**AMAZON SAGEMAKER**

* Amazon SageMaker is a service that enables a developer to build and train machine learning (ML) models for applications in the Amazon Web Services (AWS) public cloud.
* Basically it lets you to build ML model and deploy it on one single platform.
* Advantages of Sagemaker:

1. **Reduces Complexity:** All kind of work like preprocessing data, build, train, tune and deploy a model, can be done on one single platform.
2. **Provides Independency:** Here you can build a model at one place and deploy it at another place.

* What it does:

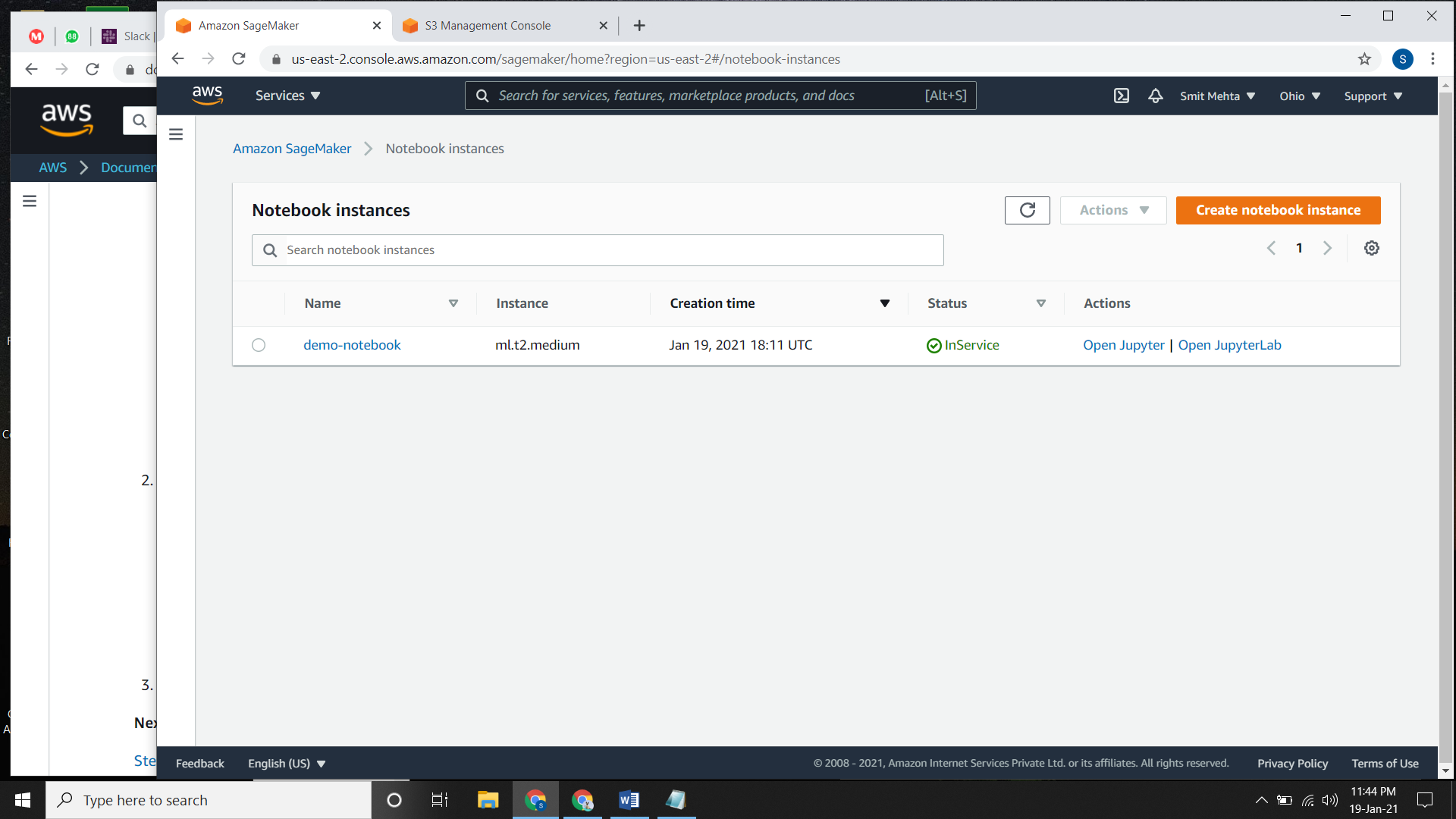
1. Select & Prepare Training Data.
2. Choose & Optimize your ML models.
3. Setup & Manage Environment for Training.
4. Train & Tune Model.
5. Deploy your ML models to Production.
6. Scale & Manage Production Environment.

* How it Works:

1. **Build:** Connects to other AWS services and transform data into Amazon Sagemaker Notebooks.
2. **Train:** You can bring your own model or can use Amazon Sagemkaer’s Algorithms and frameworks.
3. **Tune:** Amazon Sagemaker automatically tunes your model by adjusting multiple combinations of algorithm parameters.
4. **Deploy:** Once your model is ready, you can deploy it to Amazon Sagemaker endpoints for real-time prediction.

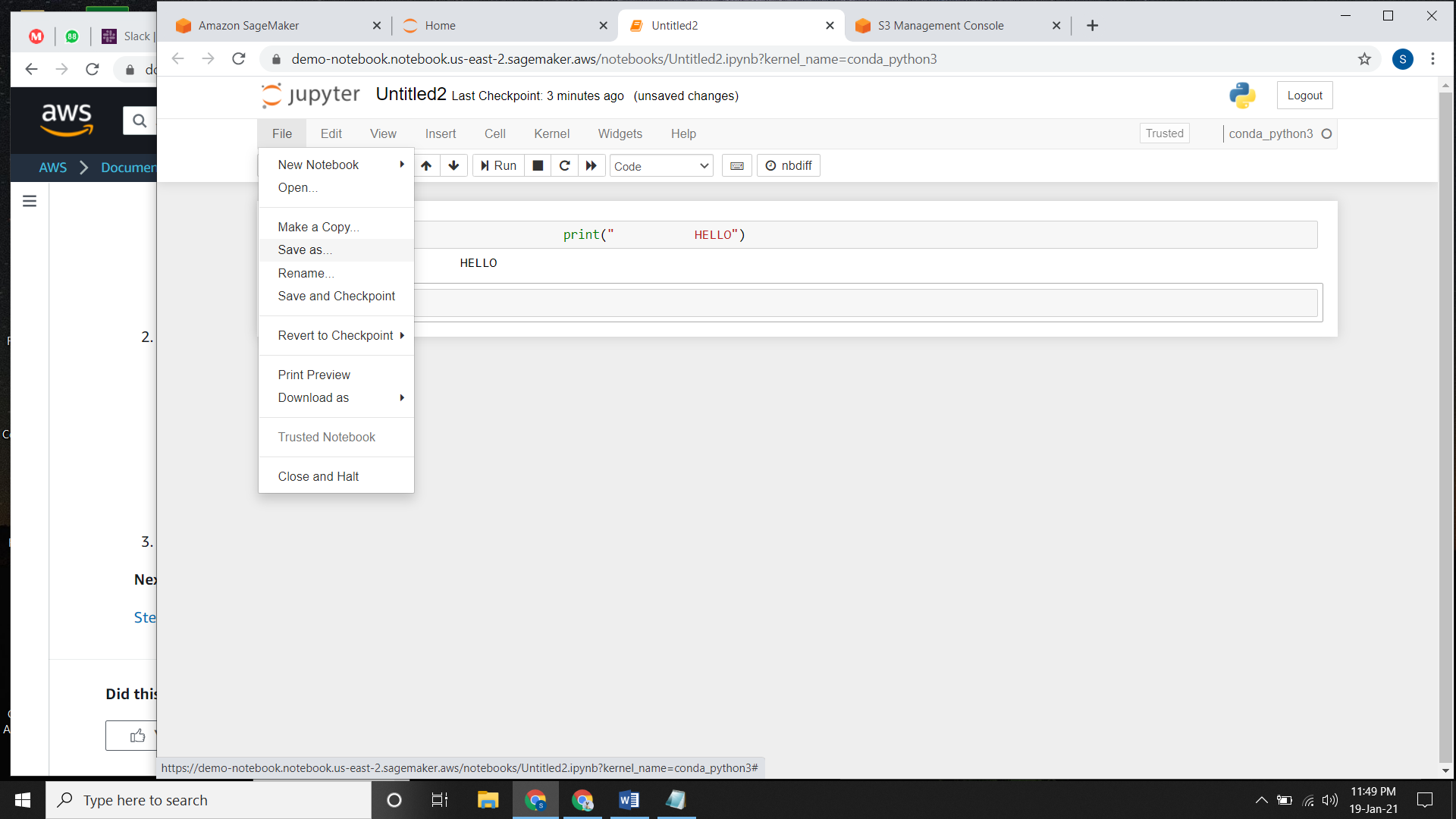
**Creating an Amazon SageMaker Notebook Instance:**

* Firstly, you need to create a S3 bucket for this.
* Go to Amazon SageMaker dashboard, click on **Notebook Instances** and create one.
* Give a **name** and select an **instance type** as required.
* Under **IAM role**, choose one, if you don’t have it just **create a role.** While creating a new role, select the **bucket** conditions accordingly and create it.
* Add **Encryption key, VPC and Git Repository** according to your requirement as it is optional and then **create notebook instance.**
* It will take some time to go into ready state.

