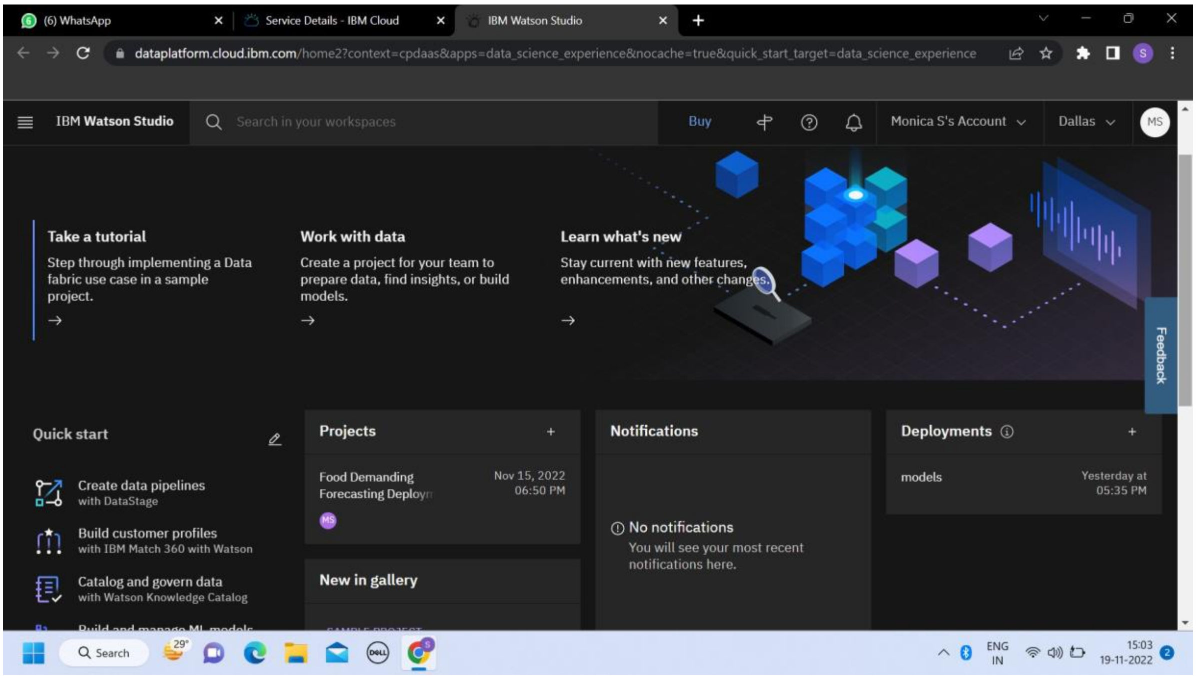
## TEAM ID: PNT2022TMID21777
## PROJECT NAME: DemandEst - AI powered Food Demand Forecaster


(6) WhatsApp
Service Details - IBM Cloud
IBM Watson Studio
dataplatform.cloud.ibm.com/home2?context=cpdaas&apps=data_science_experience&nocache=true&quick_start_target=data_science_experience
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Feedback
Search
29°
ENG
IN
15:03
19-11-2022

**Take a tutorial**

Step through implementing a Data fabric use case in a sample project.

**Work with data**

Create a project for your team to prepare data, find insights, or build models.

**Learn what's new**

Stay current with new features, enhancements, and other changes.

**Quick start**

Create data pipelines
with DataStage

Build customer profiles
with IBM Match 360 with Watson

Catalog and govern data
with Watson Knowledge Catalog

Build and manage ML models

**Projects**

Food Demanding Forecasting Deploym — Nov 15, 2022 06:50 PM

**New in gallery**

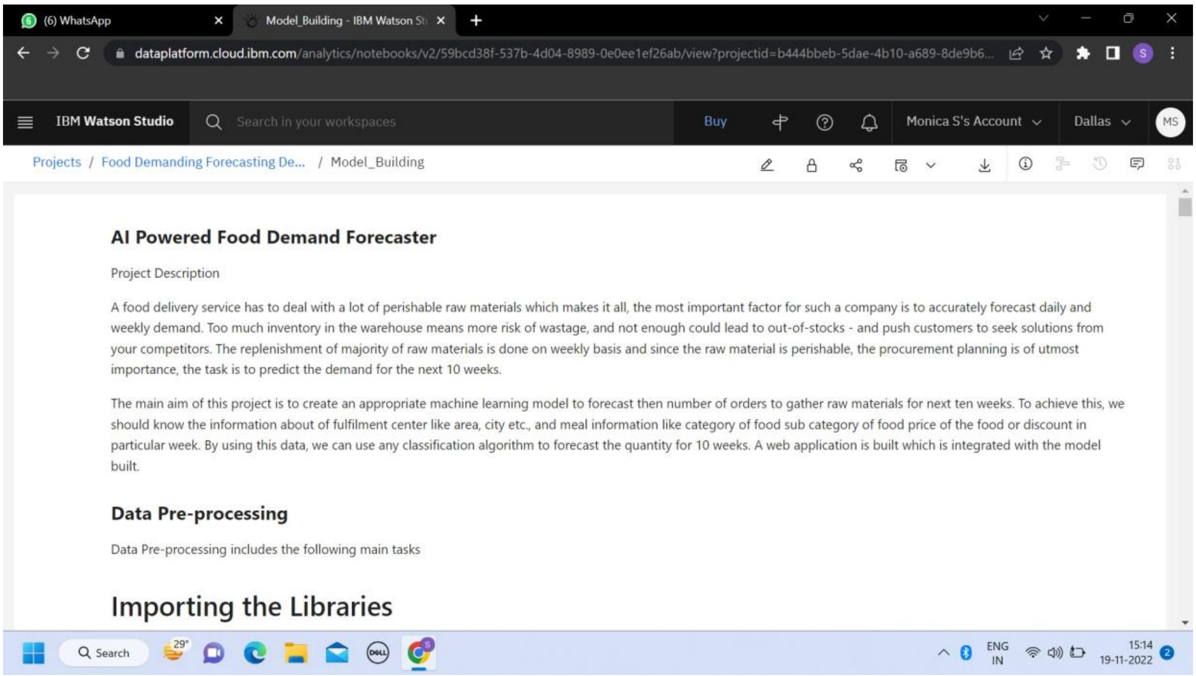
SAMPLE PROJECT

**Notifications**

No notifications
You will see your most recent notifications here.

