EC2 Setup [Perform for EC2_A and EC2_B]

Click on ***"Services"*** -> "EC2"

Click on ***"Instances"*** in the drop down ***"Instances"*** Section

Click on ***"Launch Instances"*** -> ***"Launch Instances"***

Select the "*Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type - ami-0cff7528ff583bf9a (64-bit x86) / ami-00bf5f1c358708486 (64-bit Arm)*"

Select the "t2.micro type (Free tier eligible)"

Click "Next: Configure Instance Details"

Ensure "Number of instances" is "1"

Click "Next: Add Storage"

Click "Next: Add Tags"

Insert "EC2_A" under "Key" and "Value" and “EC2_B” for the second one

Click "Next: Configure Security Group"

Click "Add Rule" to include the following and it will autofill:

SSH, HTTP, HTTPS

Under 'Source' drop down for each rule, select 'My IP'

Click "Review and Launch"

Click on "Launch" after reviewing your information

On the dialog that pops up, select "Create a new key pair" in the drop down

Name it "EC2_A_CS643" and “EC2_B_CS643” for 2nd EC2

Hit "Download key pair."

Note: Use the same keypair for 2nd EC2 aka EC2_B. Do not create a new one

---

Hit ***"Launch Instances"***

Hit ***"View Instances"***

Run the following command to set the correct permissions for the .pem file: (I used Windows Subsystem Linux for this)

$ chmod 400 EC2_A_CS643.pem

Use Putty to Connect to instance. Refer this if you do no know how to connect using PUTTY

After you have connected, run the following commands to update Java from 1.7 to 1.8 on the EC2:

$ sudo yum install java-1.8.0-devel

$ sudo /usr/sbin/alternatives --config java

Upon running the second command, you should enter the number that corresponds to

java-1.8.0-openjdk.x86_64 (/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.322.b06-2.el8_5.x86_64/jre/bin/java)

You will need to complete this step on BOTH EC2's.

Credentials Setup [Access and Secret Keys]

Click on "Services" -> "IAM"

On the left tab side, under "Access Management" -> "Users"

Click on user created (e.g. john)

Open the "Security credentials" tab, and then choose "Create access key"

To see the new access key, choose "Show". Your credentials resemble the following:

Access key ID: AKIAQB2SHQLRR6NF57M5

Secret access key: 73IWqwYi2nyPG0f16AnCjDARbBPK3zS6Uuts+p8i

To download the key pair, choose 'Download .csv' file.

Store the .csv file with keys in a secure location.

Connect into both EC2's that you created in the previous step, and on each one, create a file:

$mkdir .aws

$touch .aws/credentials

$vi .aws/credentials

**Paste those copied credentials into the credential file on each EC2 as follows:

[default]

aws_access_key_id= THI5I5NOTTH3ACCE55K3Y

aws_secret_access_key= THI5I5NOTTH3S3CR3TACCE55K3Y

Note: You will need to re-copy and paste these credentials onto both EC2's whenever they change (after your session ends).

--Java/Git Installation

If you do not have an up-to-date version of Java or Maven installed on your local machine, run the following commands locally:

$sudo amazon-linux-extras install java-openjdk11

$java -version

$alternatives --config java

Now select the one which has JAVA 11

Refer this to install git in your EC's.

SSH into EC2-B and run the following command:

$ java -jar **locate the jar file for text recognition

**This will begin running the code for text recognition. However, since it depends on items in the SQS queue to process, it will simply wait until we begin running the car recognition code. To begin running the car recognition code.

SSH into EC2-A and run the following command:

$ java -jar **locate the jar file for car recognition

Now, both programs will be running simultaneously. The program on EC2-A is processing all images in the S3 bucket (njit-cs-643) and sending the indexes of images that include cars to EC2-B through SQS, which in turn is processing these images to find out which ones include text as well.