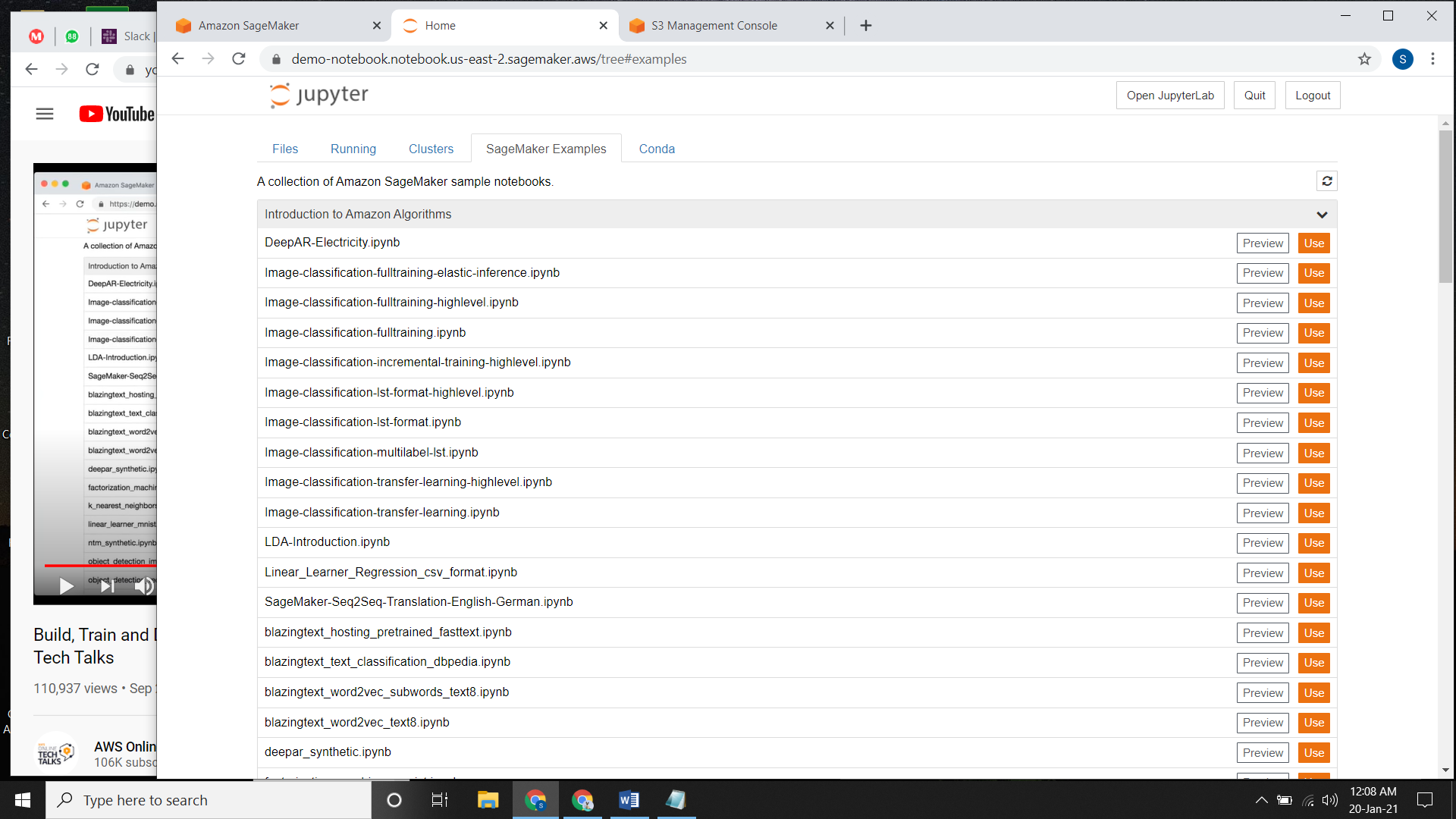


**Create a Juptyer Notebook:**

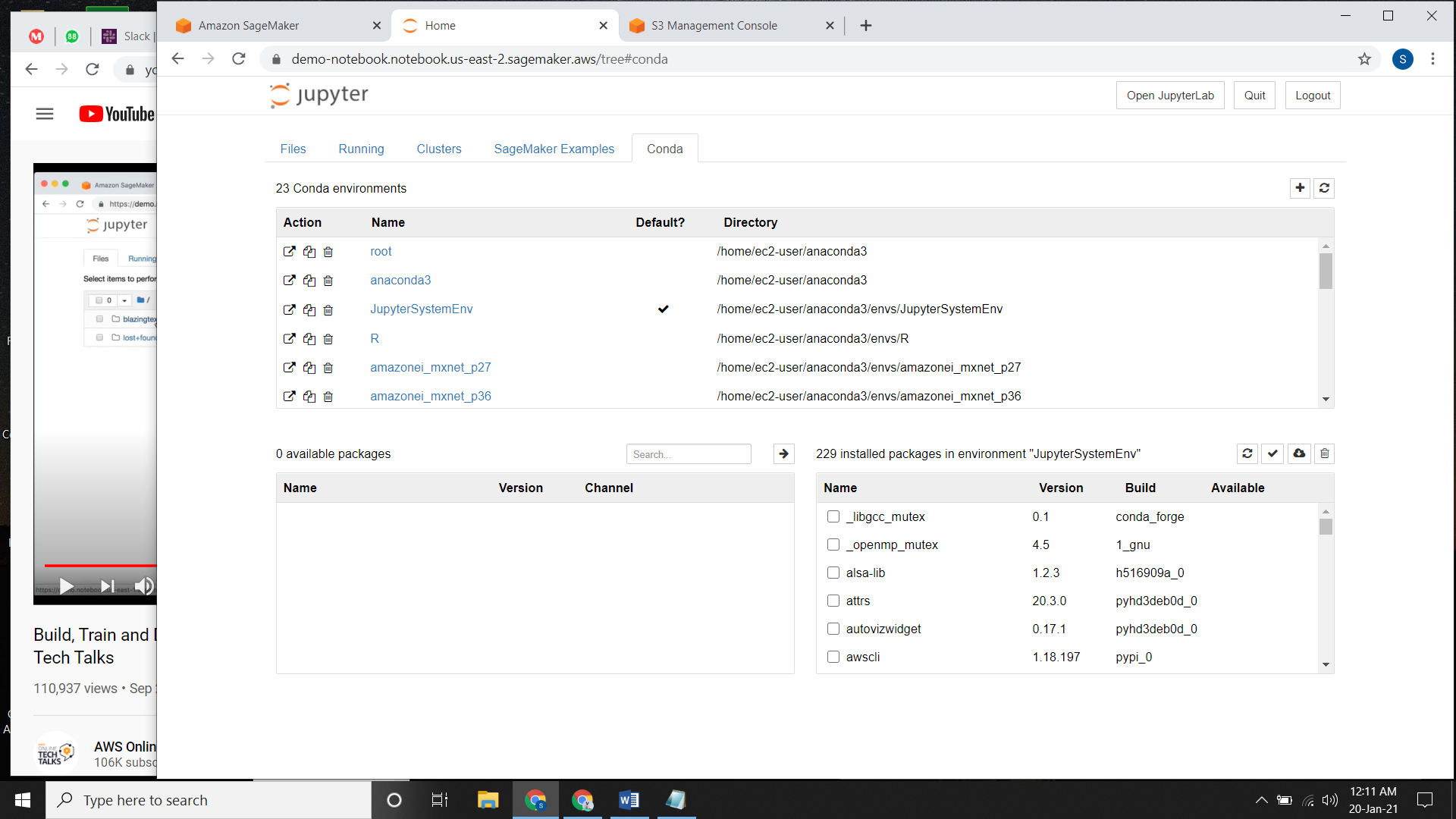
* As your notebook instance is in ready state, in the **status** tab click **Open Jupyter.**
* A new tab will open and click on **new** and choose a **Kernel.**
* Here you can write your code and in the **file** section you can **save** it too.



* In **SageMaker Examples**, there are a set of predefined notebooks that we load to the instance and use it whenever required.



* The **Conda** tab described the environment that we have created on the SageMaker notebook instance as well as it keeps an eye or keeps track on packages.



**Download the MNIST Dataset:**

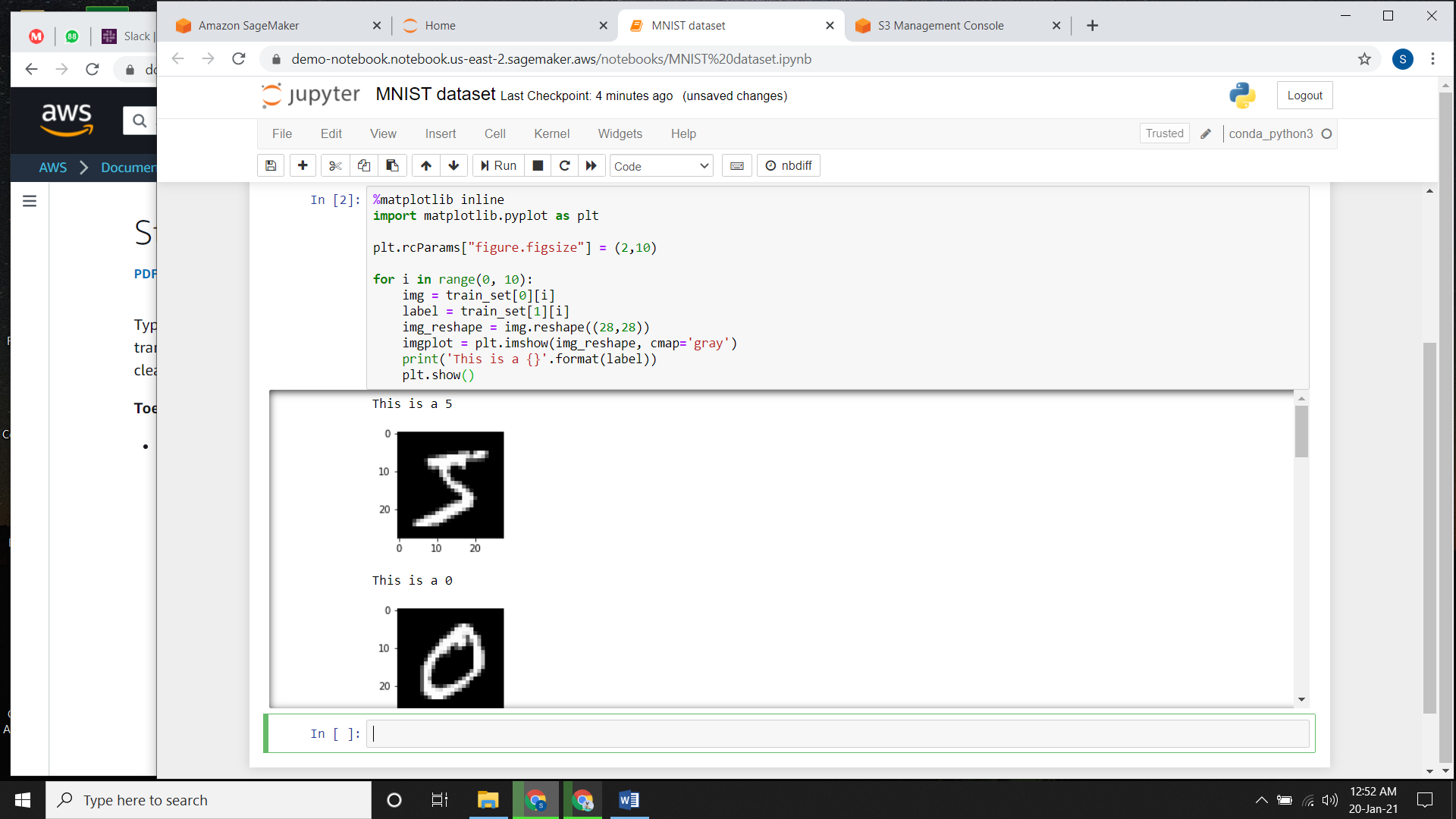
* Open the **Jupyter notebook**, select the **Kernel** and write the code.
* Following code is used to download the MINST dataset.



* After writing this code, go to your **home** page for jupyter notebook and download the minst.pkl file and save your code file too with a name.
* This code downloads the MNIST dataset, unzips it and read the following datasets. The datasets are **train\_set, valid\_set and test\_set.**

**Explore the Training Dataset:**

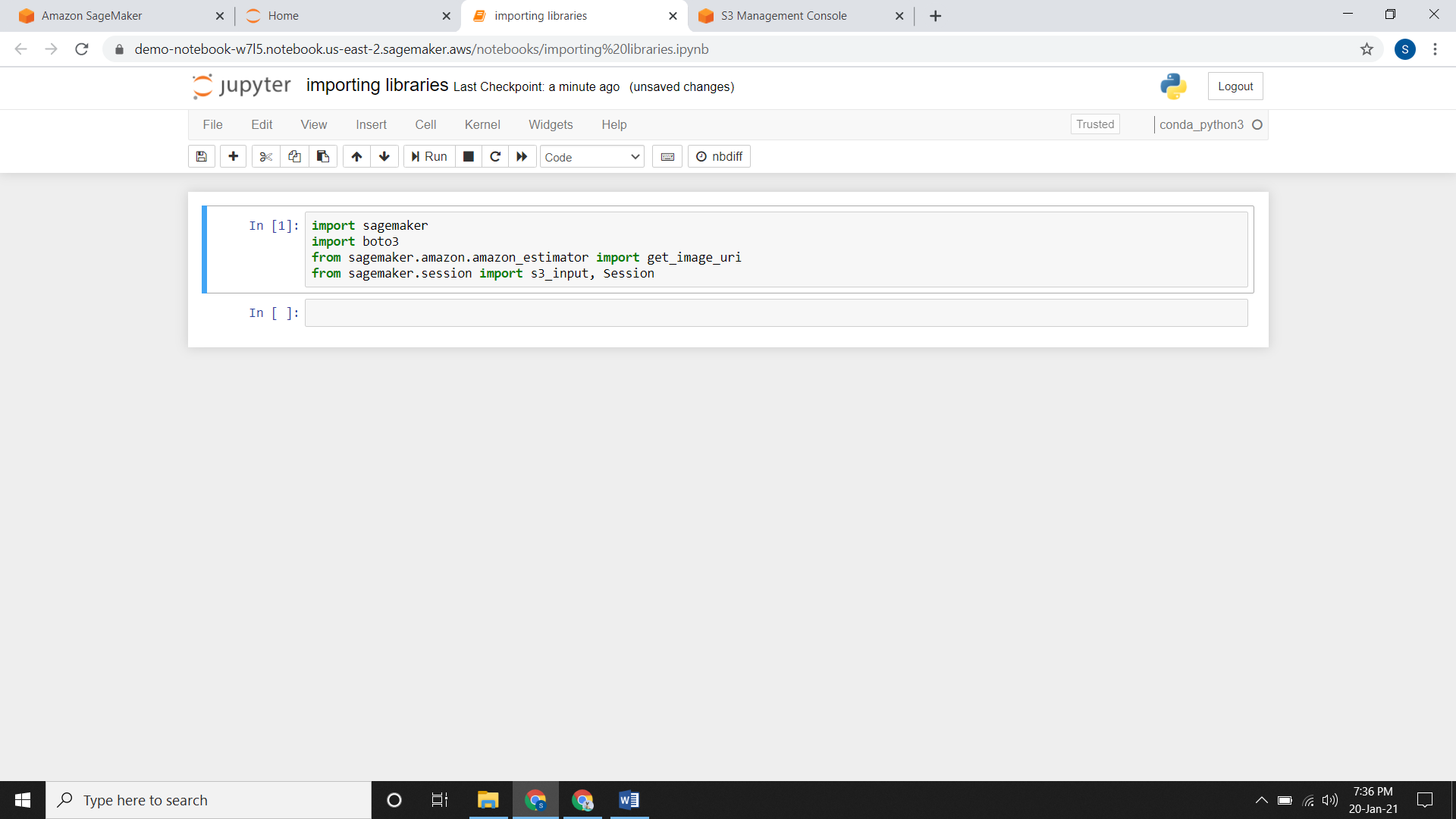
* After downloading the file, write the following code for displaying the first 10 images in train\_set.