**Deployments**

models — Yesterday at 05:35 PM


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:14
19-11-2022

### AI Powered Food Demand Forecaster

Project Description

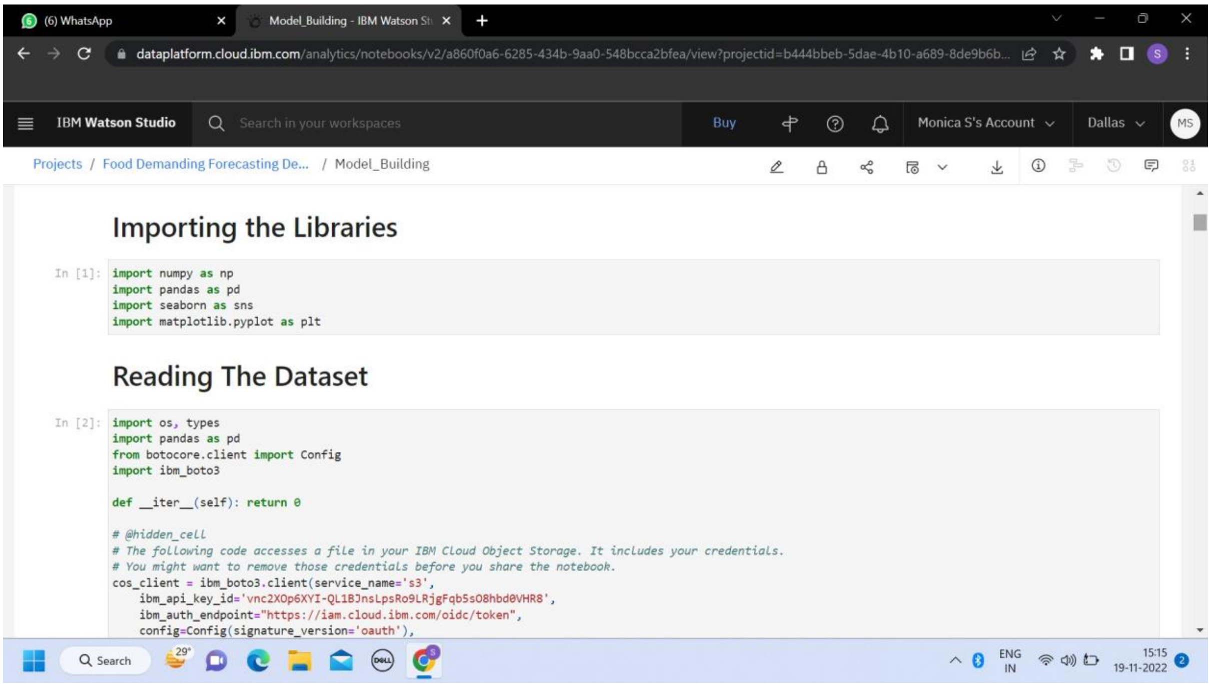
A food delivery service has to deal with a lot of perishable raw materials which makes it all, the most important factor for such a company is to accurately forecast daily and weekly demand. Too much inventory in the warehouse means more risk of wastage, and not enough could lead to out-of-stocks - and push customers to seek solutions from your competitors. The replenishment of majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance, the task is to predict the demand for the next 10 weeks.

The main aim of this project is to create an appropriate machine learning model to forecast then number of orders to gather raw materials for next ten weeks. To achieve this, we should know the information about of fulfilment center like area, city etc., and meal information like category of food sub category of food price of the food or discount in particular week. By using this data, we can use any classification algorithm to forecast the quantity for 10 weeks. A web application is built which is integrated with the model built.

### Data Pre-processing

Data Pre-processing includes the following main tasks

## Importing the Libraries


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

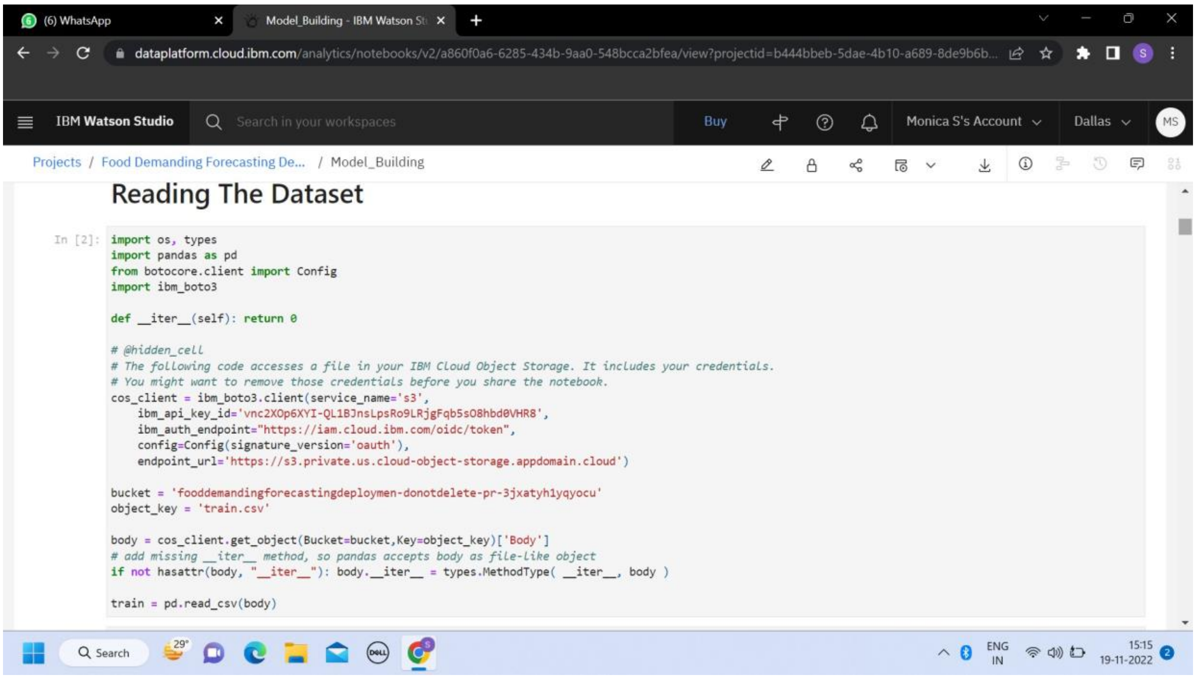
## Importing the Libraries

```
In [1]: import numpy as np
        import pandas as pd
        import seaborn as sns
        import matplotlib.pyplot as plt
```

## Reading The Dataset

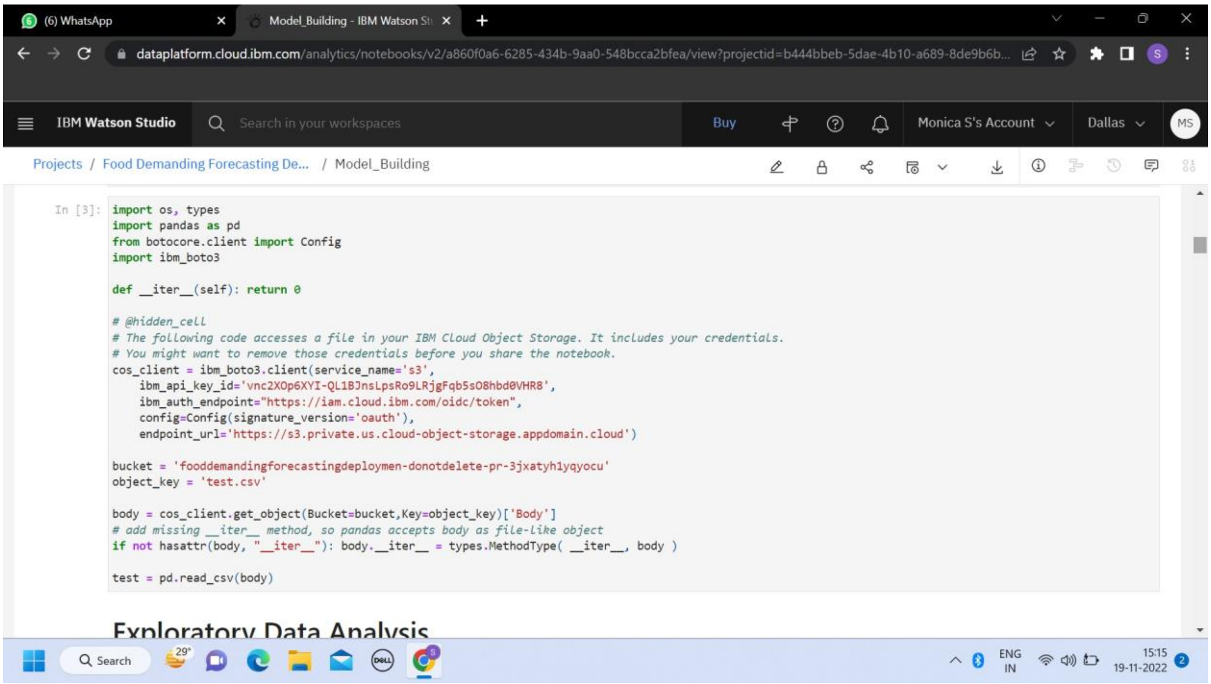
```
In [2]: import os, types
        import pandas as pd
        from botocore.client import Config
        import ibm_boto3

        def __iter__(self): return 0

        # @hidden_cell
        # The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
        # You might want to remove those credentials before you share the notebook.
        cos_client = ibm_boto3.client(service_name='s3',
            ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
            ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
            config=Config(signature_version='oauth'),
```

