**Execution Steps**

Pre-requisites : Setup the EC2 instance and download the .pem file which will be required to connect to the instance from the system.

**Connect to ec2 instance start and stop the instance**

- Start the ec2 instance using the below command from the terminal

**aws ec2 start-instances --instance-ids <instance id>**

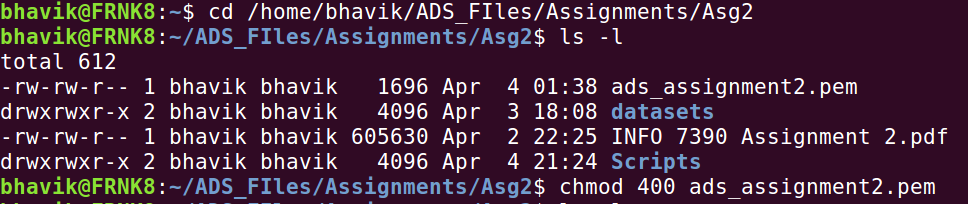
- You can check the S3 Management console through the browser for the status of the instance and other management details.

**ssh the ec2 instance from local.**

1. Connect to the EC2 Instance

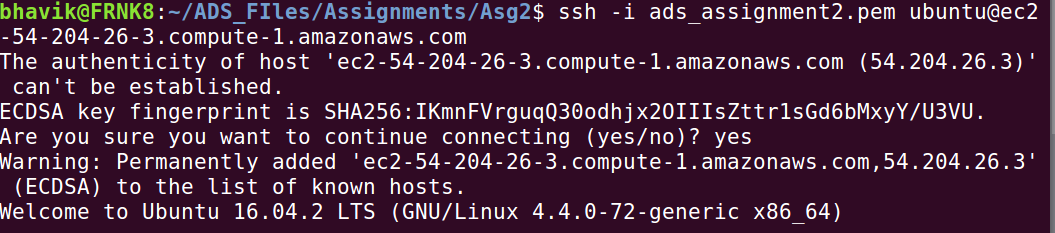
- Go to the directory where the .pem file is downloaded for the image. This file will be required to connect to the ec2 instance.

- Change the permissions of the file to 400.



- Time to connect to the instance using “ssi -i <.pem file> <user>@<public-ip of the ec2 instance>” command

- Can use elastic IP as well and can execute the connection to instance through a script.



- If everything is followed correctly till now, you will see the Ubuntu Instance connected (message “Welcome to Ubuntu...” is seen).

**- Install Docker on ec2 Linux AMI Instance**

**(Note : Just to keep in mind, we are executing the below commands after connecting to the ec2 instance and not on the local system. verify by checking user@<private ip of the ec2 instance>)**

- Below are the commands to install Docker on Linux 16.04 ec2 instance

- The URL or any key value can be used as-is as seen in the screenshot for the installation process.







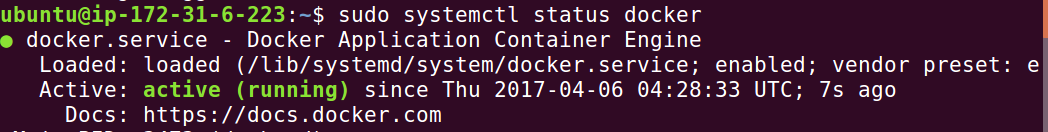




- Install docker engine



- Using the below command the status can be seen if the engine is running. (active)

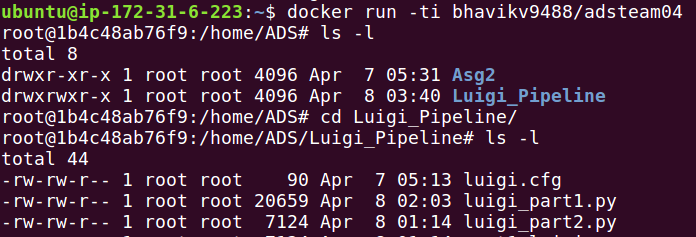


- You can now continue working on the docker on the ec2 instance.

**Pull and run the Docker image** ( “bhavikv9488/adsteam04” )

- Once the docker is installed on the system, pull the mentioned image.

- Go to the directory where the python scripts are stored( **/home/ADS/Luigi\_Pipeline**)



**To execute the Pipeline scripts manually**

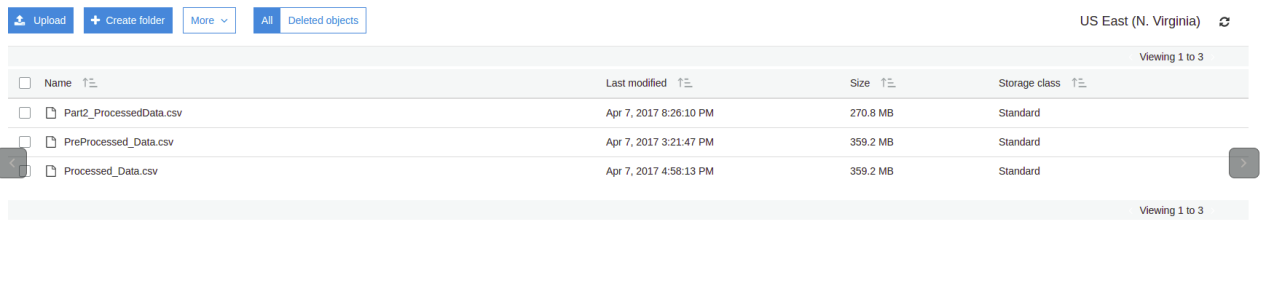
**Part1 - Loans Accepted**

*python luigi\_part1.py UploadToS3 --local-scheduler*

**Part2 - Loans Declined**

*python luigi\_part2.py UploadToS3 --local-scheduler*

- Once the Execution is complete for both the scripts the file can be seen on the S3 console as seen below.

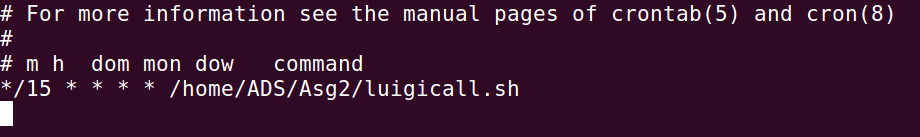


**Scheduling of the Pipeline**

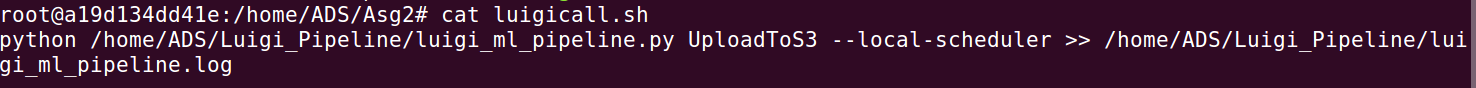
**- We have scheduled the pipeline on the docker image using the crontab to be executed every hour of the day.**

**Luigi Pipeline Schedule (On the Docker Image)**

- Execute script to schedule a pipeline on the docker image executing every 15 mins.



- Luigicall.sh contains the command to execute the pipeline script



- Once all the operations are complete on the docker, exit the docker image. Commit and push the image if required.

- Disassociate and Release the Elastic IP if used else you can directly exit from the ec2 instance.

- Stop the ec2 instance.

**aws ec2 stop-instances --instance-ids <instance id>**