**Building an XGBoost Algorithm**

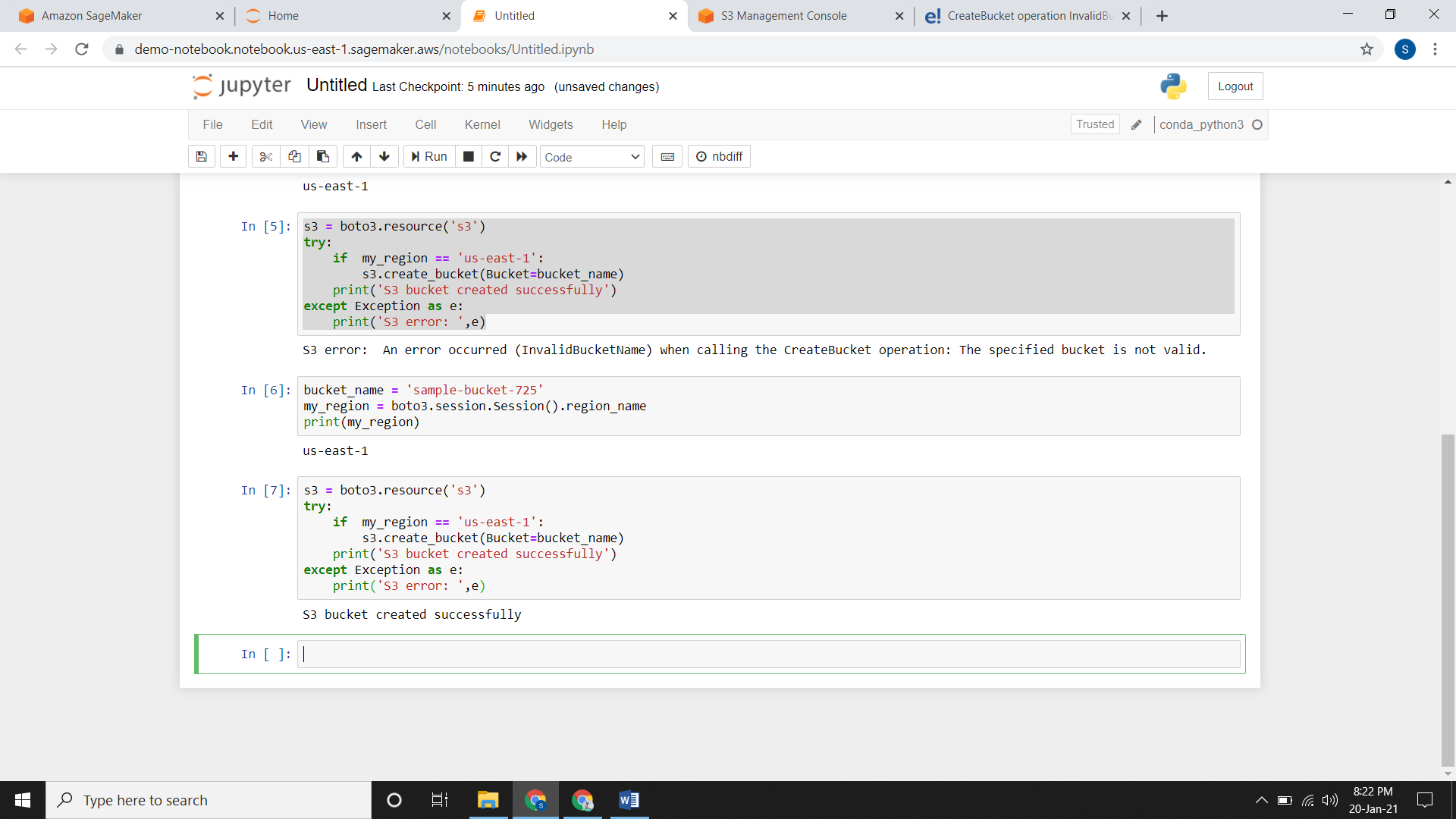
**Importing Libraries:**

* For training any model you need to import the libraries which is needed

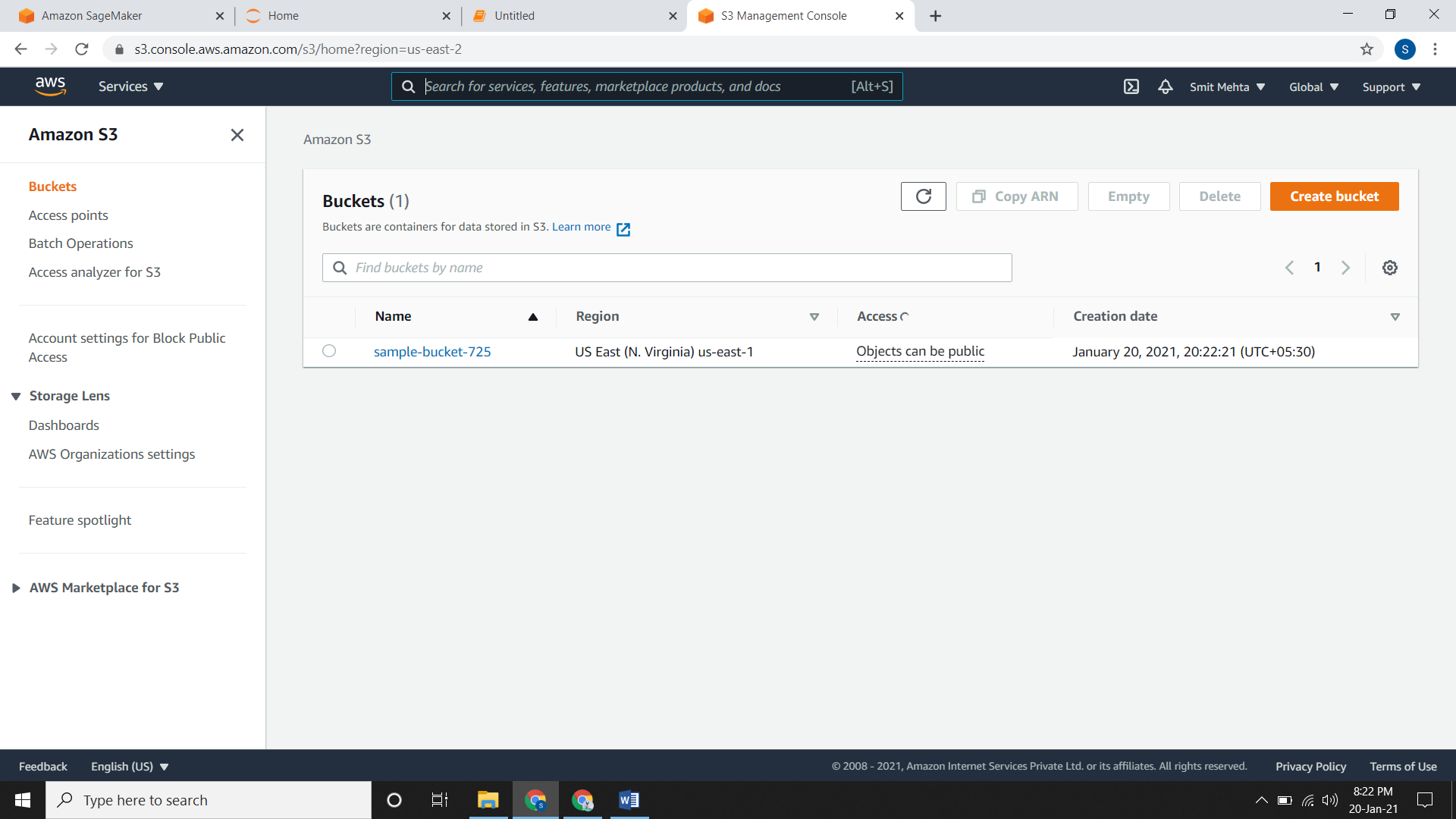


**Creating S3 Bucket:**

* We will create a S3 bucket by which all the data will be stored in it.

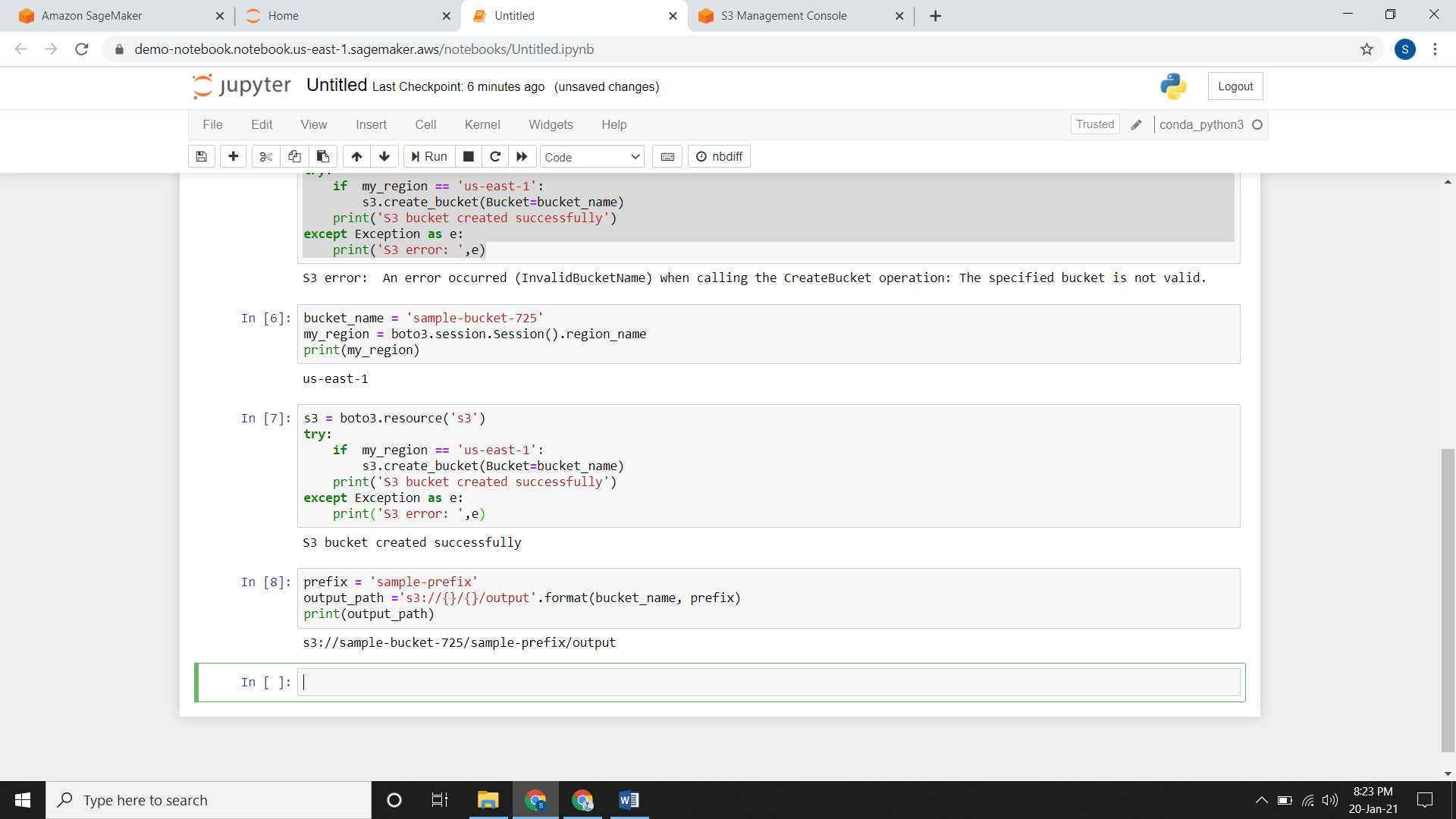


* Now if you check in the S3 console, the bucket will be there.



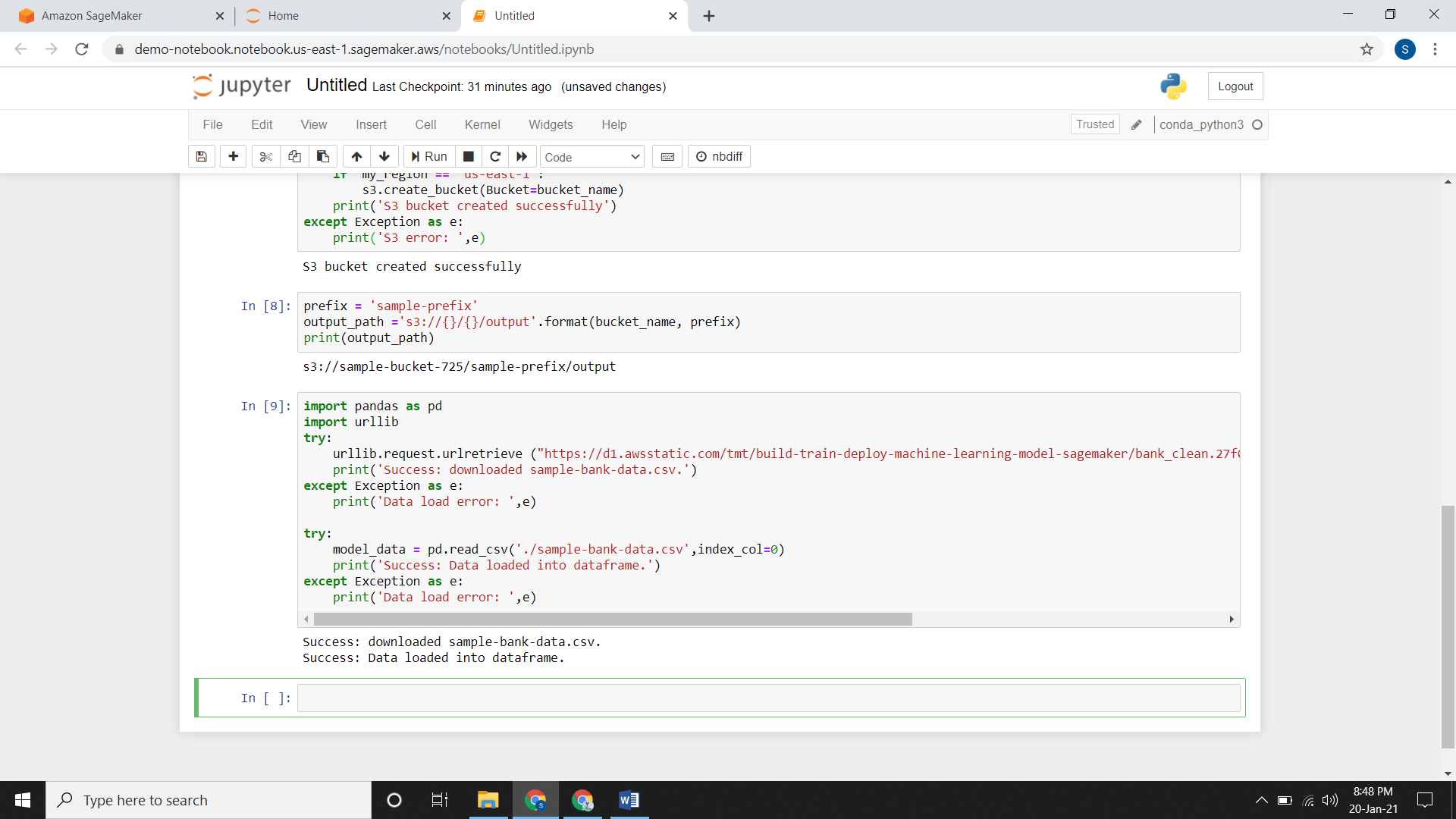
**Setting an Output Path:**

* You need to specify an output path which will tell you that where your your files and folders going to save.



