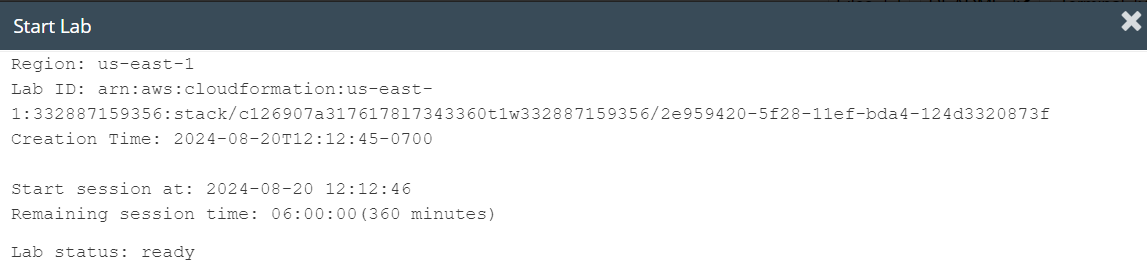
**CHALLENGE LAB**

Objective: Amazon SageMaker Challenge Lab: Predicting Airplane Delays

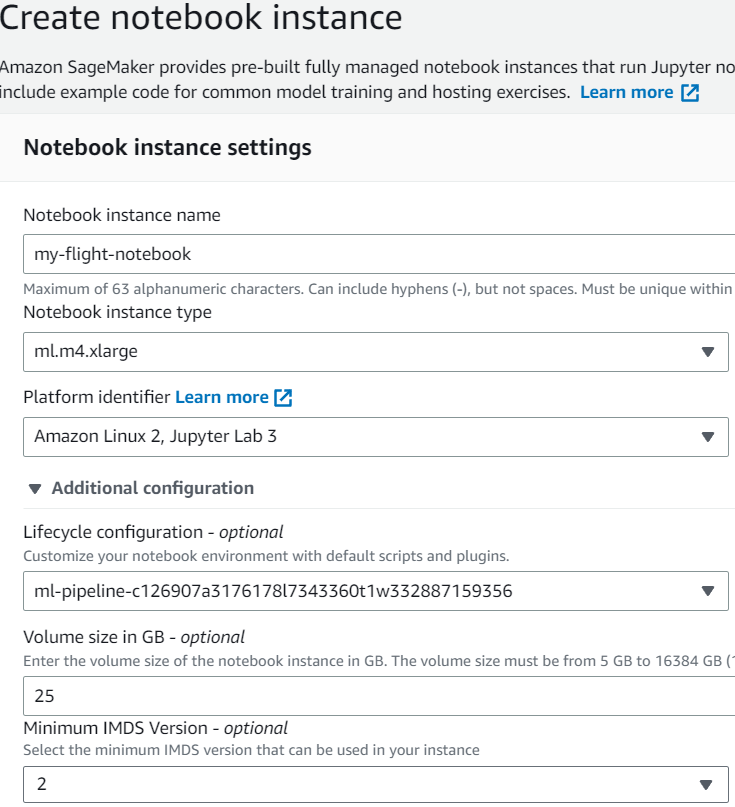
Procedure:

1. Accessing the AWS Management Console

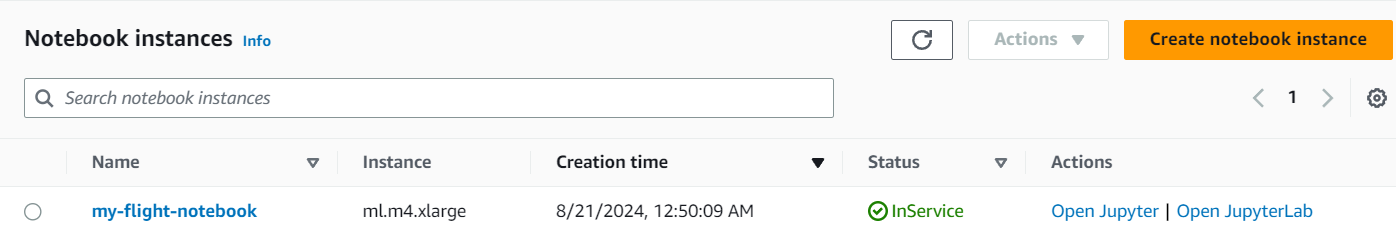


**Task 1: Creating a Jupyter notebook with Amazon SageMaker**

1. Access Amazon SageMaker in the AWS Console and create a Jupyter notebook*, my-flight-notebook*

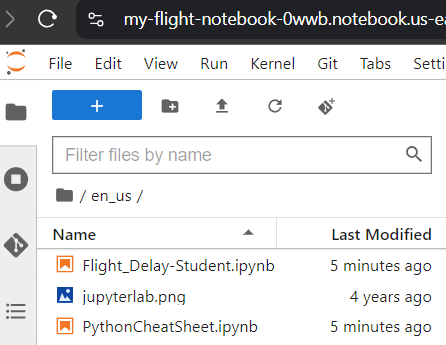


**Task 2: Opening your Jupyter notebook instance**

****

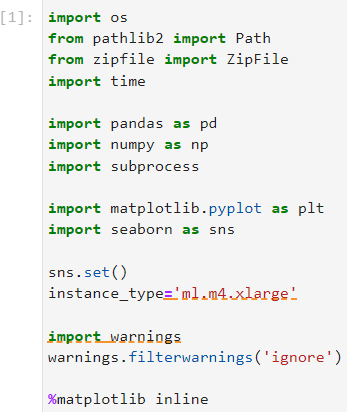
**Task 3: Loading the flight delay student notebook**

