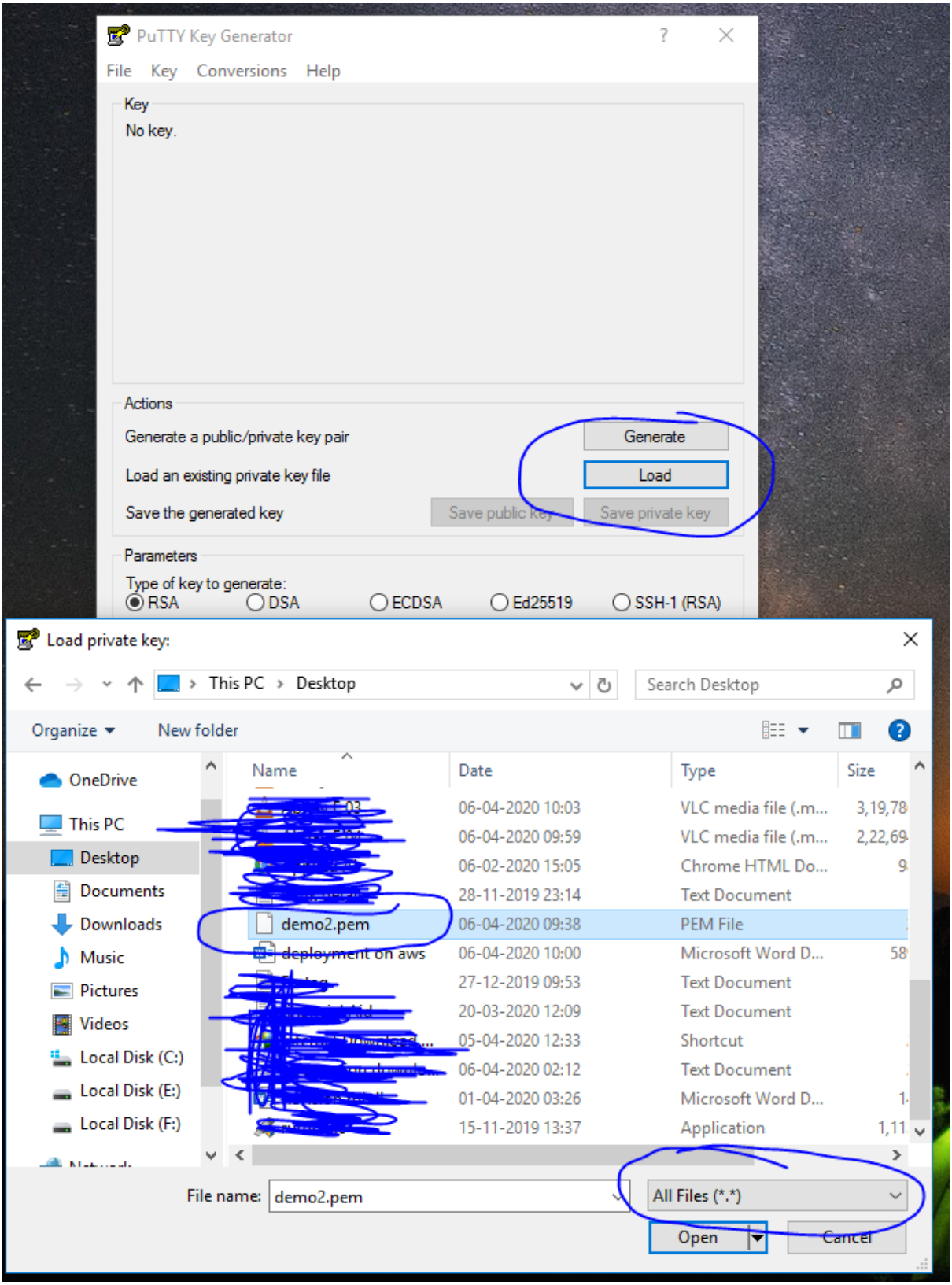


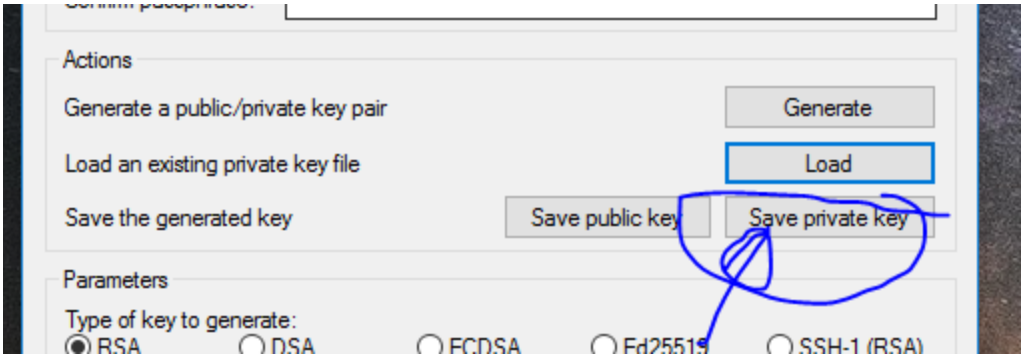
[3] Connect to the AWS box


Connect to your instance
Connection method
A standalone SSH client
Session Manager
EC2 Instance Connect (browser-based SSH connection)
To access your instance:
1. Open an SSH client. (find out how to connect using PuTTY)
2. Locate your private key file (demo2.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
chmod 400 demo2.pem
4. Connect to your instance using its Public DNS:
ec2-3-19-222-232.us-east-2.compute.amazonaws.com