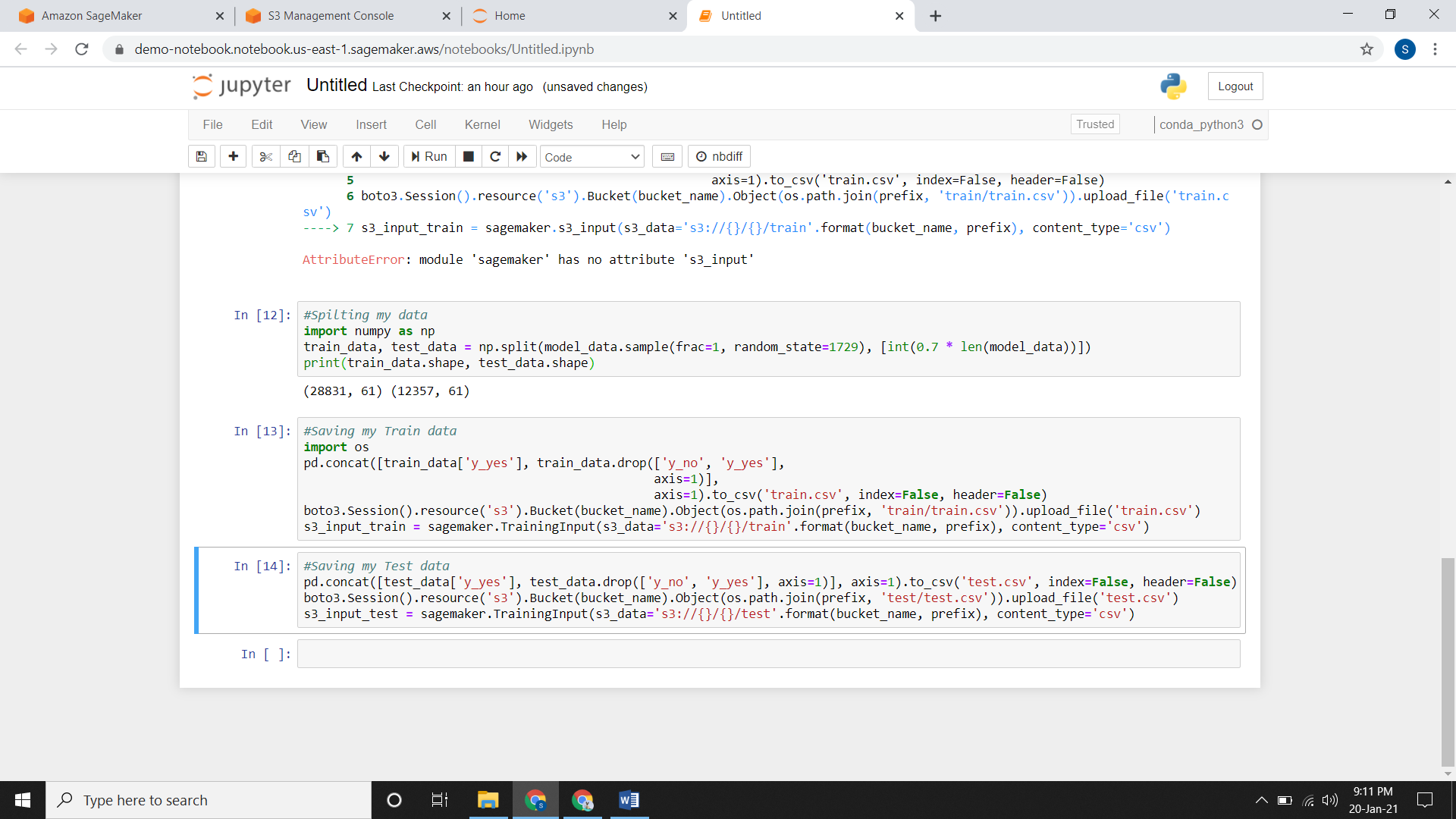
**Downloading the Dataset:**

* You need to download your dataset and store it in the dataframe.

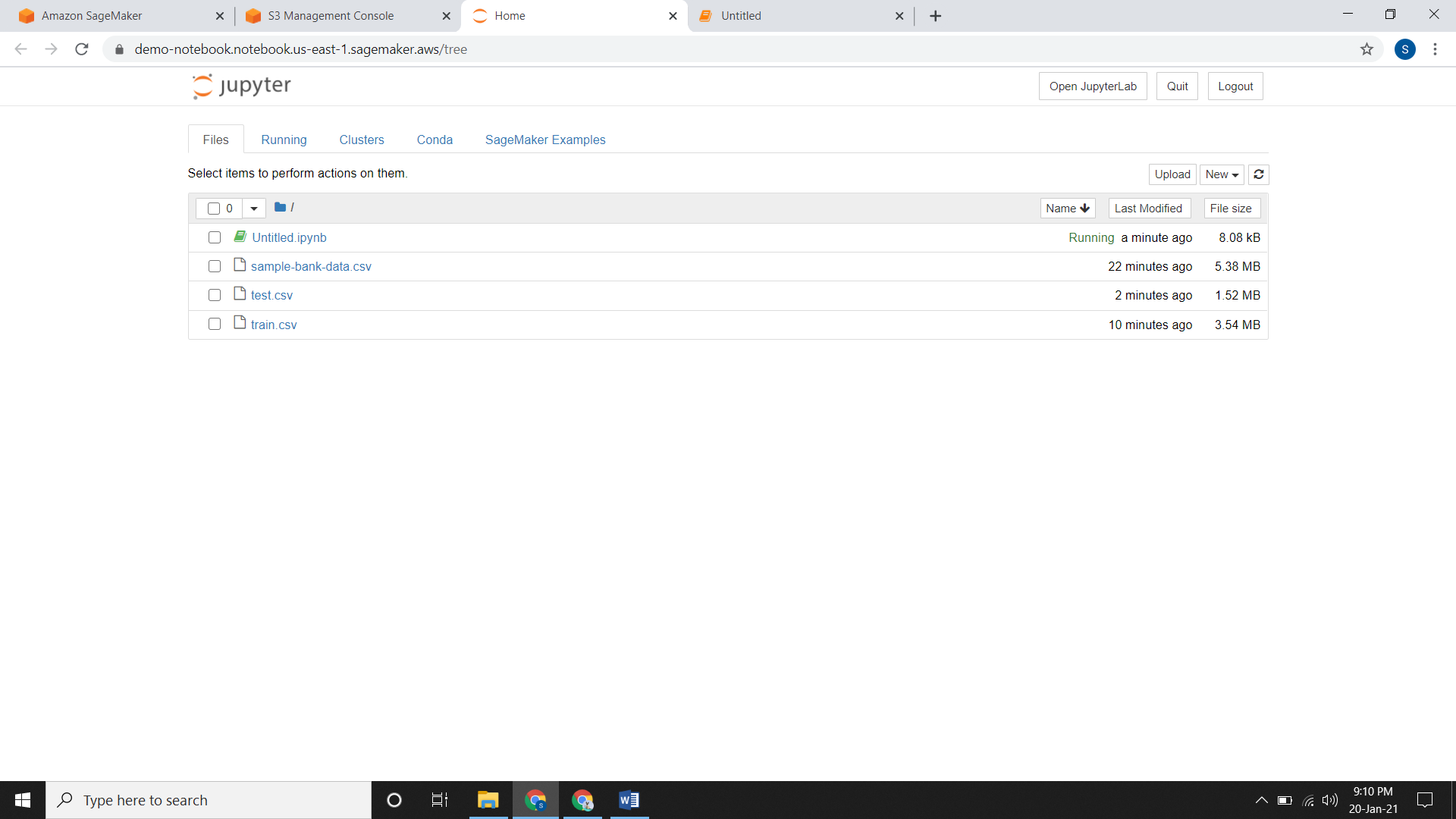


**Saving the data to S3**

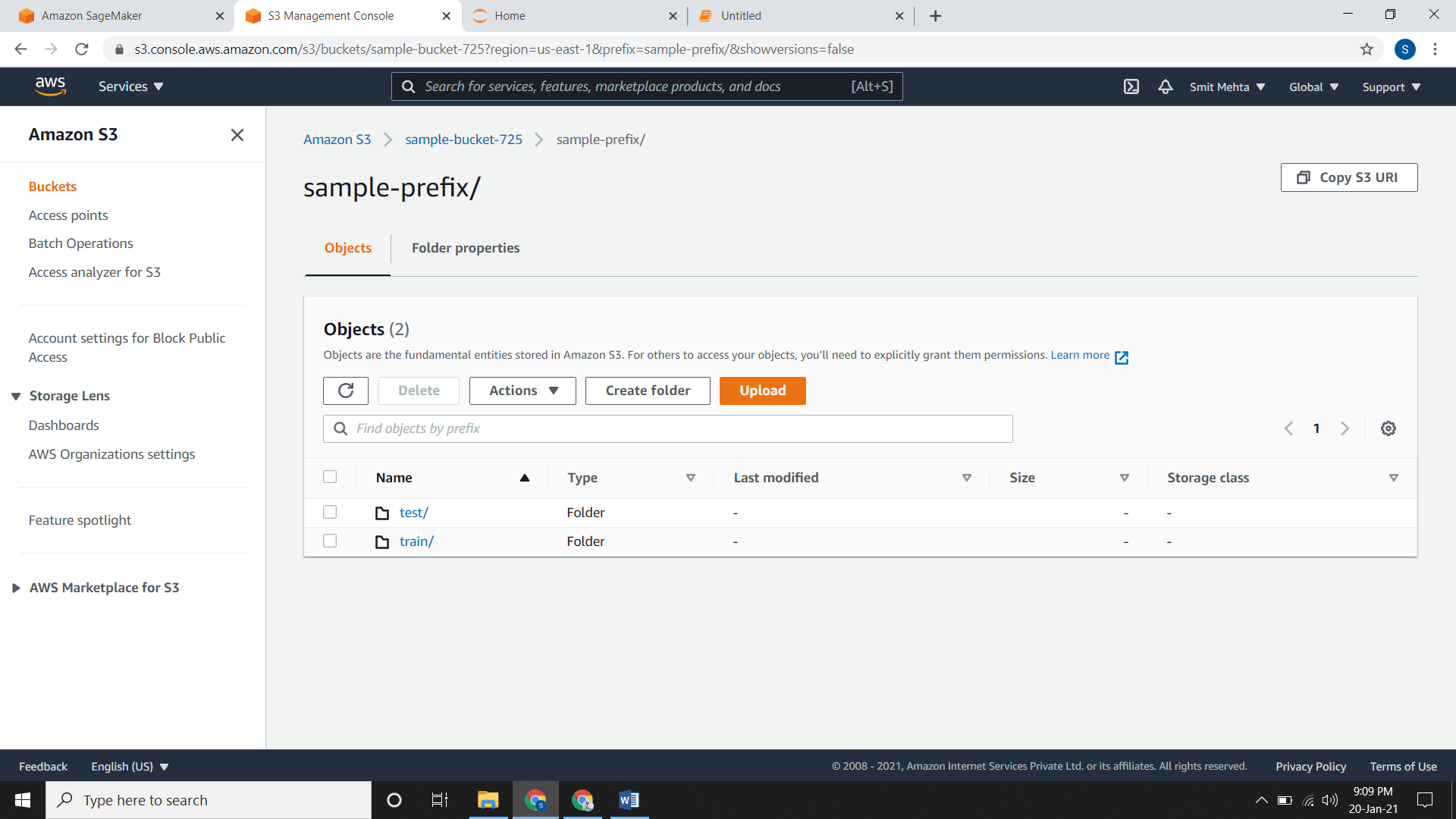
* After modifying the data, like I have spilt my data we need to save it to the S3 bucket



* Now check your dataframe or the home page for jupyter/jupyter notebook that will contain your csv files.