1. Open the en\_us/Flight\_Delay-Student.ipynb notebook by selecting it.



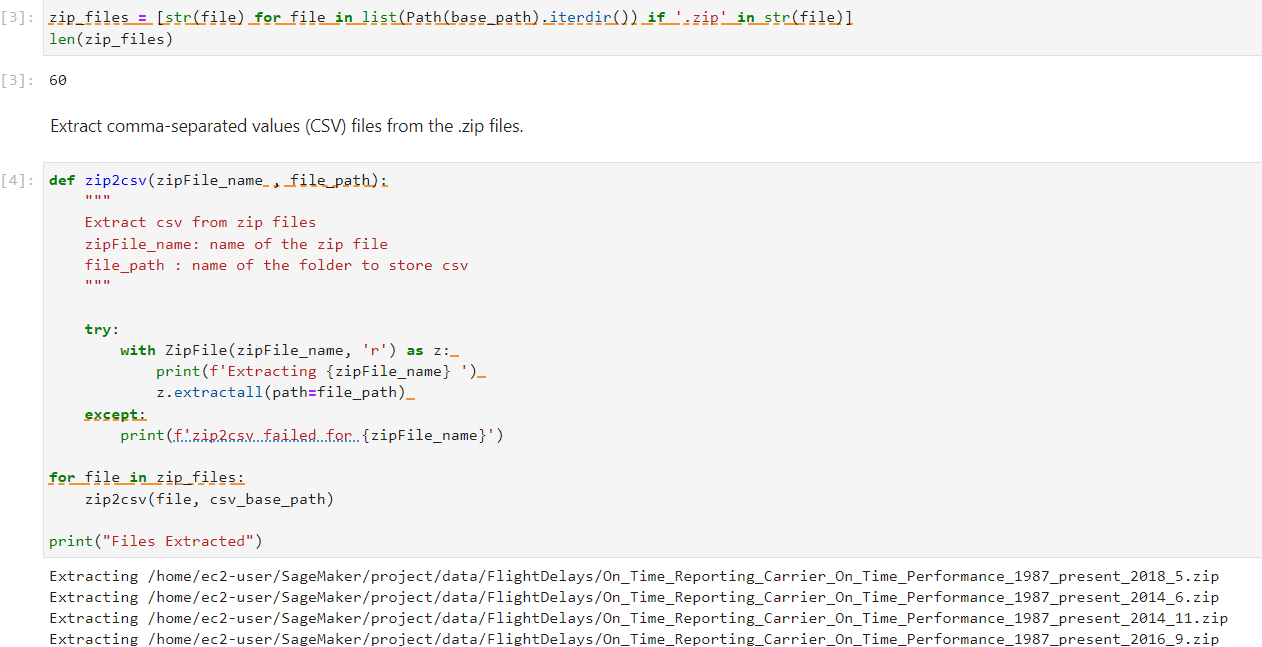
1. Follow the instructions in the notebook.

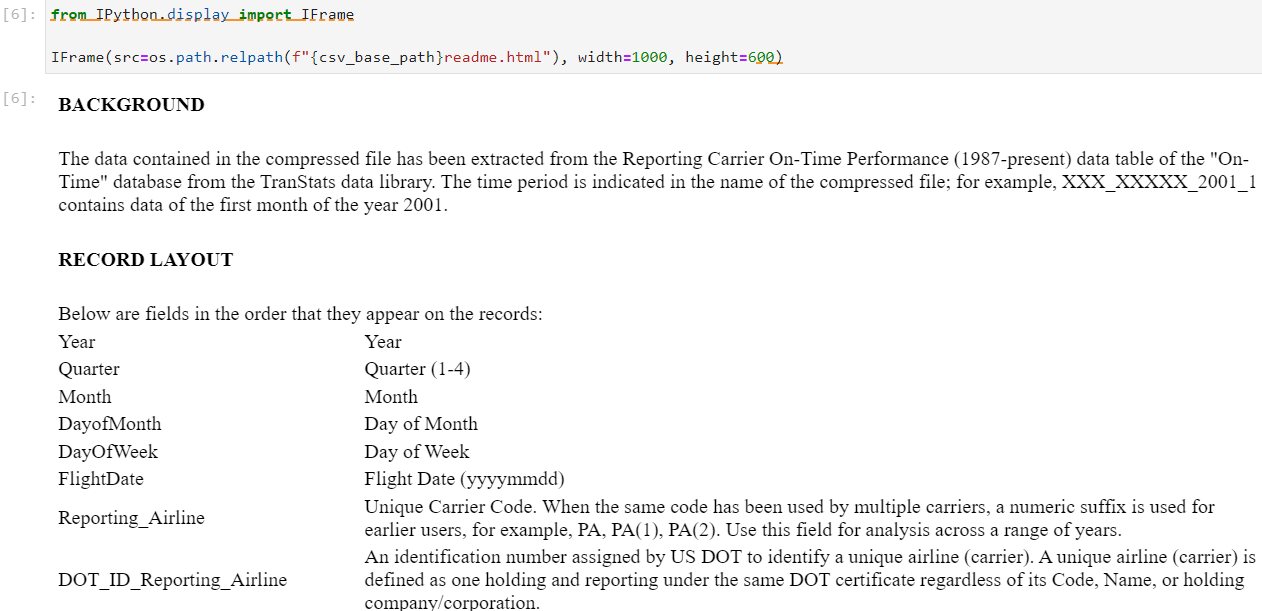
**Step 1: Problem formulation and data collection** (Setup)