Example:
ssh -i "demo2.pem" ubuntu@ec2-3-19-222-232.us-east-2.compute.amazonaws.com
Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.
If you need any assistance connecting to your instance, please see our connection documentation.
Close
Private DNS
ip-172-31-33-127.us-east-2.compute.internal
Availability zone
us-east-2c
Private IPs
172.31.33.127
Security groups

Download putty for windows.

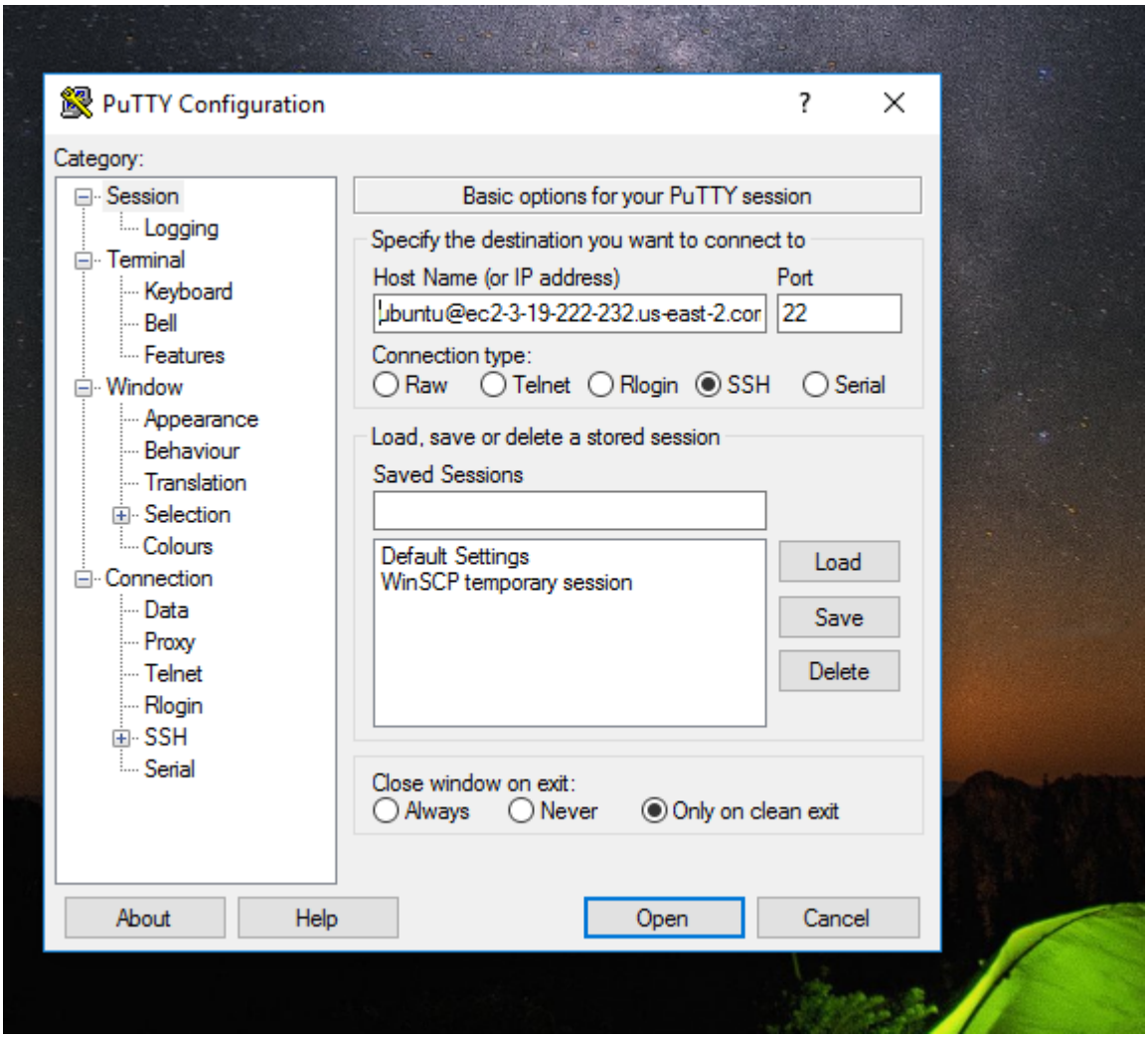
1. Open puttygen (to convert .pem key to .ppk)
   Steps to do that.