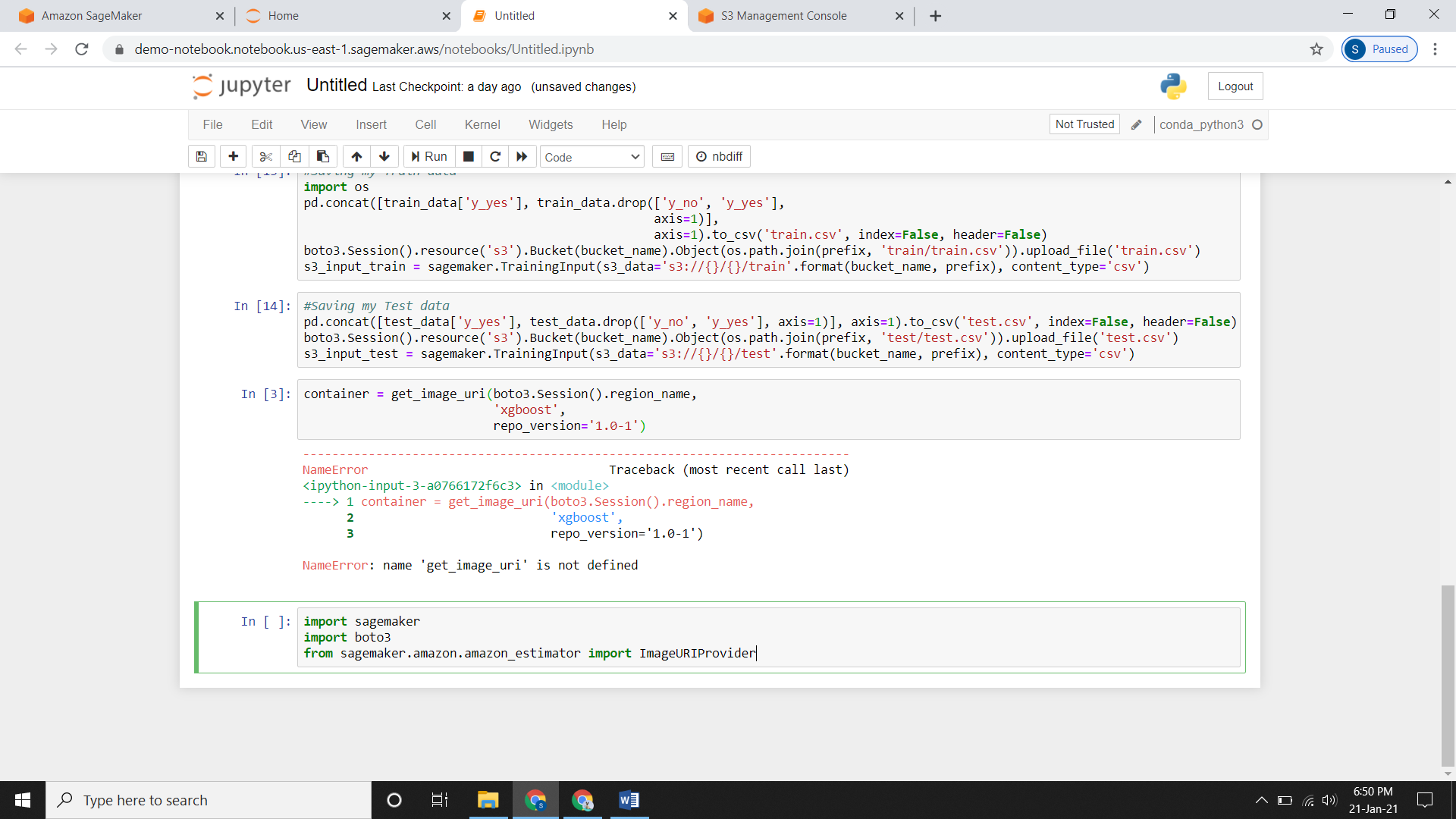


* Your file will also be there in your bucket.

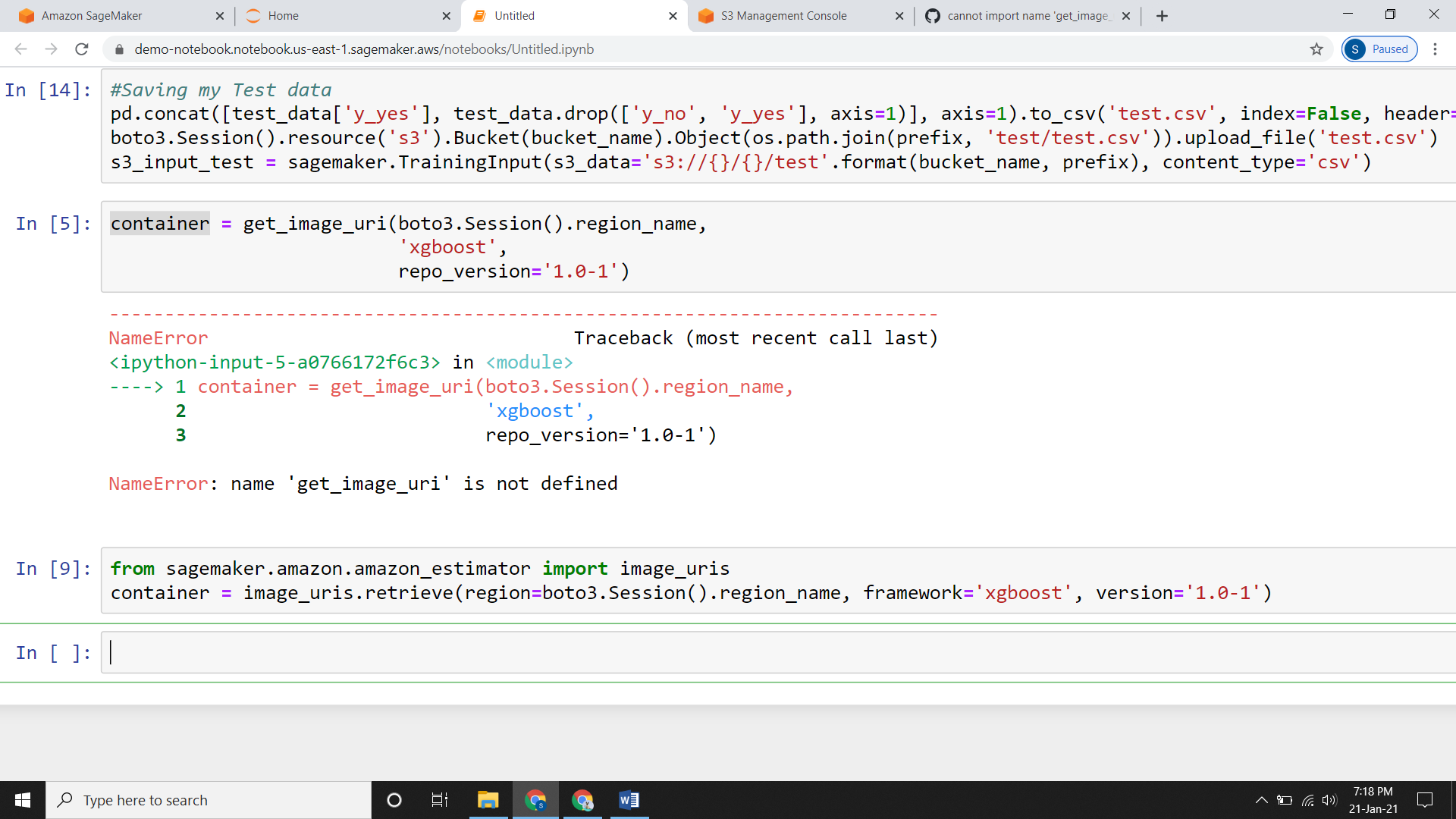


**Building & Training an Inbuilt Xgboost Algorithm:**

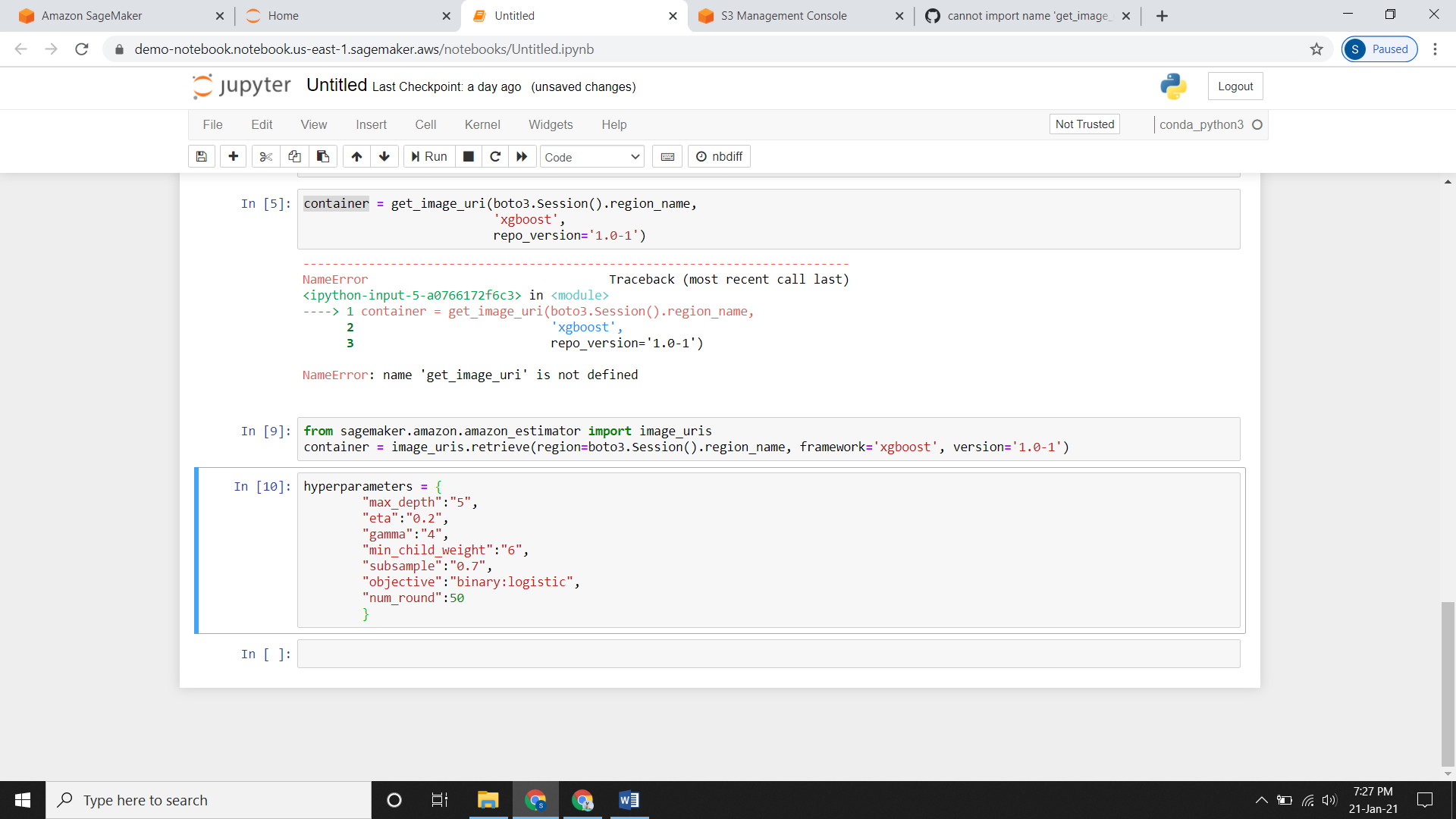
* Since this is an inbuilt algorithm we need to pull the containers into our local system or the instances.
* You need to look for the XGBoost image uri and build a XGBoost container.
* If you are working on old version you will get this error.



* You need to type this command.

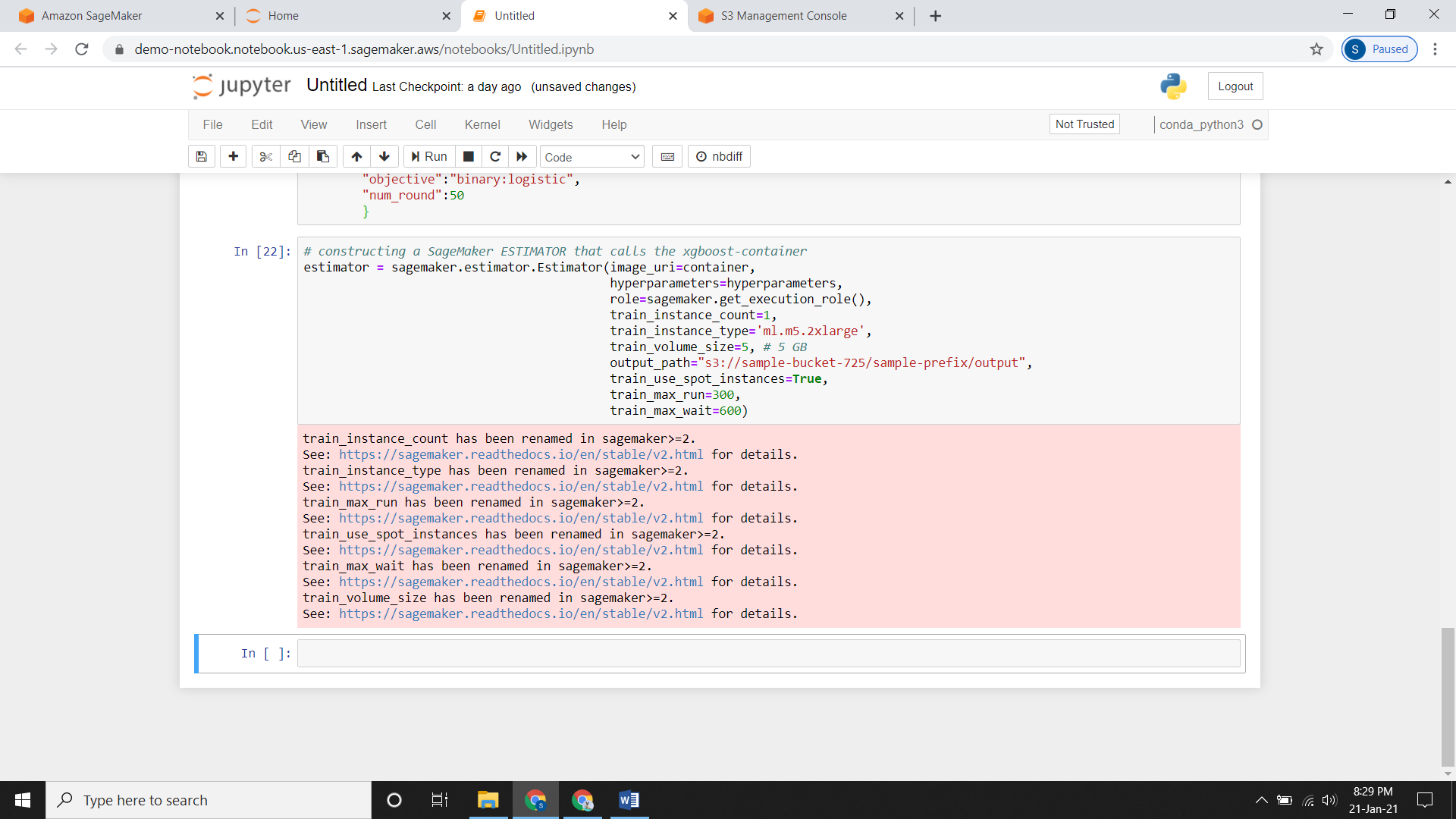


* Now you need to give some hyperparameters.
* It finds the best version of a model by running many training jobs on your dataset using the algorithm and ranges of hyperparameters that you specify.
* We can give hyperparameters from Sagemaker also but it will cost us more so it is efficient that we use the hyperparameters from here.