Search
29°
ENG
IN
15:15
19-11-2022


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

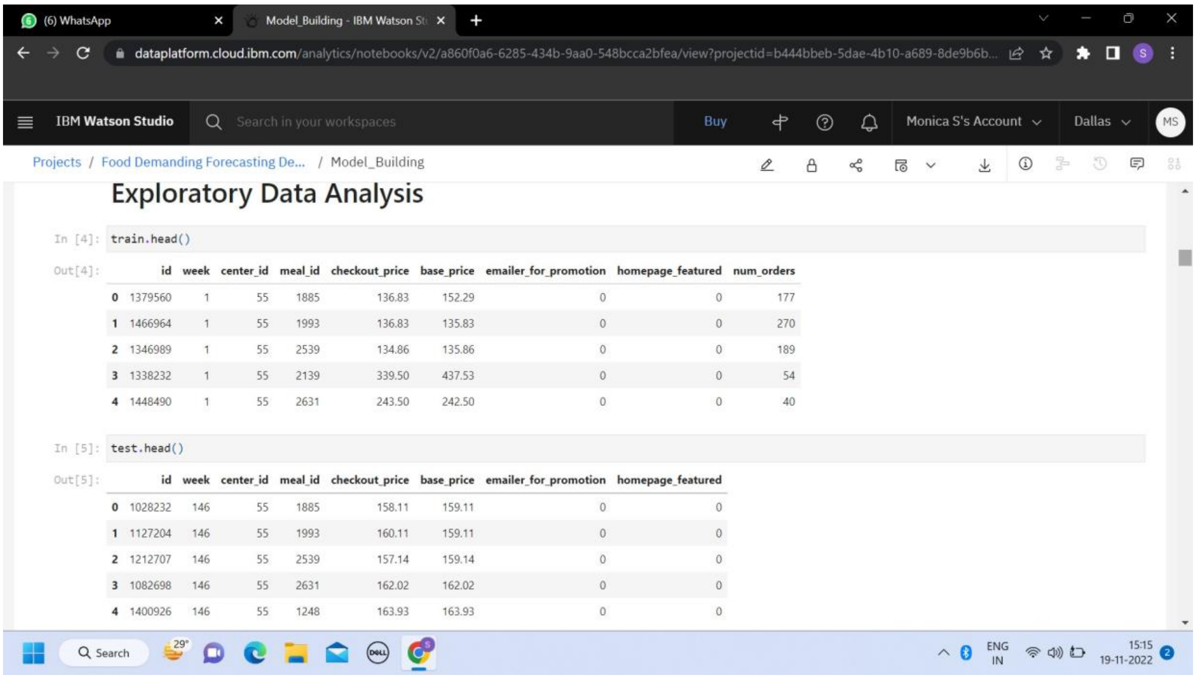
## Reading The Dataset

```
In [2]: import os, types
        import pandas as pd
        from botocore.client import Config
        import ibm_boto3

        def __iter__(self): return 0

        # @hidden_cell
        # The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
        # You might want to remove those credentials before you share the notebook.
        cos_client = ibm_boto3.client(service_name='s3',
            ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
            ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
            config=Config(signature_version='oauth'),
            endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

        bucket = 'fooddemandingforecastingdeploymen-donotdelete-pr-3jxatyh1yqyocu'
        object_key = 'train.csv'

        body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
        # add missing __iter__ method, so pandas accepts body as file-like object
        if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

        train = pd.read_csv(body)
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

In [3]:

```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'fooddemandingforecastingdeploymen-donotdelete-pr-3jxatyh1yqyocu'
object_key = 'test.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

test = pd.read_csv(body)
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

## Exploratory Data Analysis

In [4]:

```
train.head()
```

Out[4]:

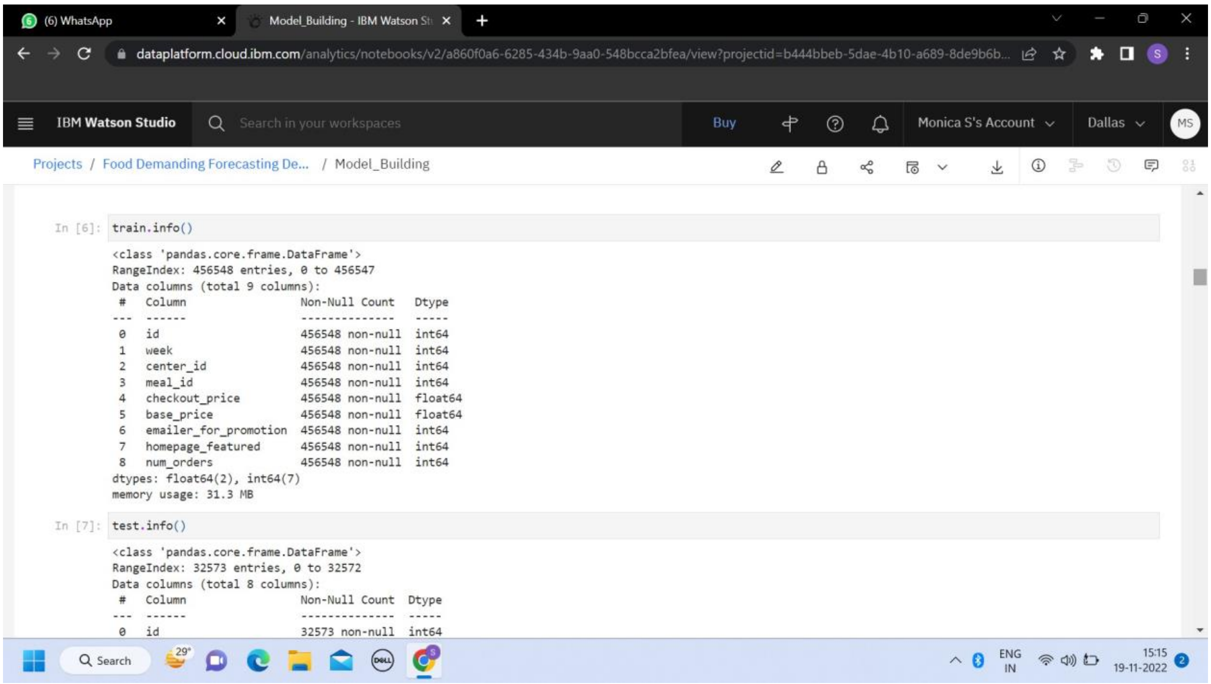
| | id | week | center_id | meal_id | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1379560 | 1 | 55 | 1885 | 136.83 | 152.29 | 0 | 0 | 177 |
| **1** | 1466964 | 1 | 55 | 1993 | 136.83 | 135.83 | 0 | 0 | 270 |
| **2** | 1346989 | 1 | 55 | 2539 | 134.86 | 135.86 | 0 | 0 | 189 |
| **3** | 1338232 | 1 | 55 | 2139 | 339.50 | 437.53 | 0 | 0 | 54 |
| **4** | 1448490 | 1 | 55 | 2631 | 243.50 | 242.50 | 0 | 0 | 40 |