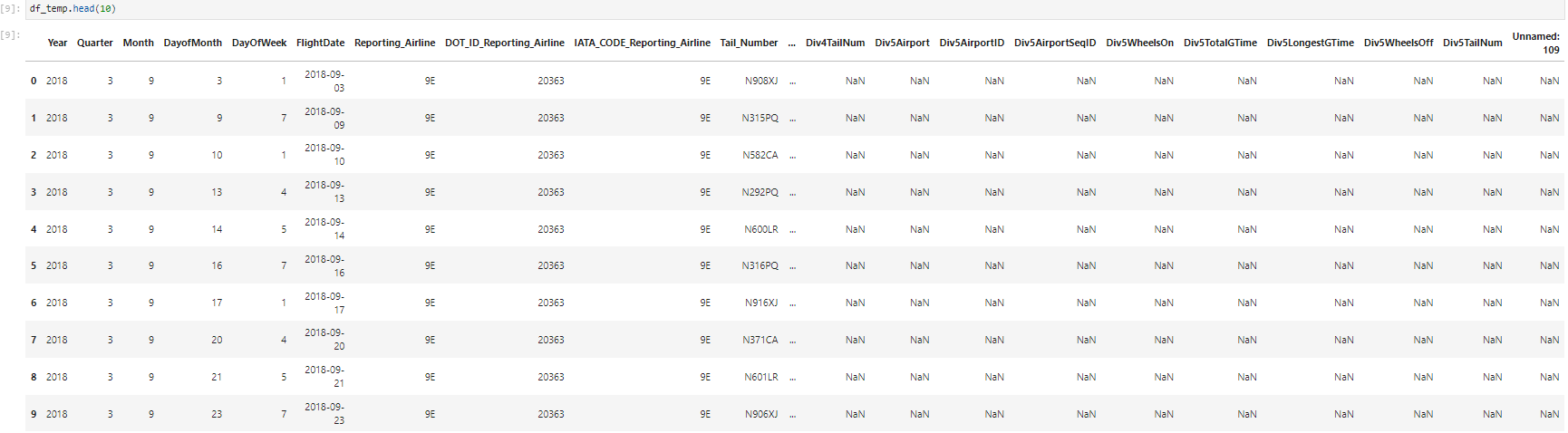
****

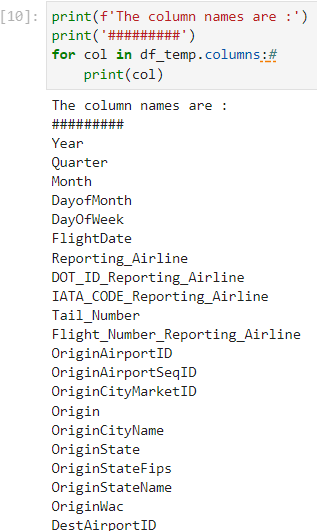
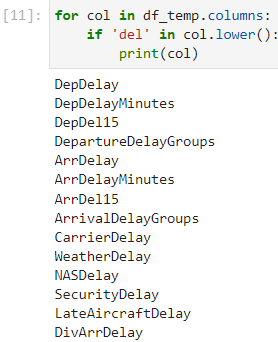
**Step 2: Data preprocessing and visualization**