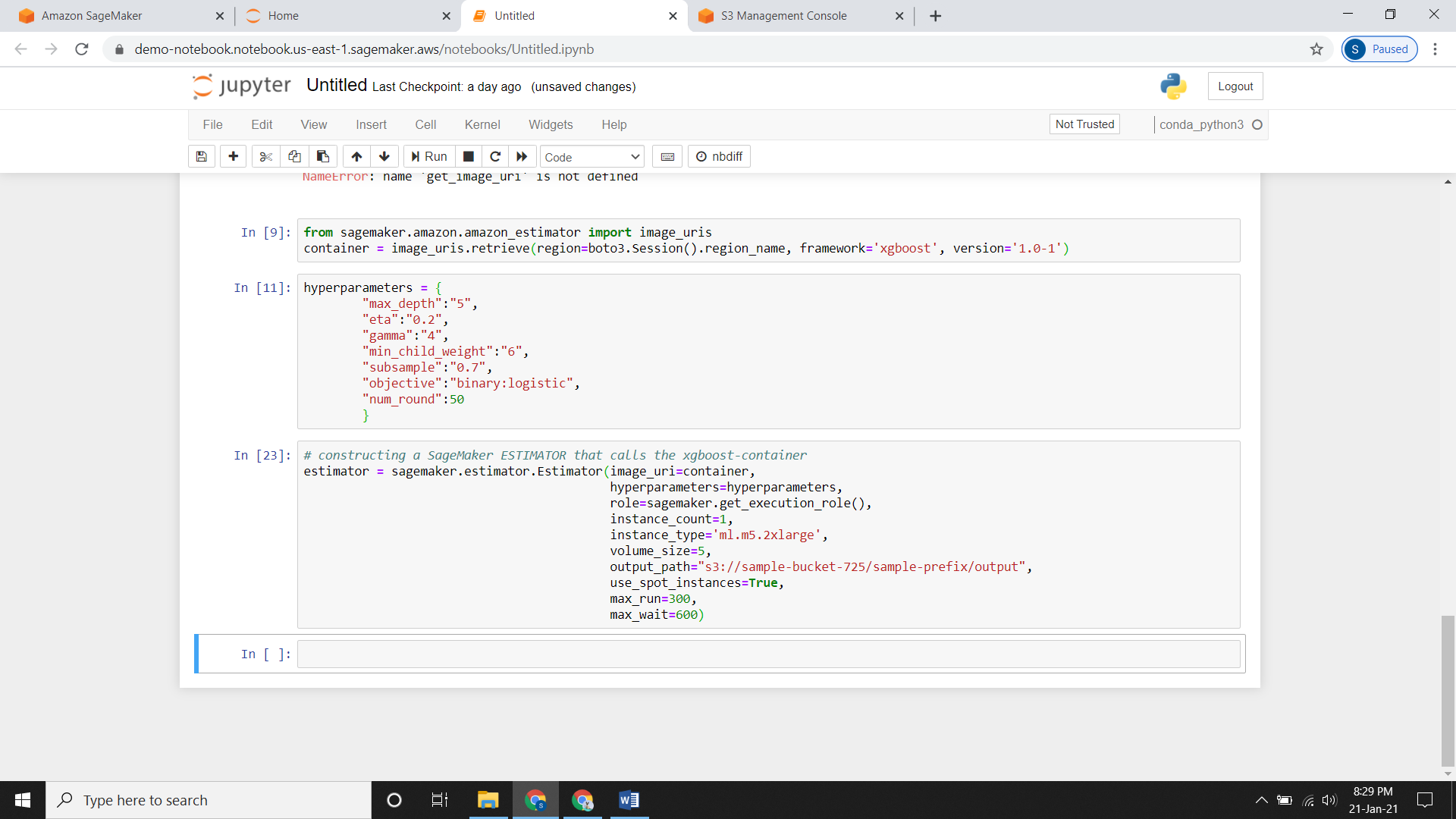
**Constructing an Estimator:**

* Now the image which we have in our container needs to be called.
* For that SageMaker estimator is used so that it will call the xgboost container.
* Now if you are using the new version and you write this command then you will face these errors.



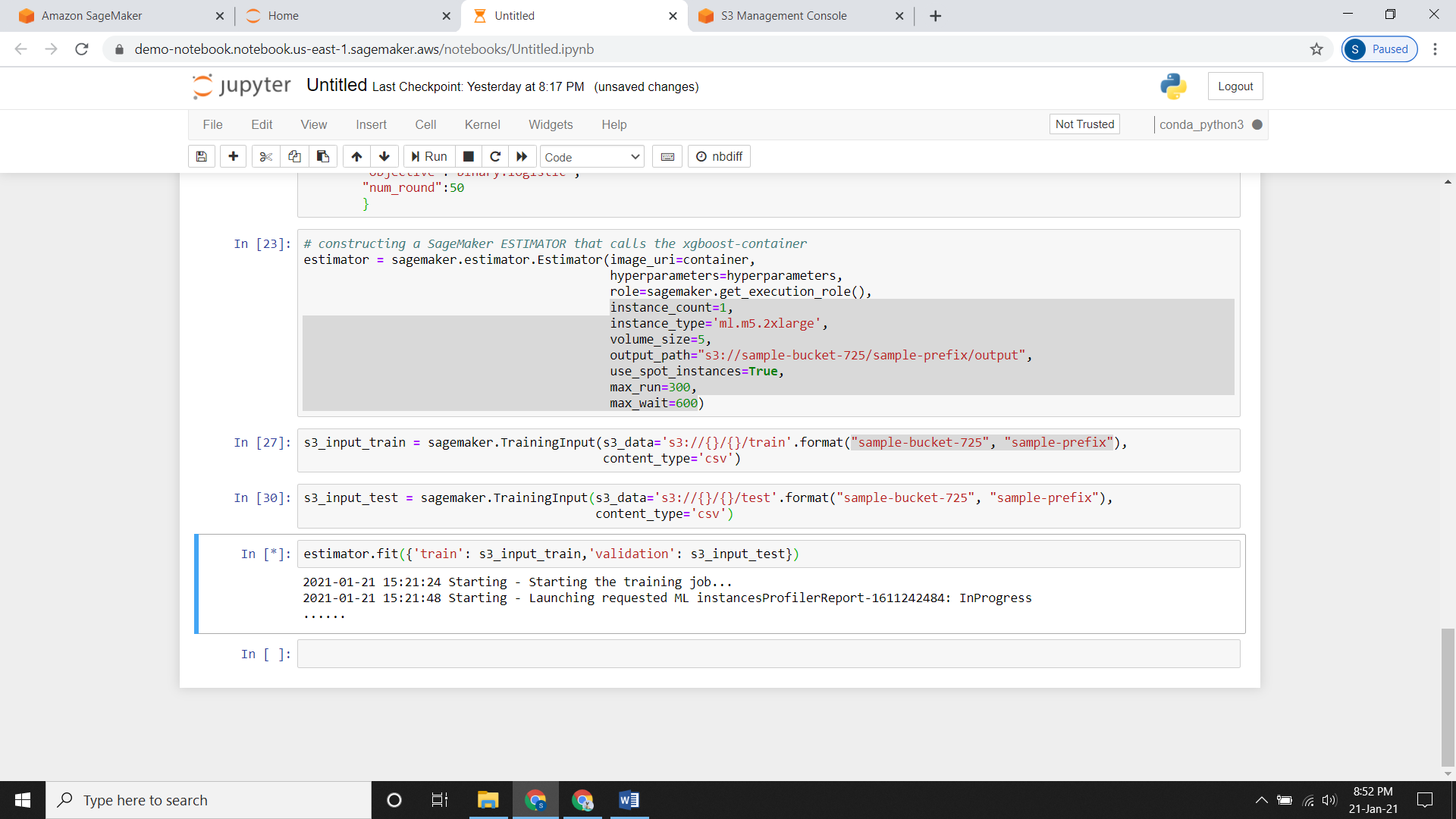
You just need to modify some commands over here like

train\_instance\_count => instance\_count, train\_instance\_type => instance\_type train\_volume\_size => volume\_size, train\_use\_spot\_instances => use\_spot\_instances, train\_max\_run => max\_run, train\_max\_wait => max\_wait