In [5]:

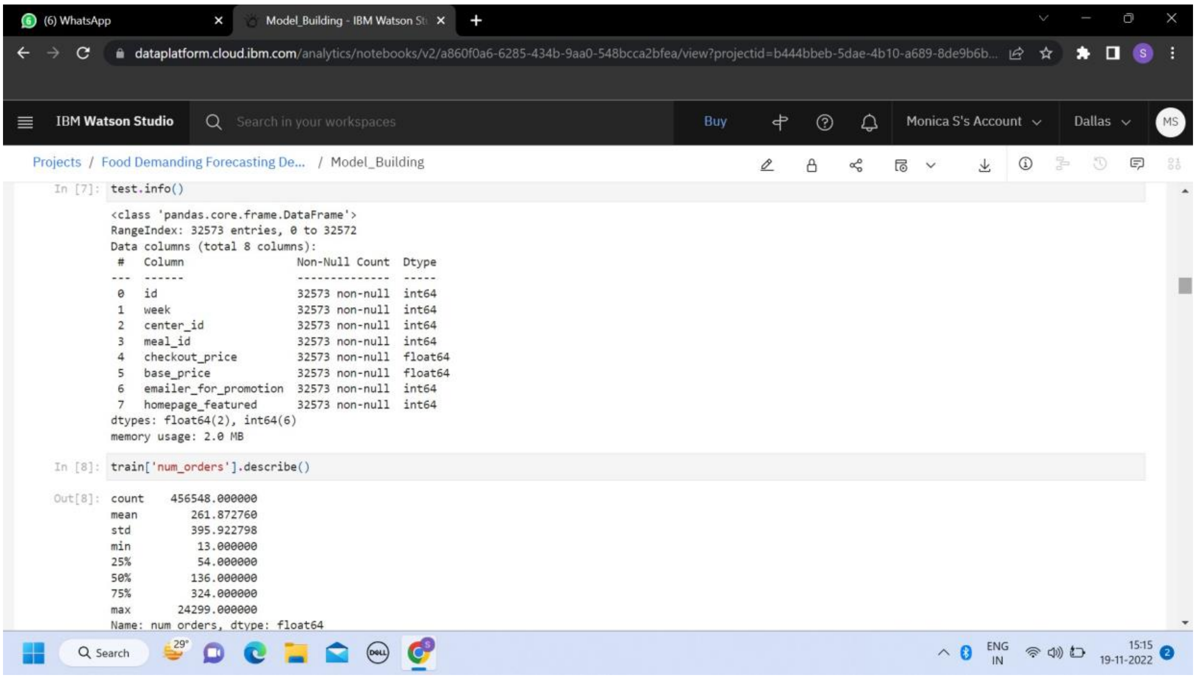
```
test.head()
```

Out[5]:

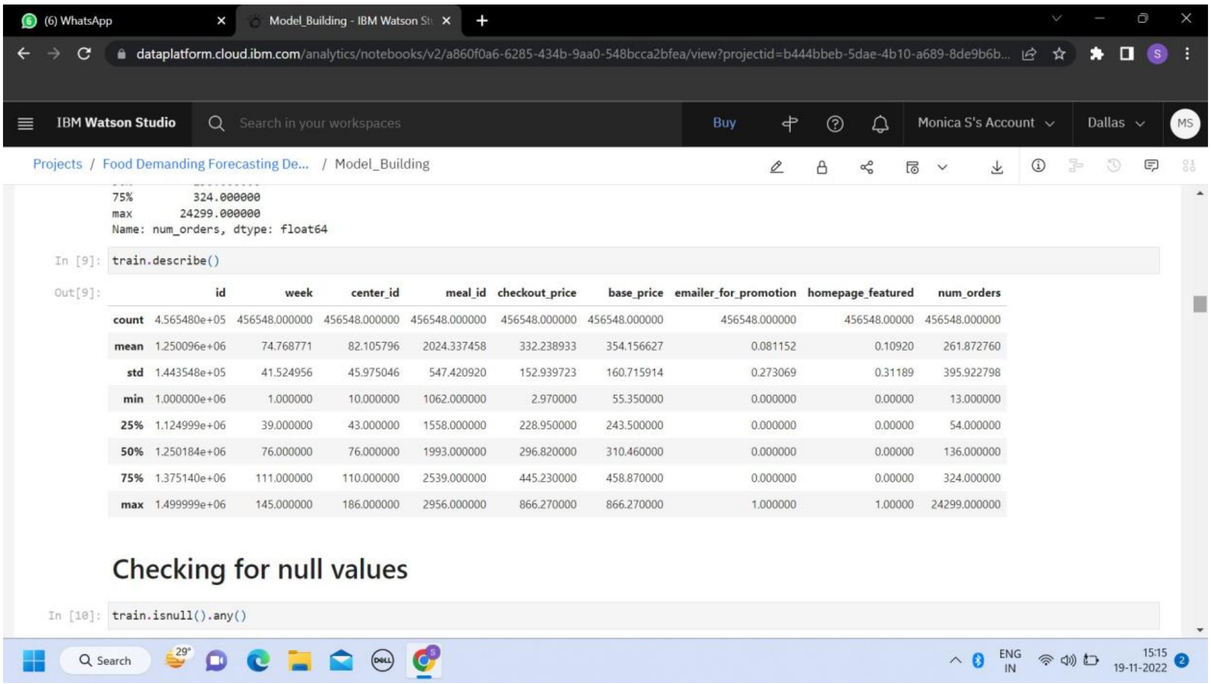
| | id | week | center_id | meal_id | checkout_price | base_price | emailer_for_promotion | homepage_featured |
|---|---|---|---|---|---|---|---|---|
| **0** | 1028232 | 146 | 55 | 1885 | 158.11 | 159.11 | 0 | 0 |
| **1** | 1127204 | 146 | 55 | 1993 | 160.11 | 159.11 | 0 | 0 |
| **2** | 1212707 | 146 | 55 | 2539 | 157.14 | 159.14 | 0 | 0 |
| **3** | 1082698 | 146 | 55 | 2631 | 162.02 | 162.02 | 0 | 0 |
| **4** | 1400926 | 146 | 55 | 1248 | 163.93 | 163.93 | 0 | 0 |


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

```
In [6]: train.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 456548 entries, 0 to 456547
Data columns (total 9 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   id                     456548 non-null  int64
 1   week                   456548 non-null  int64
 2   center_id              456548 non-null  int64
 3   meal_id                456548 non-null  int64
 4   checkout_price         456548 non-null  float64
 5   base_price             456548 non-null  float64
 6   emailer_for_promotion  456548 non-null  int64
 7   homepage_featured      456548 non-null  int64
 8   num_orders             456548 non-null  int64
dtypes: float64(2), int64(7)
memory usage: 31.3 MB

In [7]: test.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32573 entries, 0 to 32572
Data columns (total 8 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   id                     32573 non-null  int64
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

```
In [7]: test.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32573 entries, 0 to 32572
Data columns (total 8 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   id                     32573 non-null  int64
 1   week                   32573 non-null  int64
 2   center_id              32573 non-null  int64
 3   meal_id                32573 non-null  int64
 4   checkout_price         32573 non-null  float64
 5   base_price             32573 non-null  float64
 6   emailer_for_promotion  32573 non-null  int64
 7   homepage_featured      32573 non-null  int64
dtypes: float64(2), int64(6)
memory usage: 2.0 MB