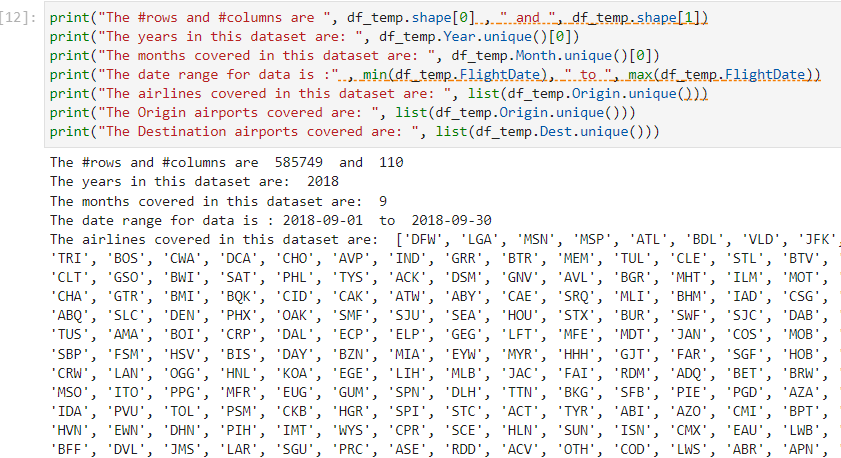
****

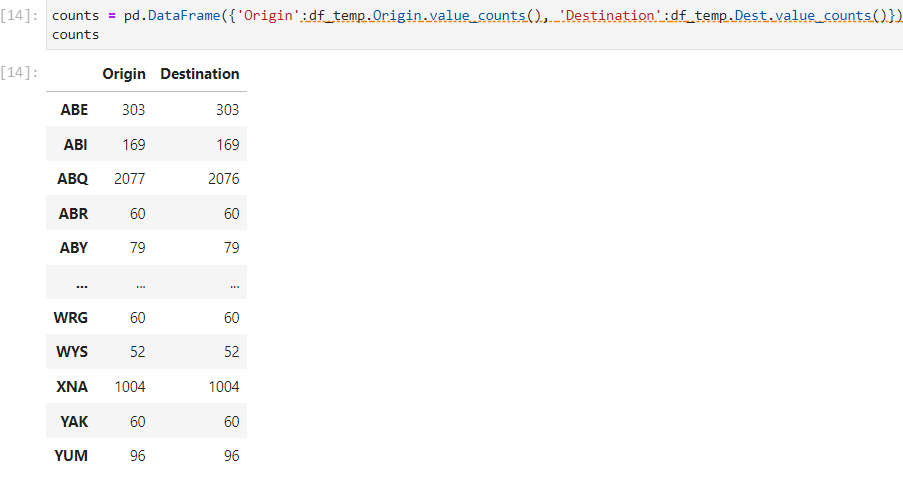
****

****

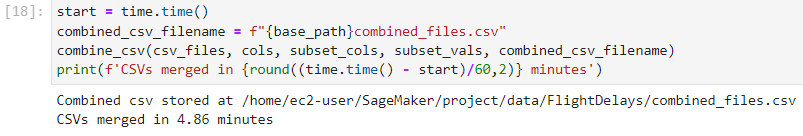
****

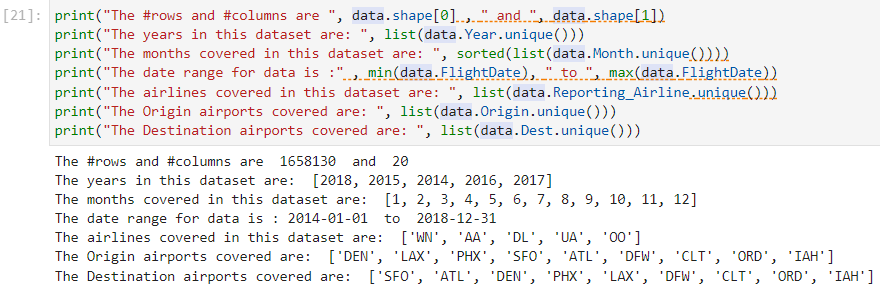
** **

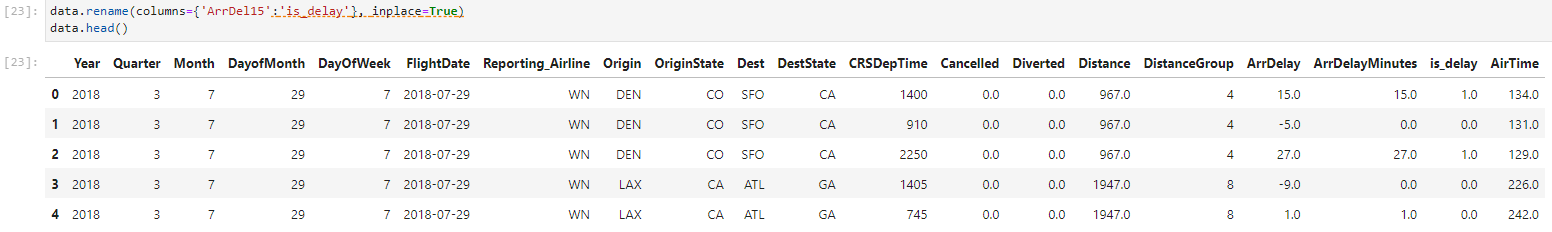
****

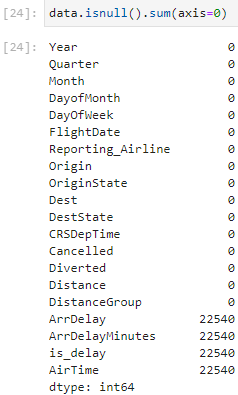
****

Combine all CSV files:

****

****

****

****

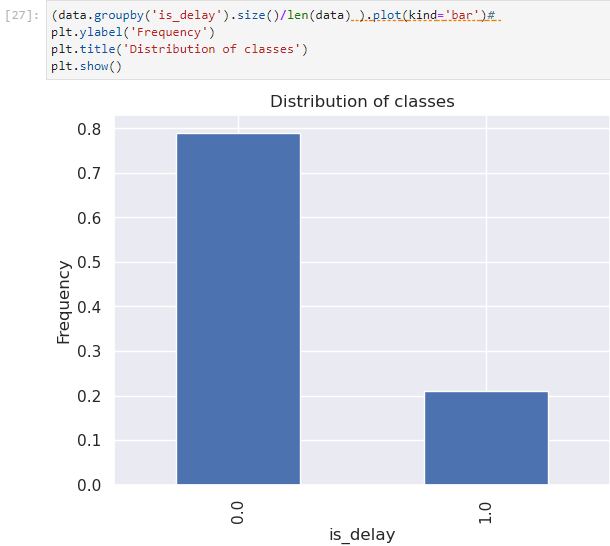
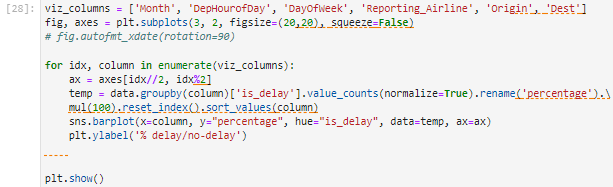
**The ML problem statement**

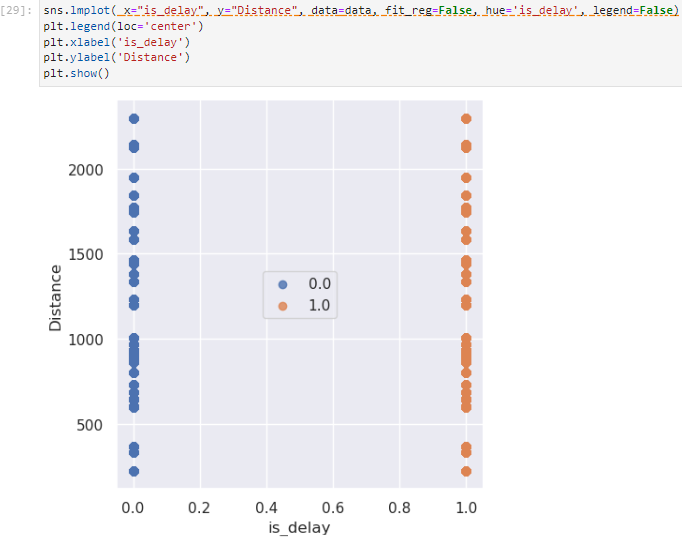
* Given a set of features, can you predict if a flight is going to be delayed more than 15 minutes?
* Because the target variable takes only a value of *0* or *1*, you could use a classification algorithm.