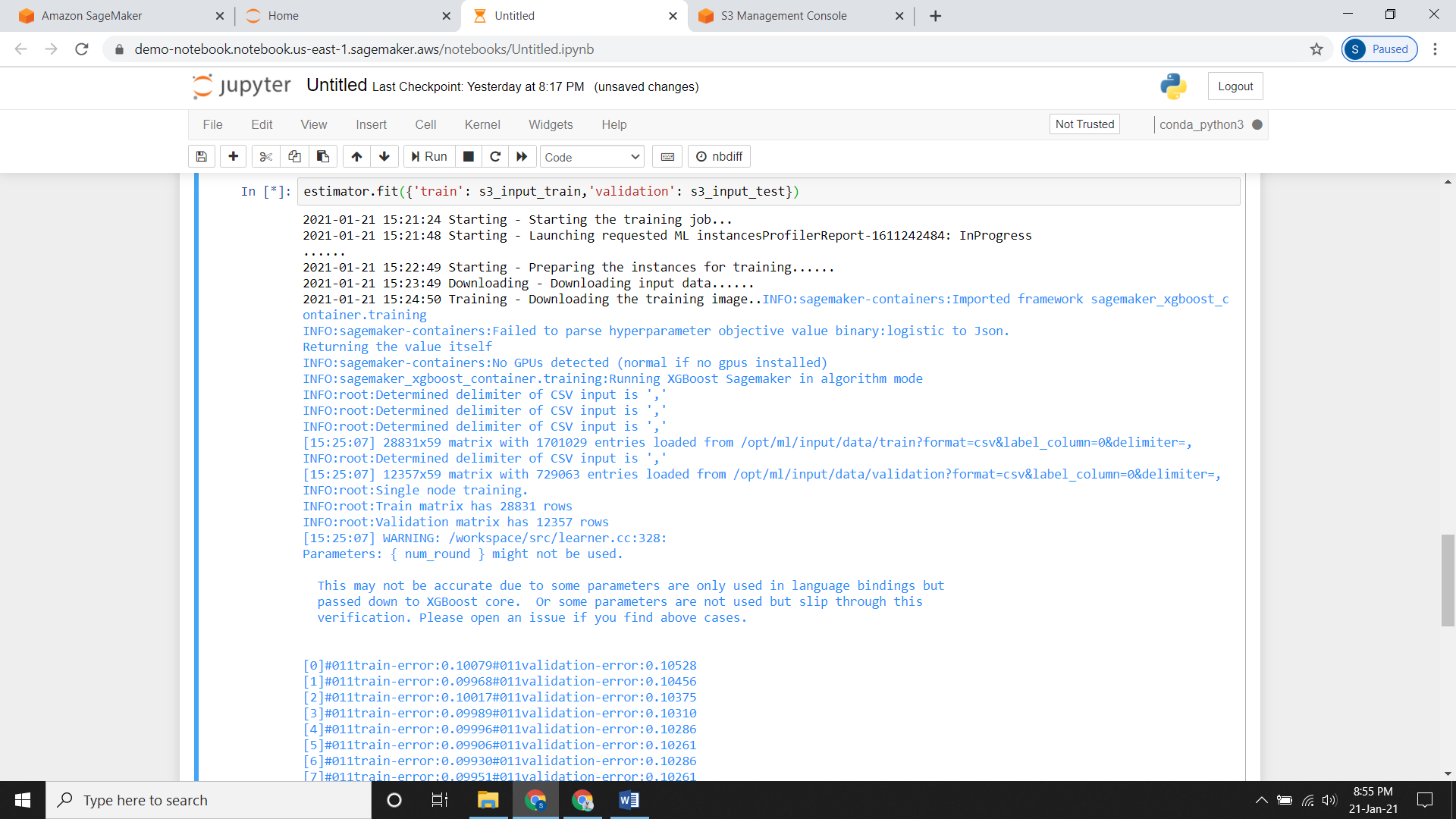


**Executing the Model:**

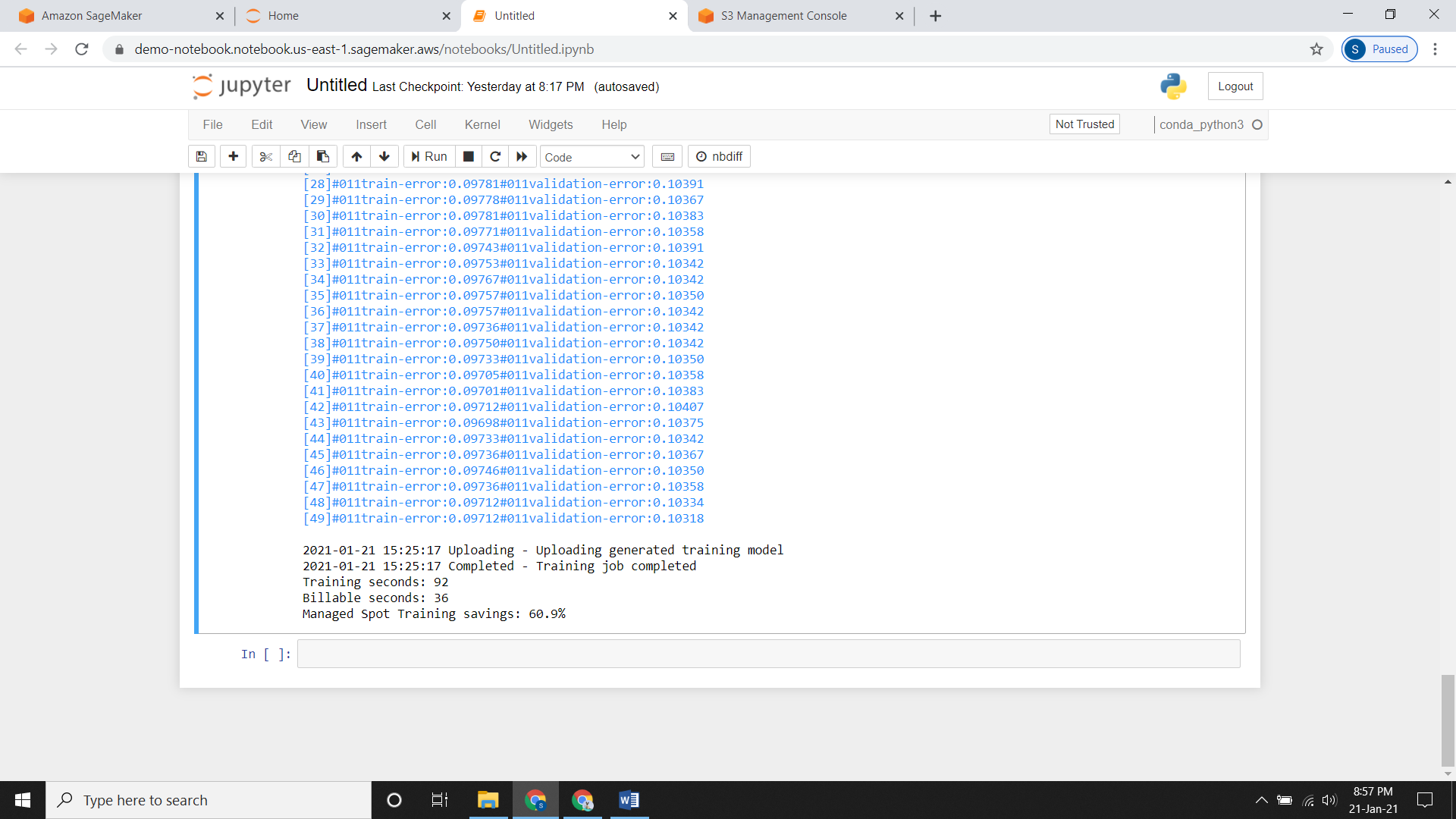
* Now for executing the model we use the command **estimator.fit**
* Here you need to specify the **train and validation** path.



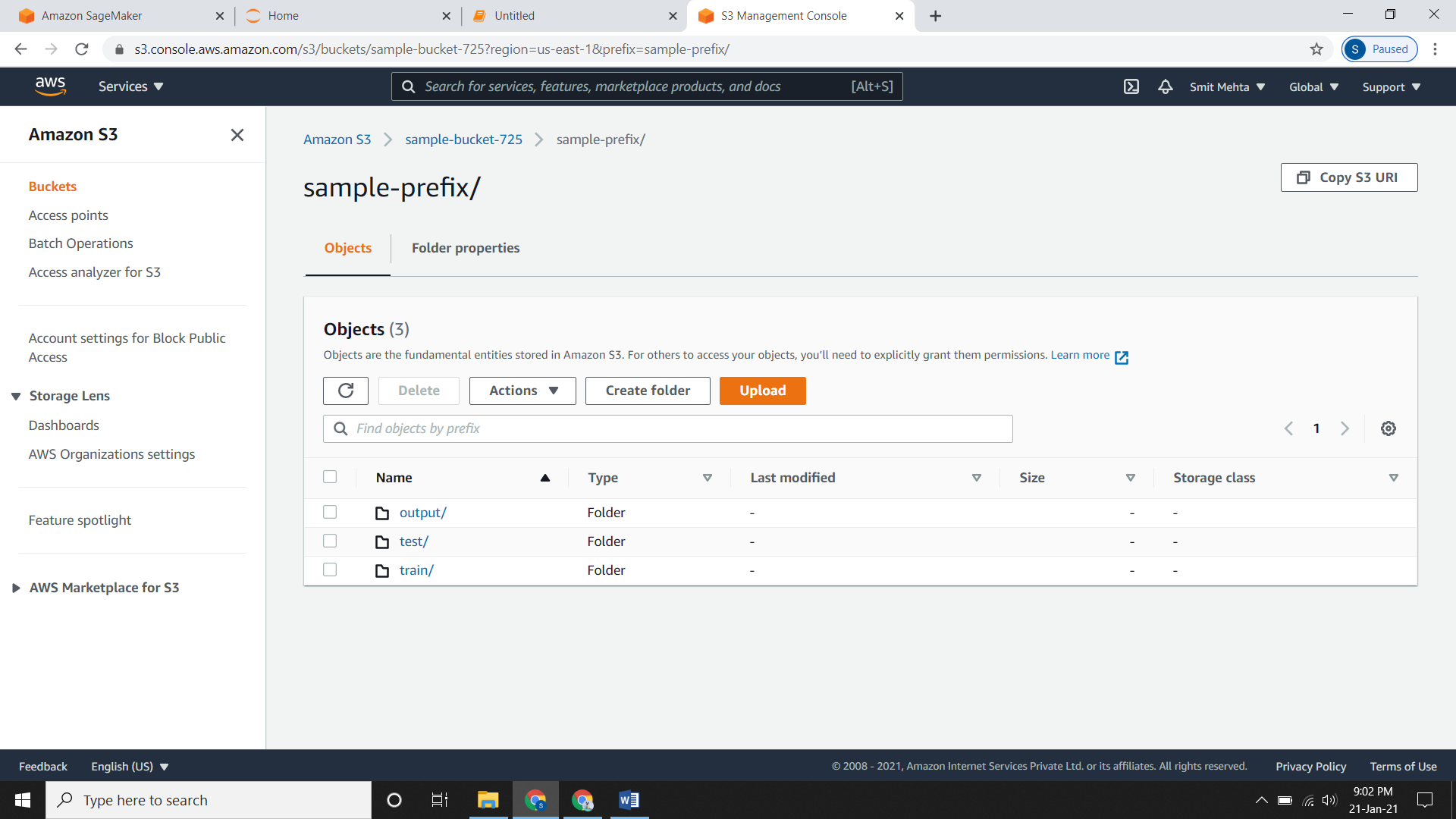
* It will take some time roughly around 2-3 minutes.





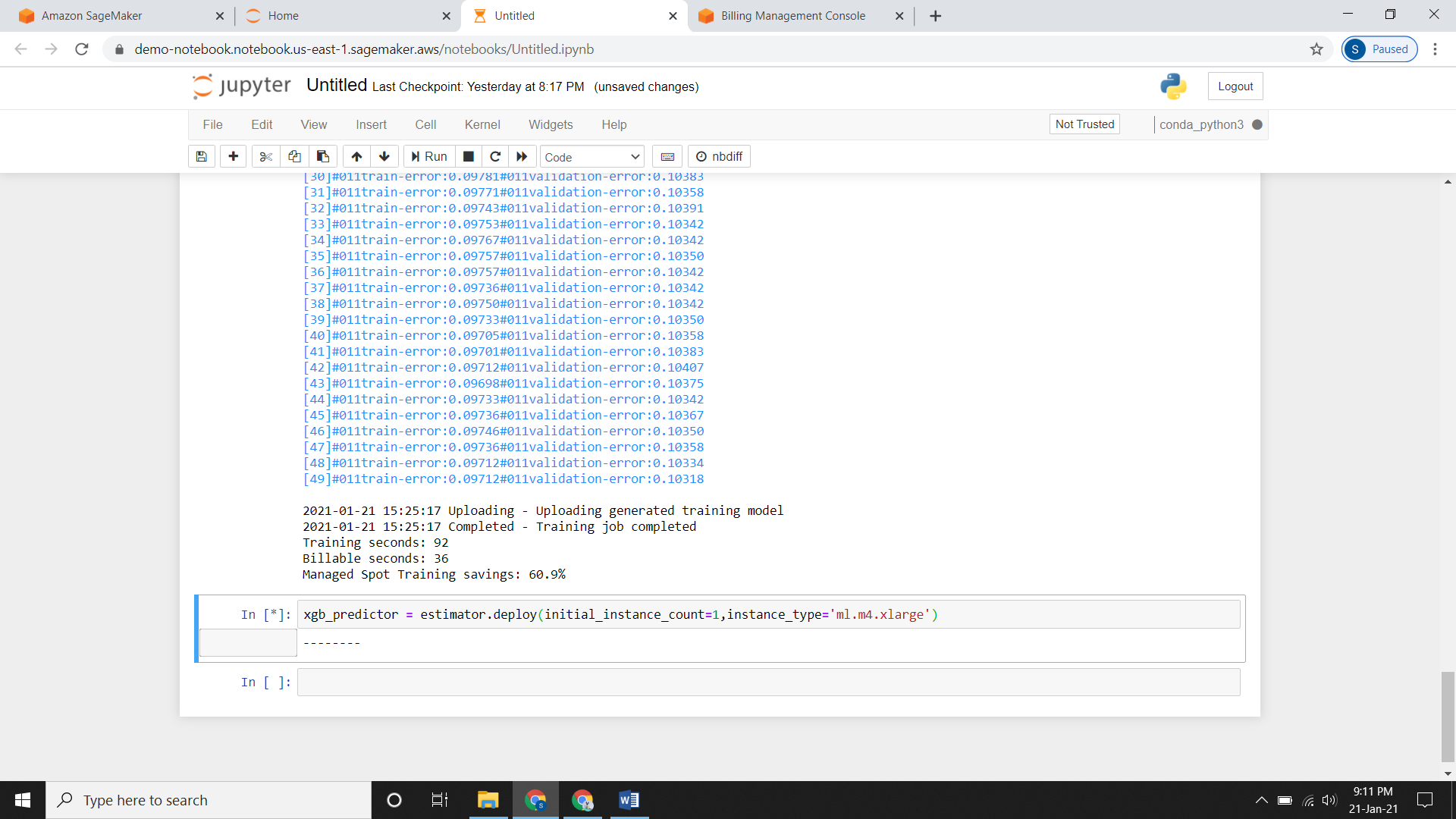


* Now since we have used the hyperparameters from here and max\_run & max\_wait commands the spot training saving is 60.9%
* So we have saved 60.9% amount in the billing section.
* Now if you check your S3 bucket it will contain your output file of this model.

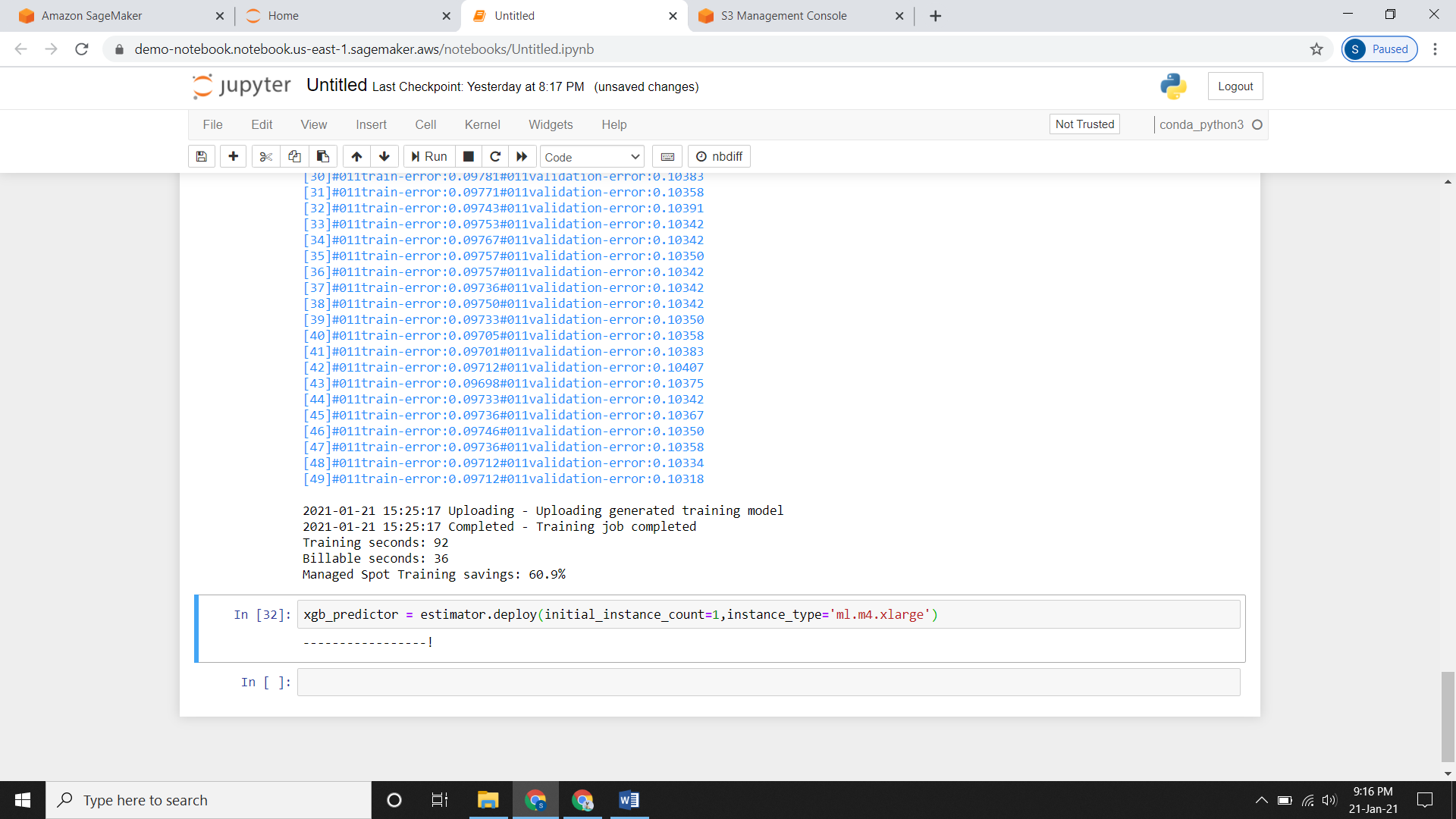


**Deploying your model:**

* Now for executing the model we use the command **estimator.deploy**
* Here you need to specify instance count as well as the type of instance.
* It will take some time to deploy the model.