In [8]: train['num_orders'].describe()

Out[8]: count    456548.000000
        mean        261.872760
        std         395.922798
        min          13.000000
        25%          54.000000
        50%         136.000000
        75%         324.000000
        max       24299.000000
        Name: num_orders, dtype: float64
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

```
75%        324.000000
max      24299.000000
Name: num_orders, dtype: float64
```

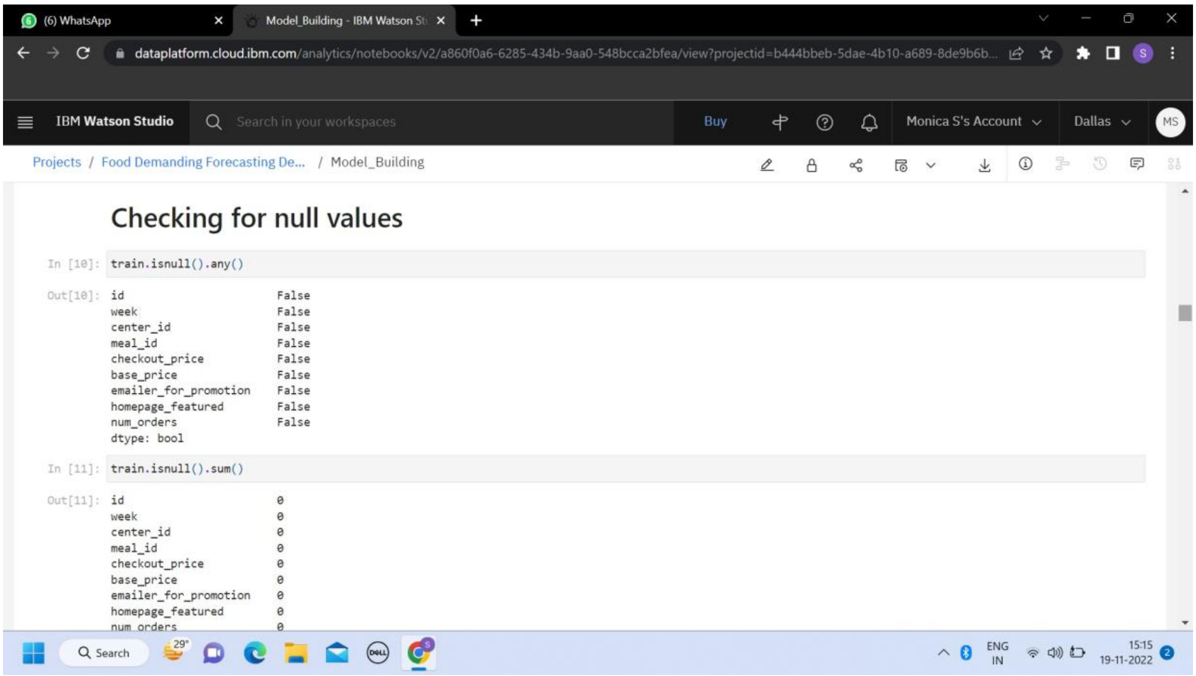
```
In [9]: train.describe()
```

Out[9]:

| | id | week | center_id | meal_id | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders |
|---|---|---|---|---|---|---|---|---|---|
| **count** | 4.565480e+05 | 456548.000000 | 456548.000000 | 456548.000000 | 456548.000000 | 456548.000000 | 456548.000000 | 456548.00000 | 456548.000000 |
| **mean** | 1.250096e+06 | 74.768771 | 82.105796 | 2024.337458 | 332.238933 | 354.156627 | 0.081152 | 0.10920 | 261.872760 |
| **std** | 1.443548e+05 | 41.524956 | 45.975046 | 547.420920 | 152.939723 | 160.715914 | 0.273069 | 0.31189 | 395.922798 |
| **min** | 1.000000e+06 | 1.000000 | 10.000000 | 1062.000000 | 2.970000 | 55.350000 | 0.000000 | 0.00000 | 13.000000 |
| **25%** | 1.124999e+06 | 39.000000 | 43.000000 | 1558.000000 | 228.950000 | 243.500000 | 0.000000 | 0.00000 | 54.000000 |
| **50%** | 1.250184e+06 | 76.000000 | 76.000000 | 1993.000000 | 296.820000 | 310.460000 | 0.000000 | 0.00000 | 136.000000 |
| **75%** | 1.375140e+06 | 111.000000 | 110.000000 | 2539.000000 | 445.230000 | 458.870000 | 0.000000 | 0.00000 | 324.000000 |
| **max** | 1.499999e+06 | 145.000000 | 186.000000 | 2956.000000 | 866.270000 | 866.270000 | 1.000000 | 1.00000 | 24299.000000 |

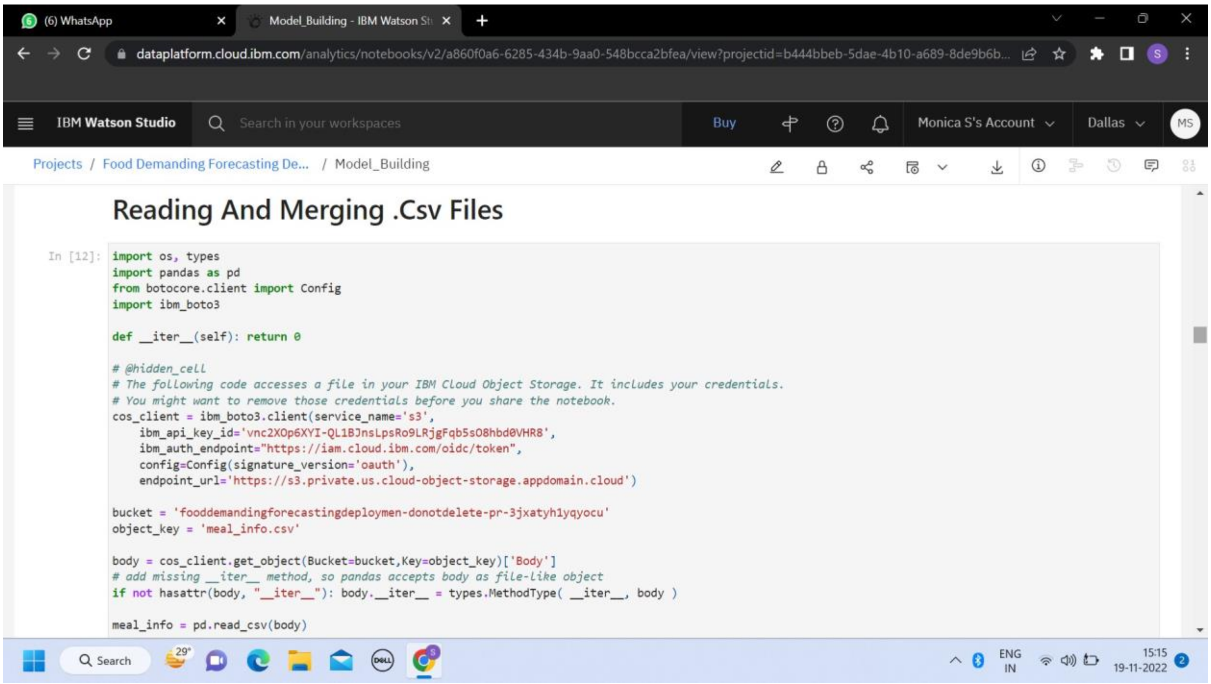
## Checking for null values

```
In [10]: train.isnull().any()
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:15
19-11-2022

## Checking for null values

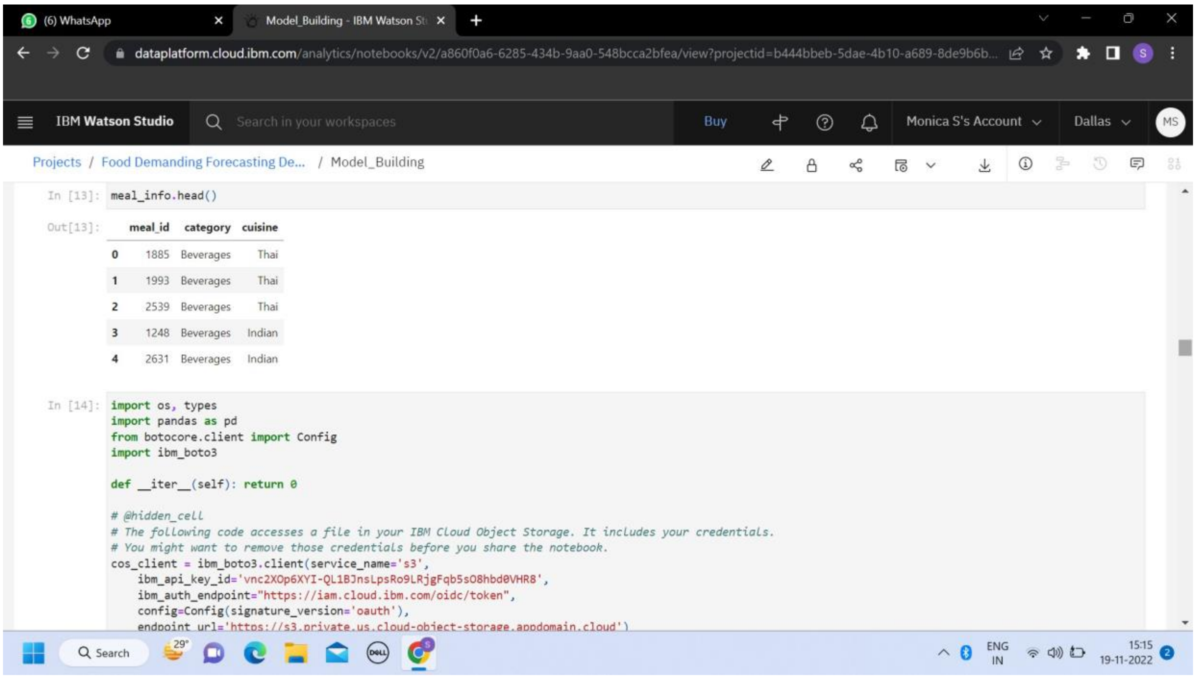
```
In [10]: train.isnull().any()