PuTTY Key Generator
File Key Conversions Help
Key
No key.
Actions
Generate a public/private key pair
Generate
Load an existing private key file
Load
Save the generated key
Save public key
Save private key
Parameters
Type of key to generate:
RSA DSA ECDSA Ed25519 SSH-1 (RSA)
Load private key:
This PC › Desktop
Search Desktop
Organize
New folder
OneDrive
This PC
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
Local Disk (F:)
Name
Date
Type
Size
06-04-2020 10:03 VLC media file (.m... 3,19,78
06-04-2020 09:59 VLC media file (.m... 2,22,69
06-02-2020 15:05 Chrome HTML Do... 9
28-11-2019 23:14 Text Document
demo2.pem 06-04-2020 09:38 PEM File
deployment on aws 06-04-2020 10:00 Microsoft Word D... 58
27-12-2019 09:53 Text Document
20-03-2020 12:09 Text Document
05-04-2020 12:33 Shortcut
06-04-2020 02:12 Text Document
01-04-2020 03:26 Microsoft Word D... 1
15-11-2019 13:37 Application 1,11
File name: demo2.pem
All Files (*.*)
Open
Cancel

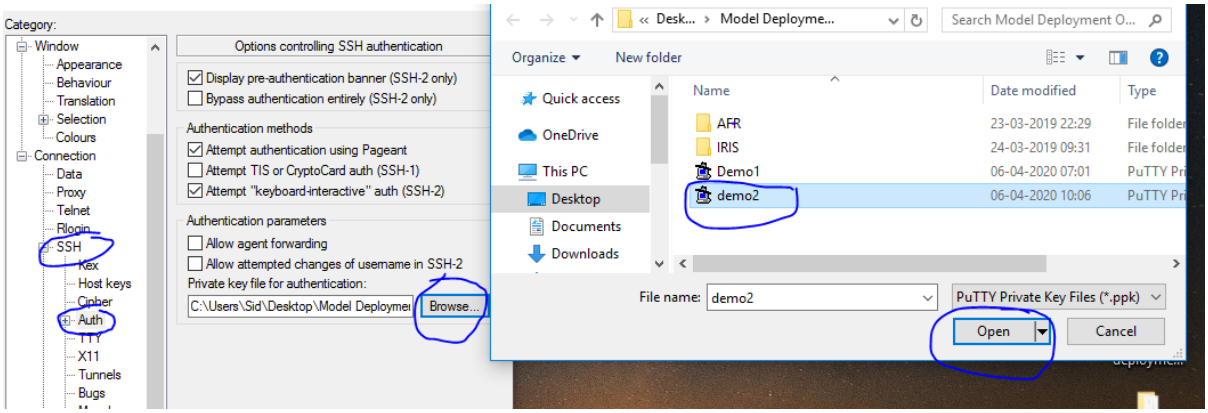


Save the private key.