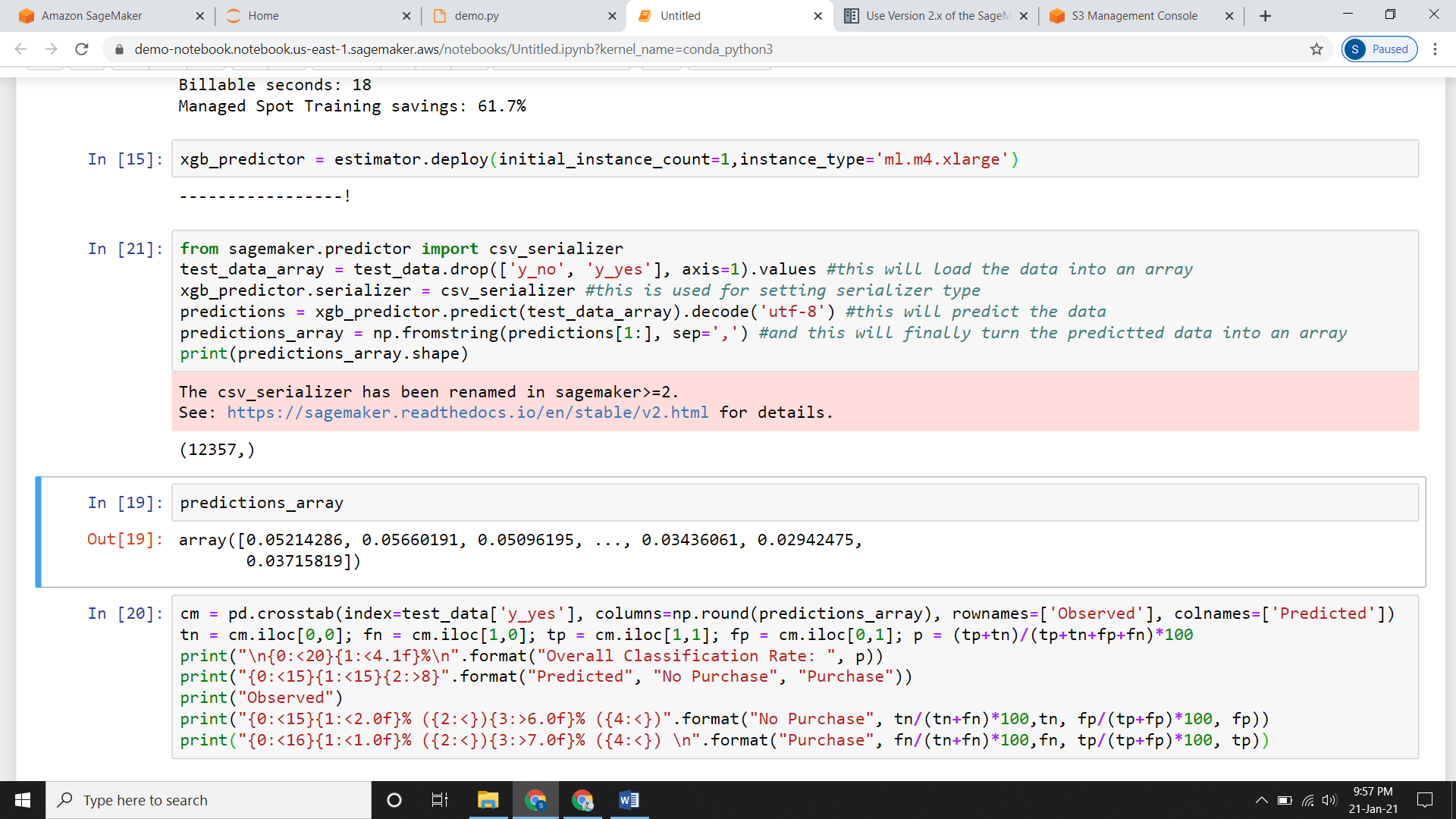


* As you see an exclamation command (**!)** that means your model is deployed.

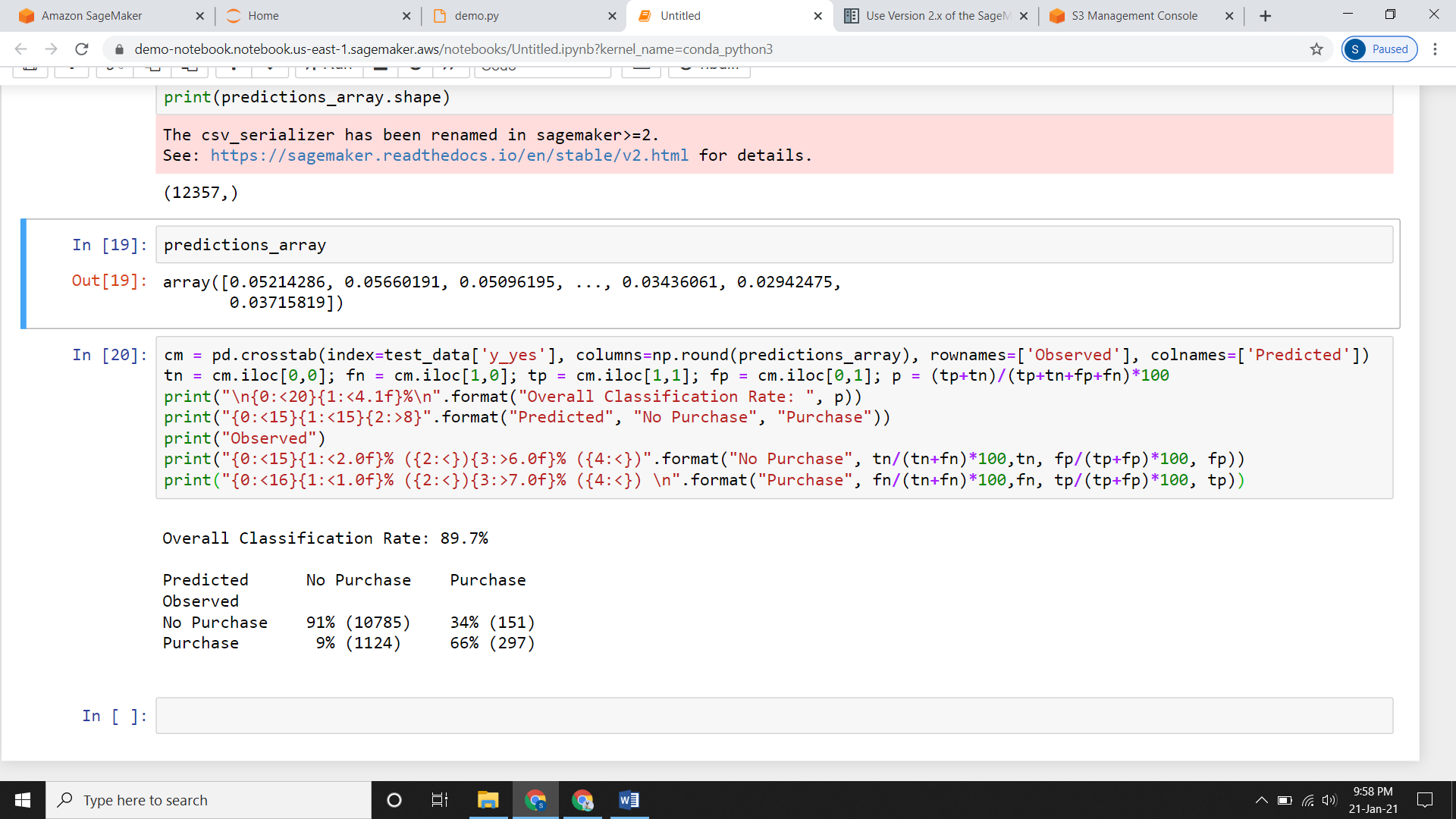


**Prediction of Data**

* After deploying you need to check that your data is in proper way and predict it so that it gets confirmed.

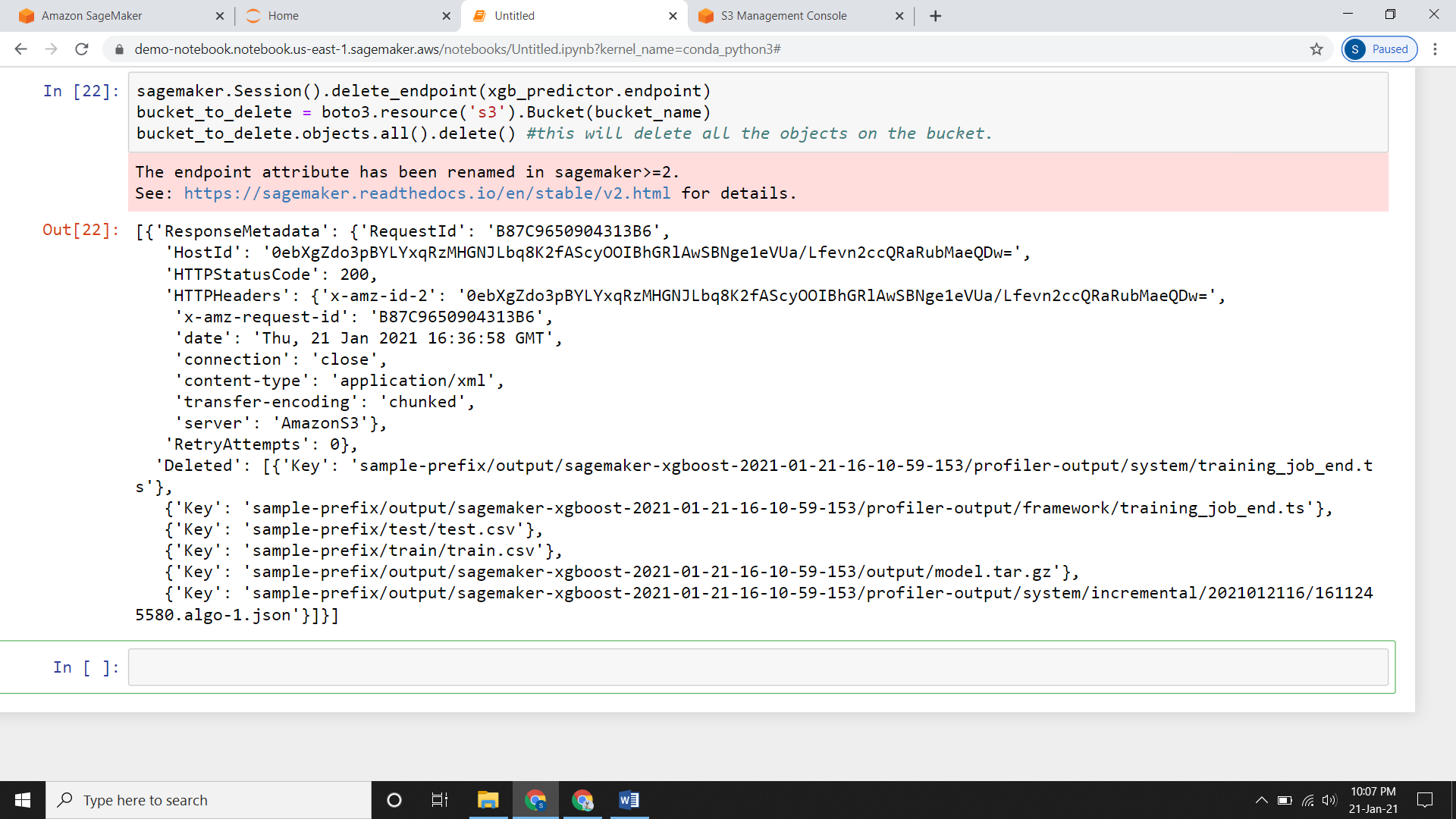


* Making a **confusion matrix (cm)** for the data.



**Deleting the Endpoint:**

* After you complete your task/practicing with SageMaker you need to delete the endpoint.
* If you do not delete it will increase your billing.



* Check your S3 bucket it will be empty.