Out[10]: id                       False
         week                     False
         center_id                False
         meal_id                  False
         checkout_price           False
         base_price               False
         emailer_for_promotion    False
         homepage_featured        False
         num_orders               False
         dtype: bool

In [11]: train.isnull().sum()

Out[11]: id                       0
         week                     0
         center_id                0
         meal_id                  0
         checkout_price           0
         base_price               0
         emailer_for_promotion    0
         homepage_featured        0
         num_orders               0
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

# Reading And Merging .Csv Files

```
In [12]: import os, types
         import pandas as pd
         from botocore.client import Config
         import ibm_boto3

         def __iter__(self): return 0

         # @hidden_cell
         # The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
         # You might want to remove those credentials before you share the notebook.
         cos_client = ibm_boto3.client(service_name='s3',
             ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
             ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
             config=Config(signature_version='oauth'),
             endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

         bucket = 'fooddemandingforecastingdeploymen-donotdelete-pr-3jxatyh1yqyocu'
         object_key = 'meal_info.csv'

         body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
         # add missing __iter__ method, so pandas accepts body as file-like object
         if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

         meal_info = pd.read_csv(body)
```

Search
29°
ENG
IN
15:15
19-11-2022
2

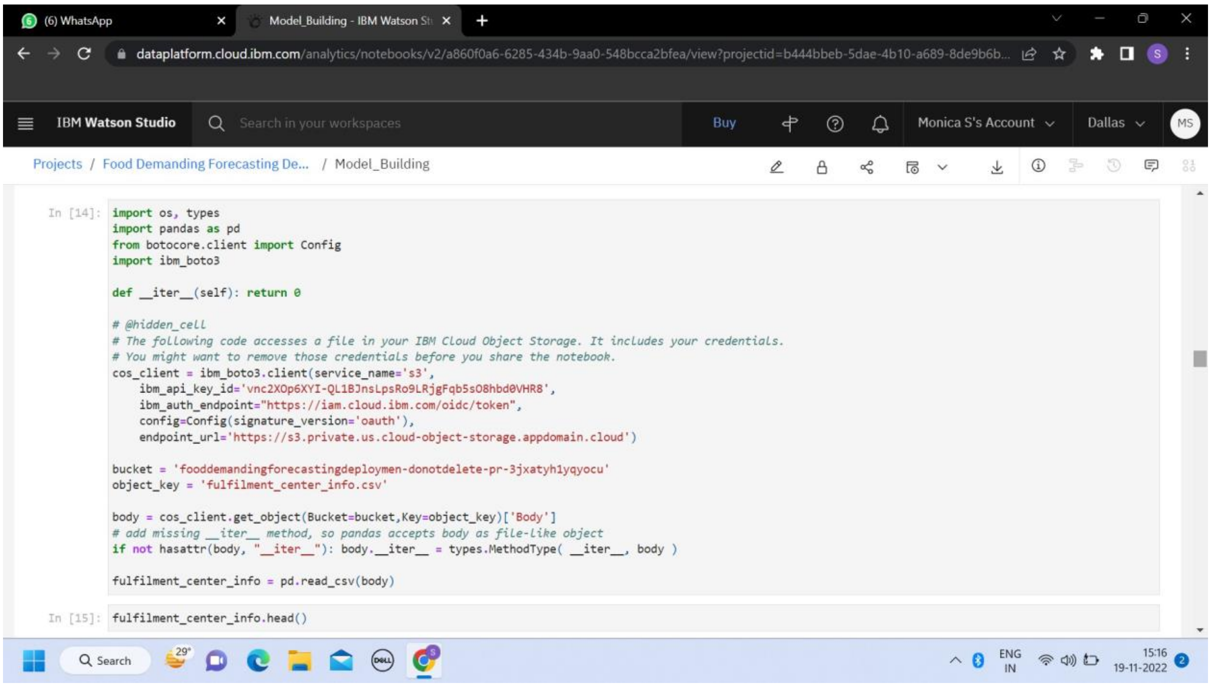

(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

```
In [13]: meal_info.head()
```

Out[13]:

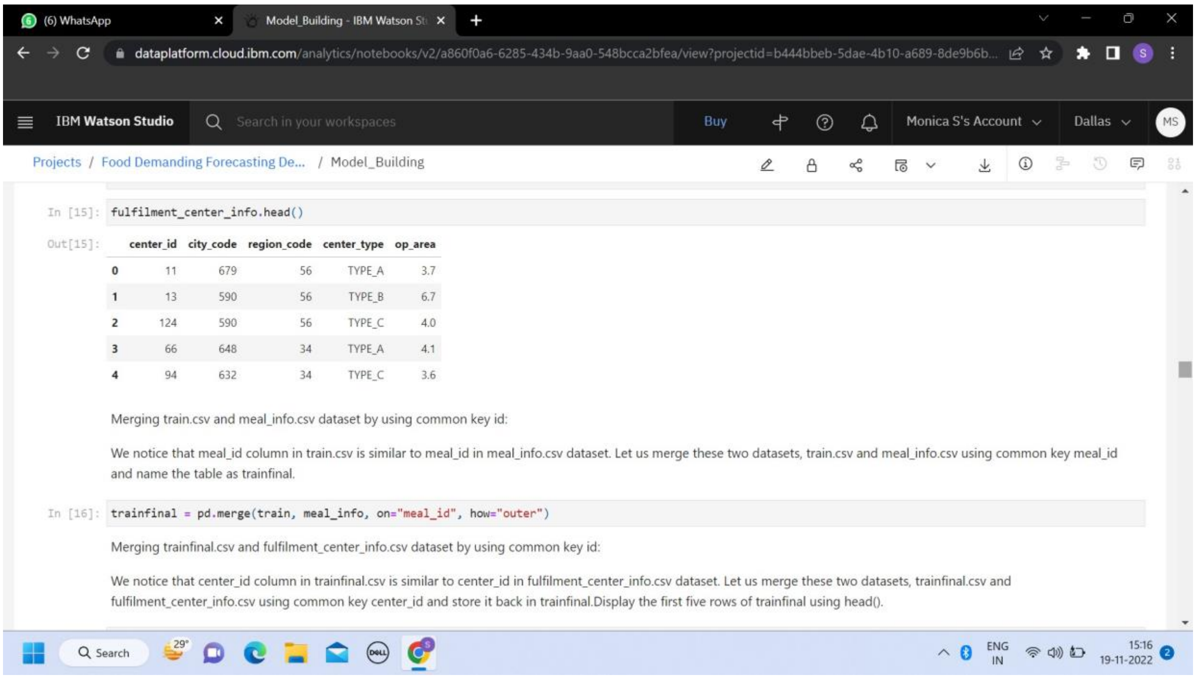
|   | meal_id | category | cuisine |
|---|---|---|---|
| 0 | 1885 | Beverages | Thai |
| 1 | 1993 | Beverages | Thai |
| 2 | 2539 | Beverages | Thai |
| 3 | 1248 | Beverages | Indian |
| 4 | 2631 | Beverages | Indian |

```
In [14]: import os, types
         import pandas as pd
         from botocore.client import Config
         import ibm_boto3

         def __iter__(self): return 0

         # @hidden_cell
         # The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
         # You might want to remove those credentials before you share the notebook.
         cos_client = ibm_boto3.client(service_name='s3',
             ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
             ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
             config=Config(signature_version='oauth'),
             endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')
```


IBM Watson Studio
Projects / Food Demanding Forecasting De... / Model_Building

In [14]:
```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='vnc2XOp6XYI-QL1BJnsLpsRo9LRjgFqb5sO8hbd0VHR8',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'fooddemandingforecastingdeploymen-donotdelete-pr-3jxatyh1yqyocu'
object_key = 'fulfilment_center_info.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

fulfilment_center_info = pd.read_csv(body)
```

In [15]:
```
fulfilment_center_info.head()
```


IBM Watson Studio
Projects / Food Demanding Forecasting De... / Model_Building

In [15]:
```
fulfilment_center_info.head()
```

Out[15]:

| | center_id | city_code | region_code | center_type | op_area |
|---|---|---|---|---|---|
| 0 | 11 | 679 | 56 | TYPE_A | 3.7 |
| 1 | 13 | 590 | 56 | TYPE_B | 6.7 |
| 2 | 124 | 590 | 56 | TYPE_C | 4.0 |
| 3 | 66 | 648 | 34 | TYPE_A | 4.1 |
| 4 | 94 | 632 | 34 | TYPE_C | 3.6 |

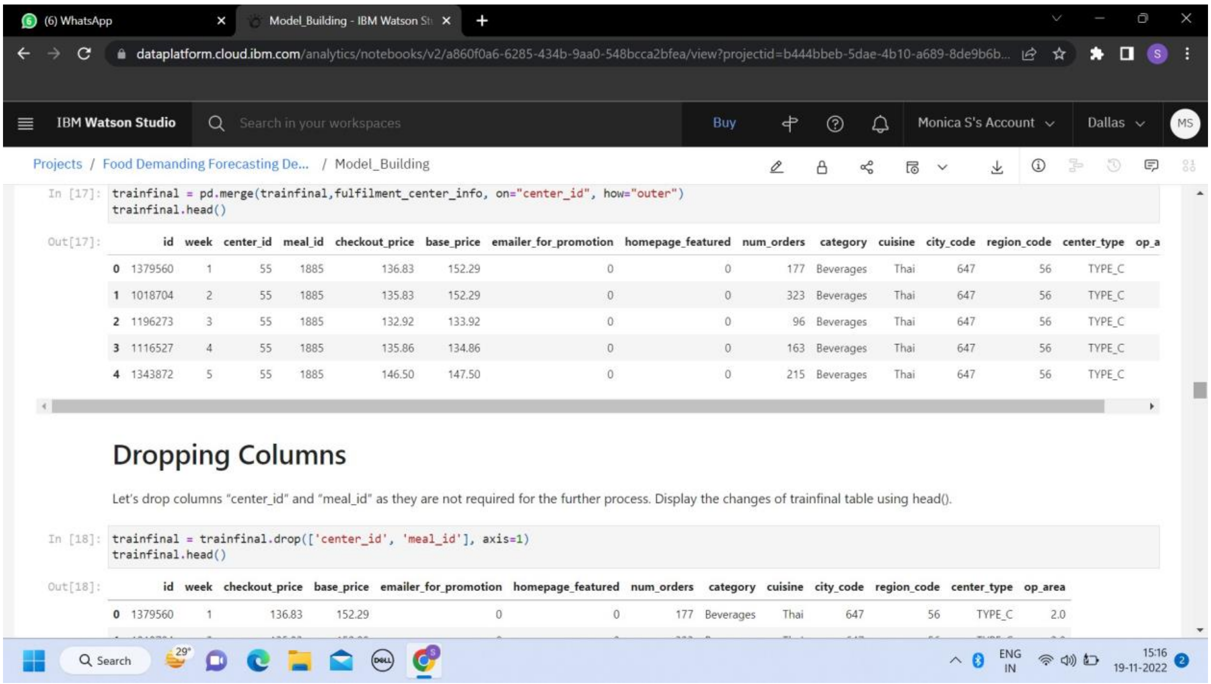
Merging train.csv and meal_info.csv dataset by using common key id:

We notice that meal_id column in train.csv is similar to meal_id in meal_info.csv dataset. Let us merge these two datasets, train.csv and meal_info.csv using common key meal_id and name the table as trainfinal.

In [16]:
```
trainfinal = pd.merge(train, meal_info, on="meal_id", how="outer")
```

Merging trainfinal.csv and fulfilment_center_info.csv dataset by using common key id:

We notice that center_id column in trainfinal.csv is similar to center_id in fulfilment_center_info.csv dataset. Let us merge these two datasets, trainfinal.csv and fulfilment_center_info.csv using common key center_id and store it back in trainfinal.Display the first five rows of trainfinal using head().