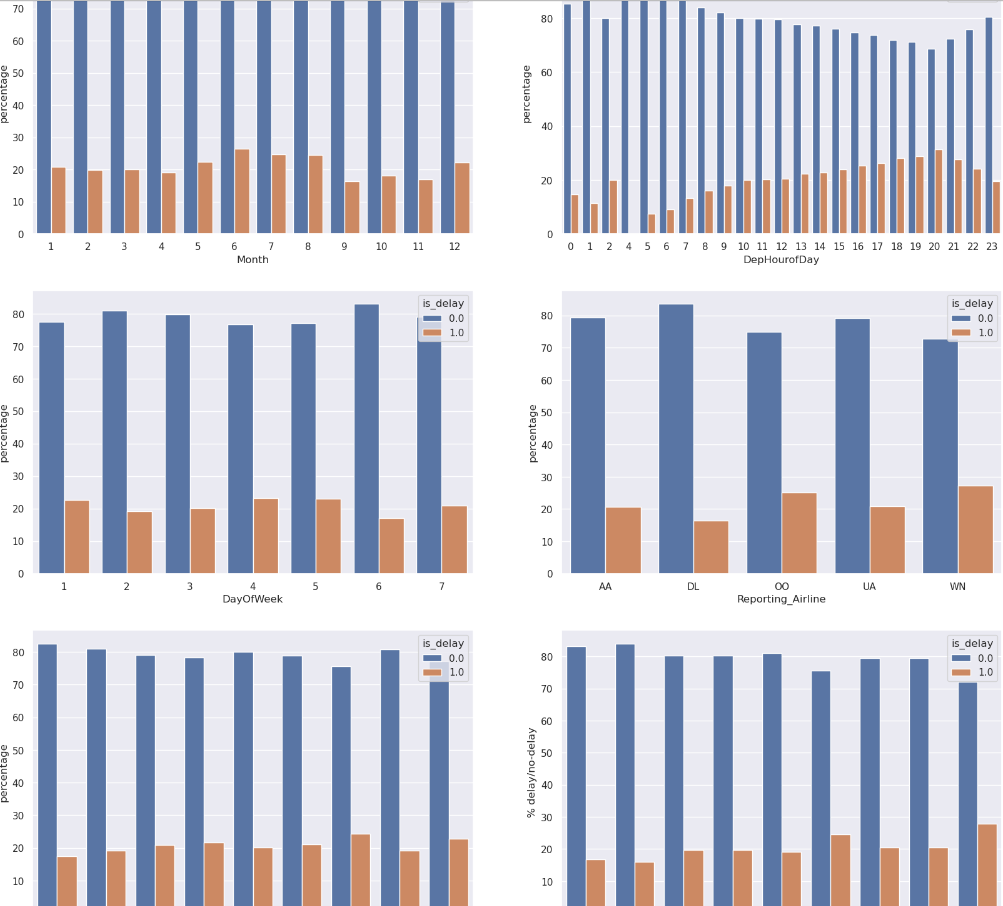
Before you start modeling, it's a good practice to look at feature distribution, correlations, and others.

* This will give you an idea of any non-linearity or patterns in the data
  + Linear models: Add power, exponential, or interaction features
  + Try a non-linear model
* Data imbalance
  + Choose metrics that won't give biased model performance (accuracy versus the area under the curve, or AUC)
  + Use weighted or custom loss functions
* Missing data
  + Do imputation based on simple statistics -- mean, median, mode (numerical variables), frequent class (categorical variables)
  + Clustering-based imputation (k-nearest neighbors, or KNNs, to predict column value)
  + Drop column

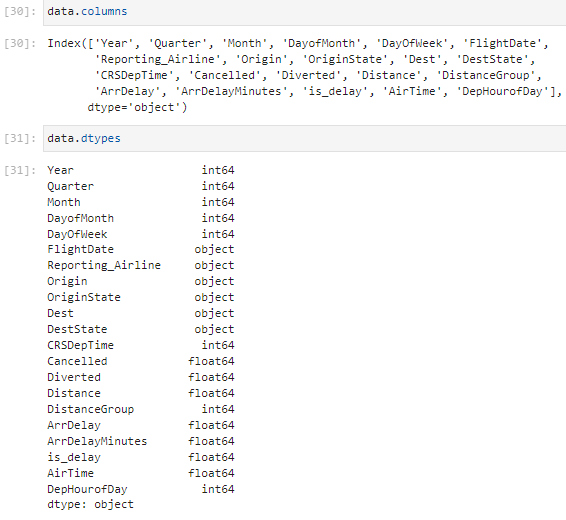
**Data exploration**

 ****

****

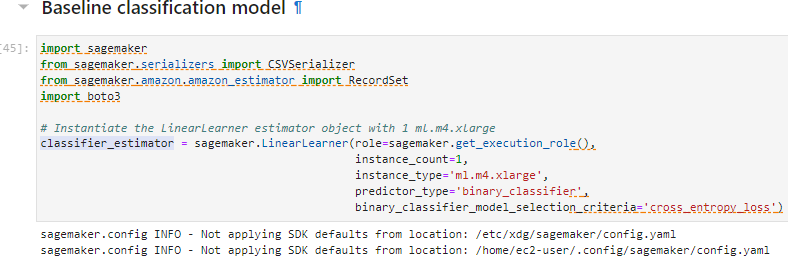
****

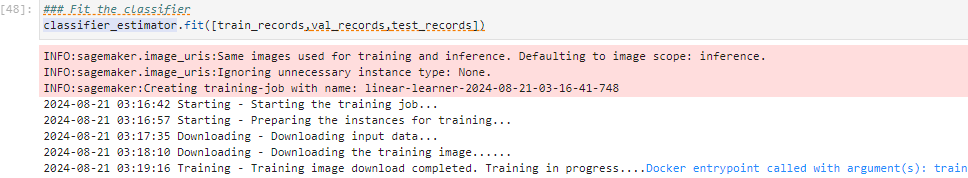
**Features**

****

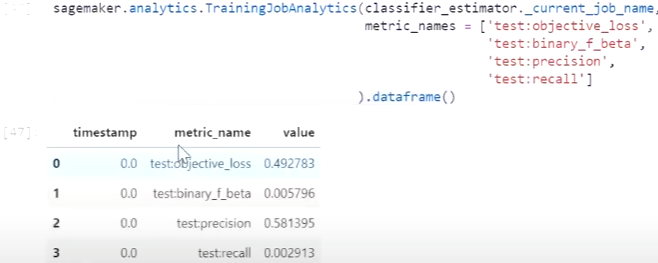
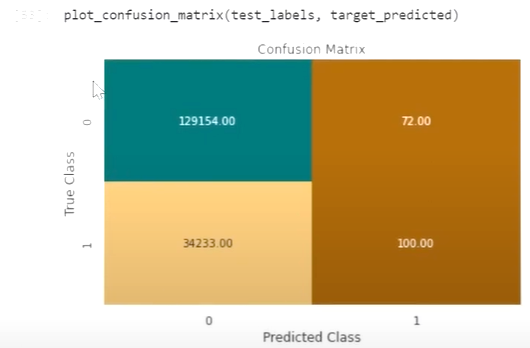
**Step 3: Model training and evaluation**