2. Open Putty.



Enter Host Name: for ubuntu --→ the default username is ubuntu@{Followed by public dns}.



Set the Auth key.
Click on Open.

Now you are in the ubuntu shell.

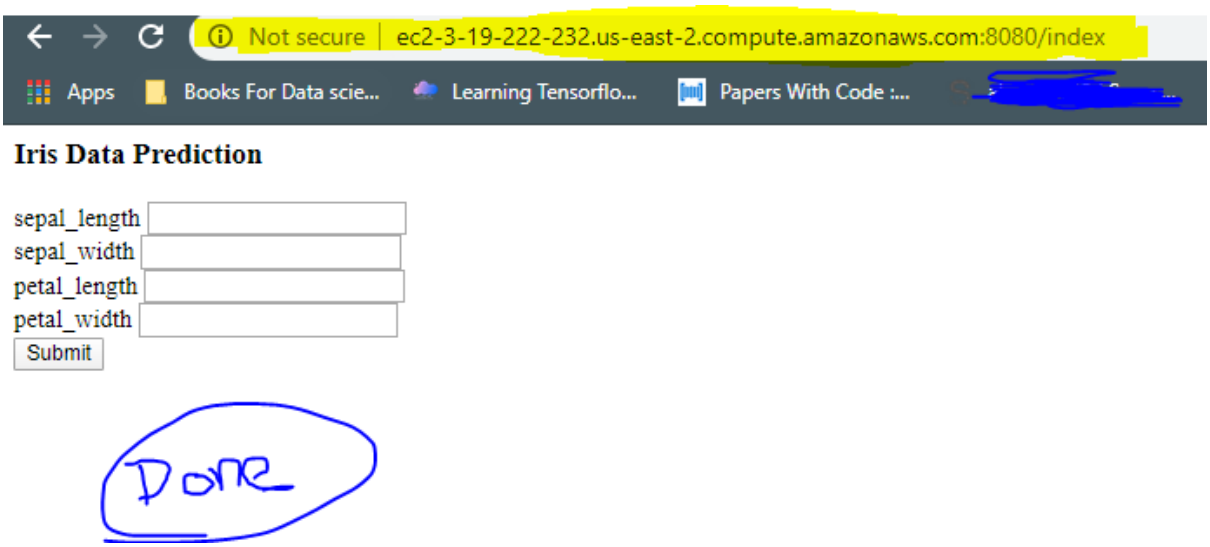
[4] Move the files to an AWS EC2 instance/box []
Command line to copy files
C:\Users\Asus\OneDrive\Desktop>scp -r -i "demo2.pem" ./AFR
ubuntu@ec2-13-59-191-237.us-east-2.compute.amazonaws.com:~/

[5] Install all packages needed on the AWS box.
sudo apt-get install python3-pip
pip3 install <each of the following packages>
packages needed:
pip3
pandas
numpy
sklearn
flask

6] Run app.py on the AWS box.

```
ubuntu@ip-172-31-33-127:~$ cd IRIS/
ubuntu@ip-172-31-33-127:~/IRIS$ ls
app.py  iris.csv  model.pkl  model.py  templates
ubuntu@ip-172-31-33-127:~/IRIS$ python3 app.py
/home/ubuntu/.local/lib/python3.6/site-packages/sklearn/externals/joblib/__init__.py:15: FutureWarni
 sklearn.externals.joblib is deprecated in 0.21 and will be removed in 0.23. Please import this func
nality directly from joblib, which can be installed with: pip install joblib. If this warning is rai
 when loading pickled models, you may need to re-serialize those models with scikit-learn 0.21+.
  warnings.warn(msg, category=FutureWarning)
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://0.0.0.0:8080/ (Press CTRL+C to quit)
```

Now app.py is running on my aws box.



Finally, Model is Deployed on AWS Ec2 Instance.

Deployment Document by:
Siddhant Jain