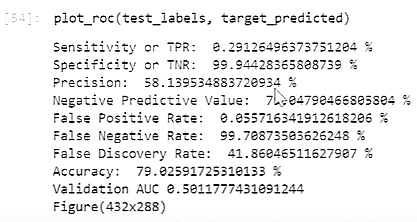
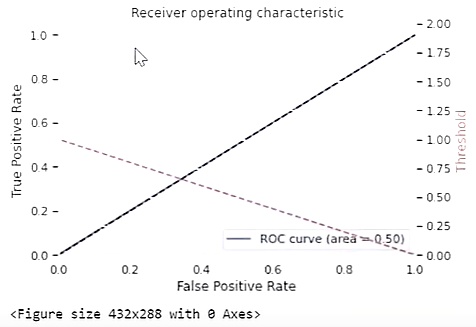
****

****

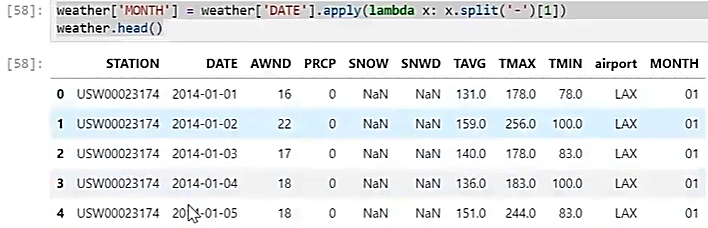
****

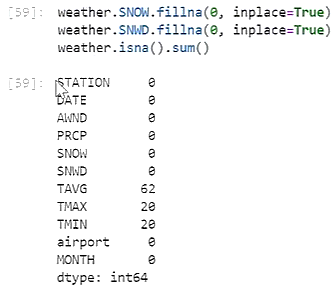
**Model evaluation**

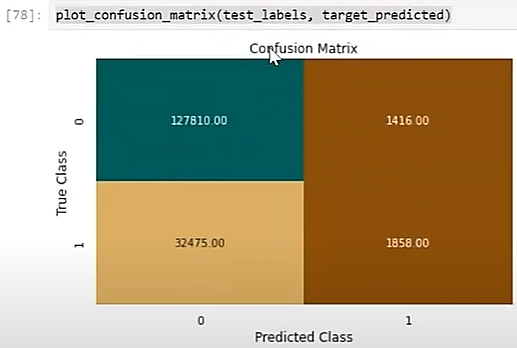
** **

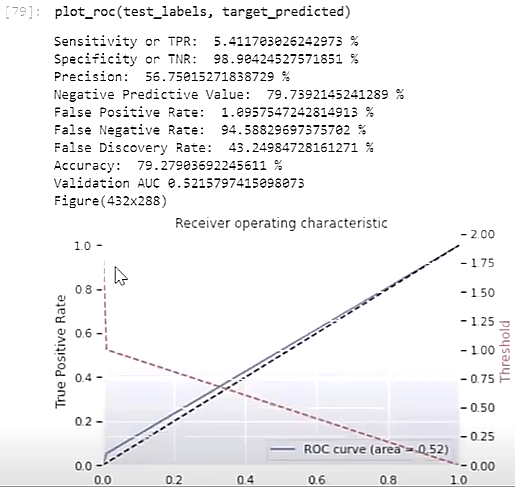
** **

**Step 4: Feature Engineering**

****

****

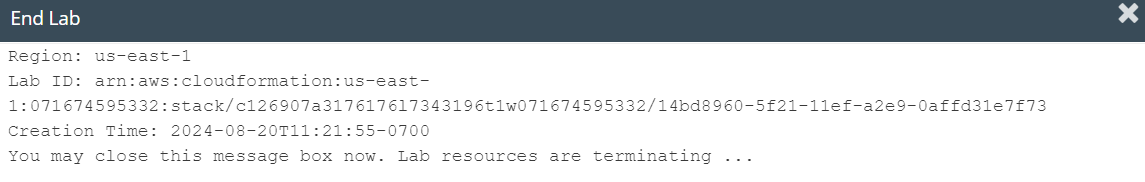
****

****

**Conclusion**

* Processed and created a dataset from downloaded .zip files
* Performed an exploratory data analysis (EDA)
* Established a baseline model
* Performed hyperparameter optimization
* Used metrics to compare the performance between models

**Lab complete –** In order to end the lab, choose End Lab, and then choose **Yes**.

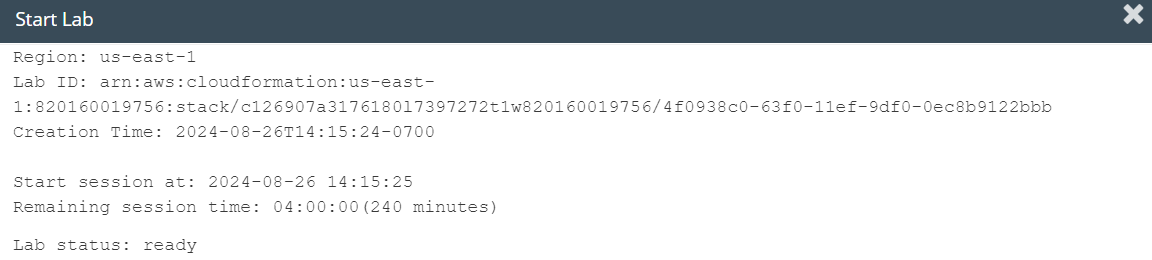


**LAB – 8**

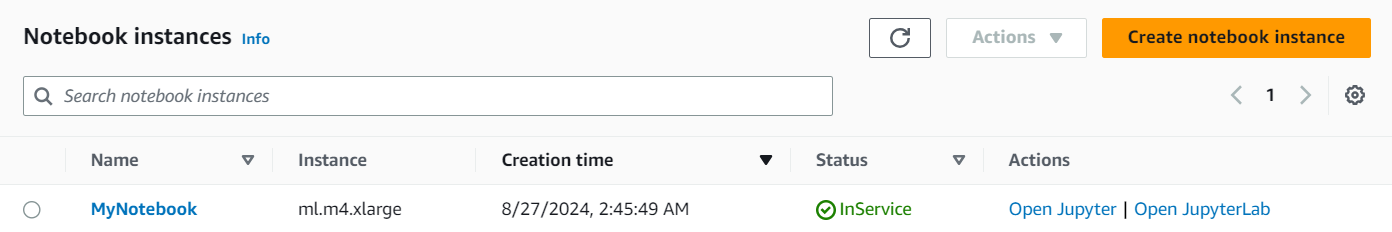
Objective: Creating a forecast with Amazon Forecast

Procedure:

1. Accessing the AWS Management Console

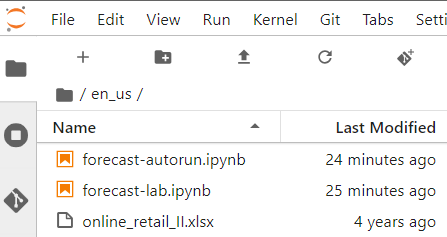


**Task 1: Accessing a notebook instance in Amazon SageMaker**



**Task 2: Opening a notebook in your notebook instance**

1. In your JupyterLab environment, go to the file browser in the left pane and locate the *forecast-lab.ipynb* file & open it.

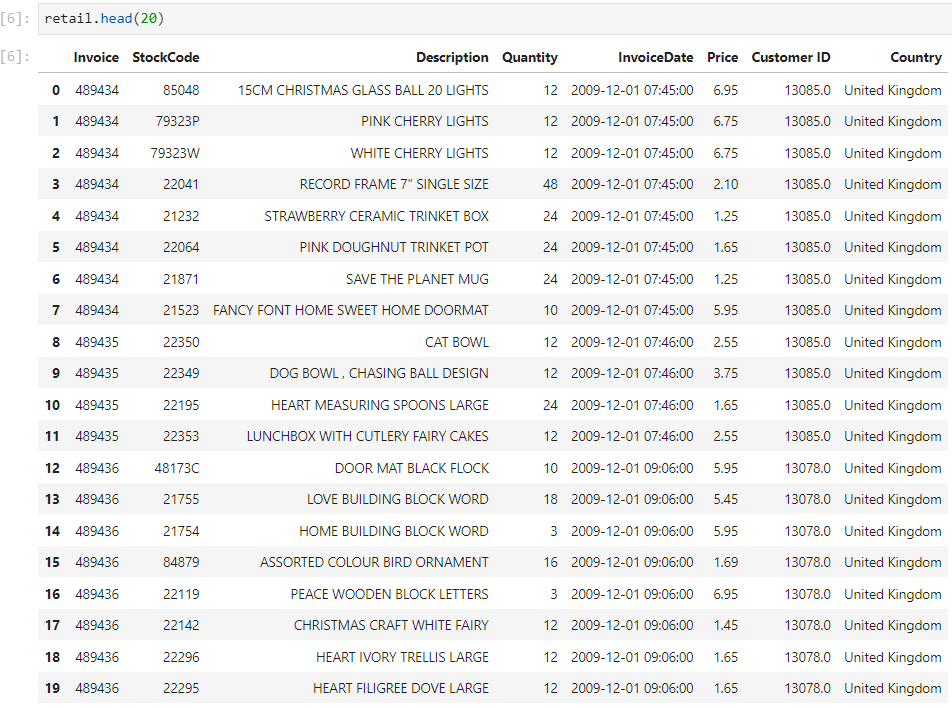


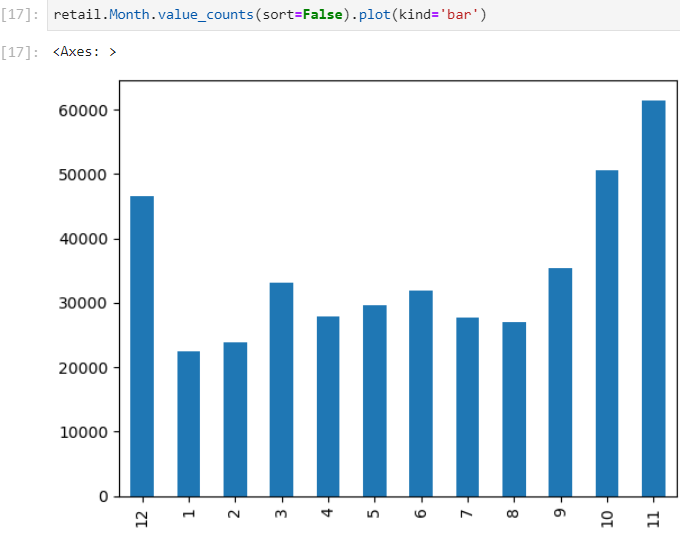
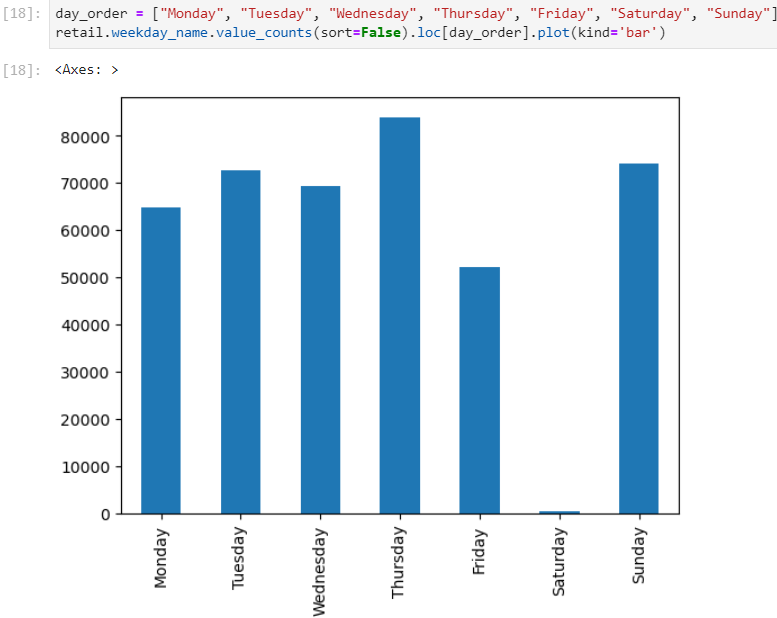
1. Follow the instructions in the notebook

**Task 1: Importing Python packages**

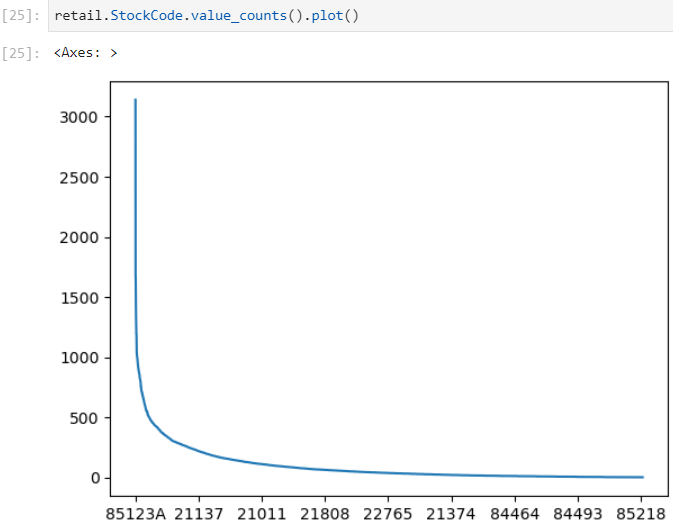
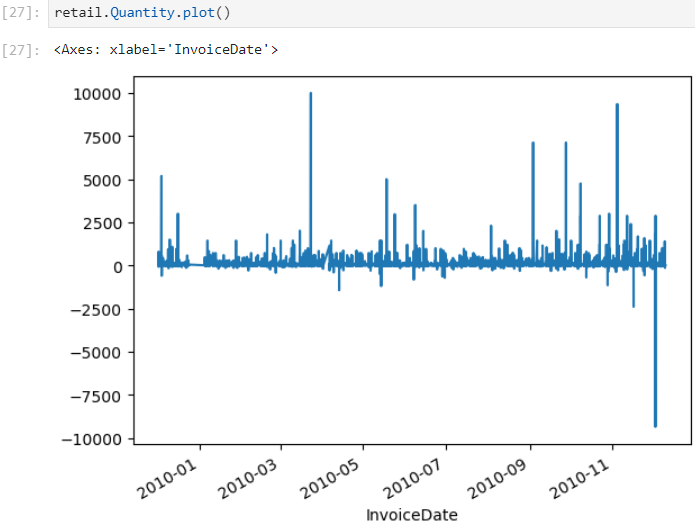
****

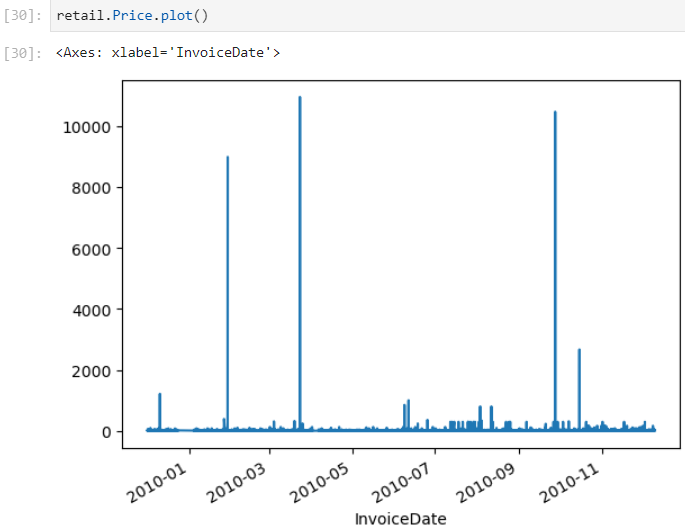
**Task 2: Exploring the data**

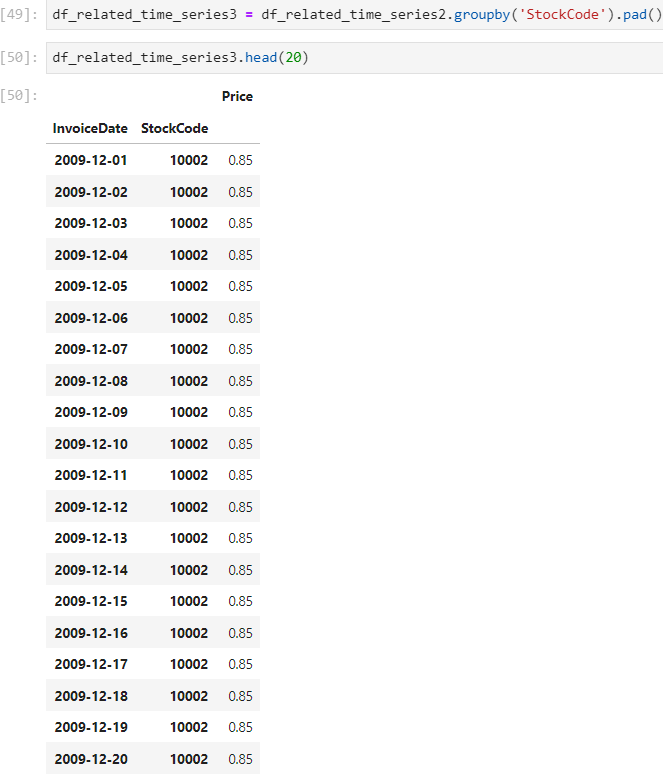
****

** **

**Task 3: Cleaning and reducing the size of the data**

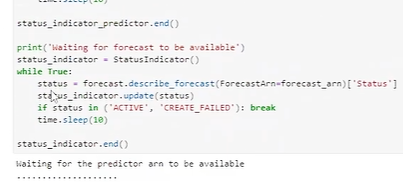
** **

****

****

**Task 4: Reviewing the creation of the forecast**

**Task 5: Waiting for the forecast creation to complete**

****

**Task 6: Using the forecast**

**Task 7: Cleaning up**

****

**Conclusion**

* Used pandas to prepare time series data
* Used Amazon Forecast to create a predictor
* Used an Amazon Forecast predictor to create a forecast
* Compared a forecast against test data

**Lab complete**

1. In order to end the lab, choose End Lab, and then choose **Yes**.