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:16
19-11-2022

In [17]:
```
trainfinal = pd.merge(trainfinal,fulfilment_center_info, on="center_id", how="outer")
trainfinal.head()
```

Out[17]:

| | id | week | center_id | meal_id | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders | category | cuisine | city_code | region_code | center_type | op_a |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1379560 | 1 | 55 | 1885 | 136.83 | 152.29 | 0 | 0 | 177 | Beverages | Thai | 647 | 56 | TYPE_C | |
| **1** | 1018704 | 2 | 55 | 1885 | 135.83 | 152.29 | 0 | 0 | 323 | Beverages | Thai | 647 | 56 | TYPE_C | |
| **2** | 1196273 | 3 | 55 | 1885 | 132.92 | 133.92 | 0 | 0 | 96 | Beverages | Thai | 647 | 56 | TYPE_C | |
| **3** | 1116527 | 4 | 55 | 1885 | 135.86 | 134.86 | 0 | 0 | 163 | Beverages | Thai | 647 | 56 | TYPE_C | |
| **4** | 1343872 | 5 | 55 | 1885 | 146.50 | 147.50 | 0 | 0 | 215 | Beverages | Thai | 647 | 56 | TYPE_C | |

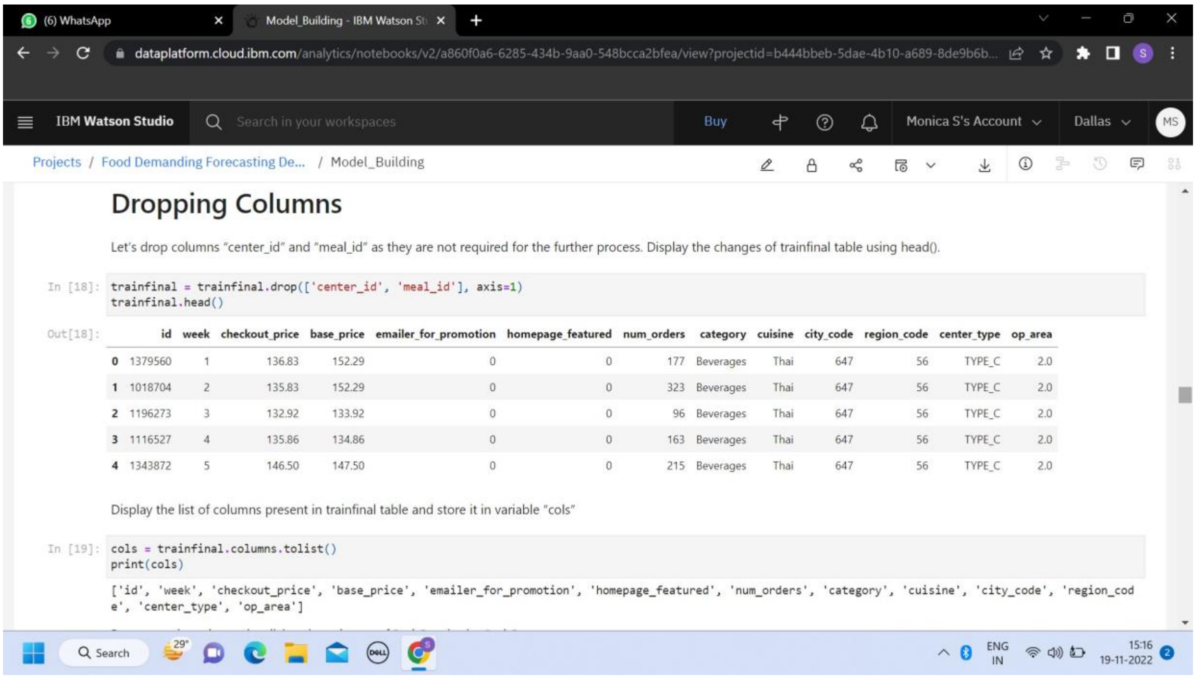
# Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

In [18]:
```
trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
trainfinal.head()
```

Out[18]:

| | id | week | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders | category | cuisine | city_code | region_code | center_type | op_area |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1379560 | 1 | 136.83 | 152.29 | 0 | 0 | 177 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:16
19-11-2022

# Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

In [18]:
```
trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
trainfinal.head()
```

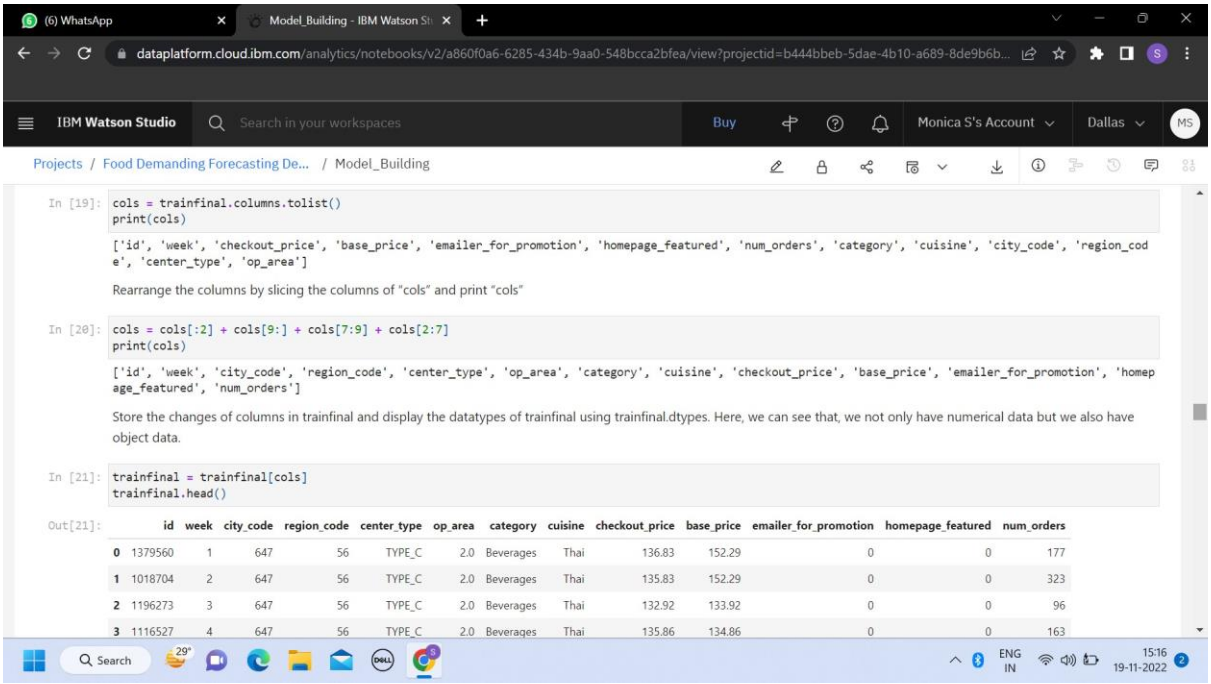
Out[18]:

| | id | week | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders | category | cuisine | city_code | region_code | center_type | op_area |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1379560 | 1 | 136.83 | 152.29 | 0 | 0 | 177 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |
| **1** | 1018704 | 2 | 135.83 | 152.29 | 0 | 0 | 323 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |
| **2** | 1196273 | 3 | 132.92 | 133.92 | 0 | 0 | 96 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |
| **3** | 1116527 | 4 | 135.86 | 134.86 | 0 | 0 | 163 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |
| **4** | 1343872 | 5 | 146.50 | 147.50 | 0 | 0 | 215 | Beverages | Thai | 647 | 56 | TYPE_C | 2.0 |

Display the list of columns present in trainfinal table and store it in variable "cols"

In [19]:
```
cols = trainfinal.columns.tolist()
print(cols)
```

```
['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_cod
e', 'center_type', 'op_area']
```



(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

```
In [19]: cols = trainfinal.columns.tolist()
         print(cols)

         ['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_cod
         e', 'center_type', 'op_area']
```

Rearrange the columns by slicing the columns of "cols" and print "cols"

```
In [20]: cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]
         print(cols)

         ['id', 'week', 'city_code', 'region_code', 'center_type', 'op_area', 'category', 'cuisine', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homep
         age_featured', 'num_orders']
```

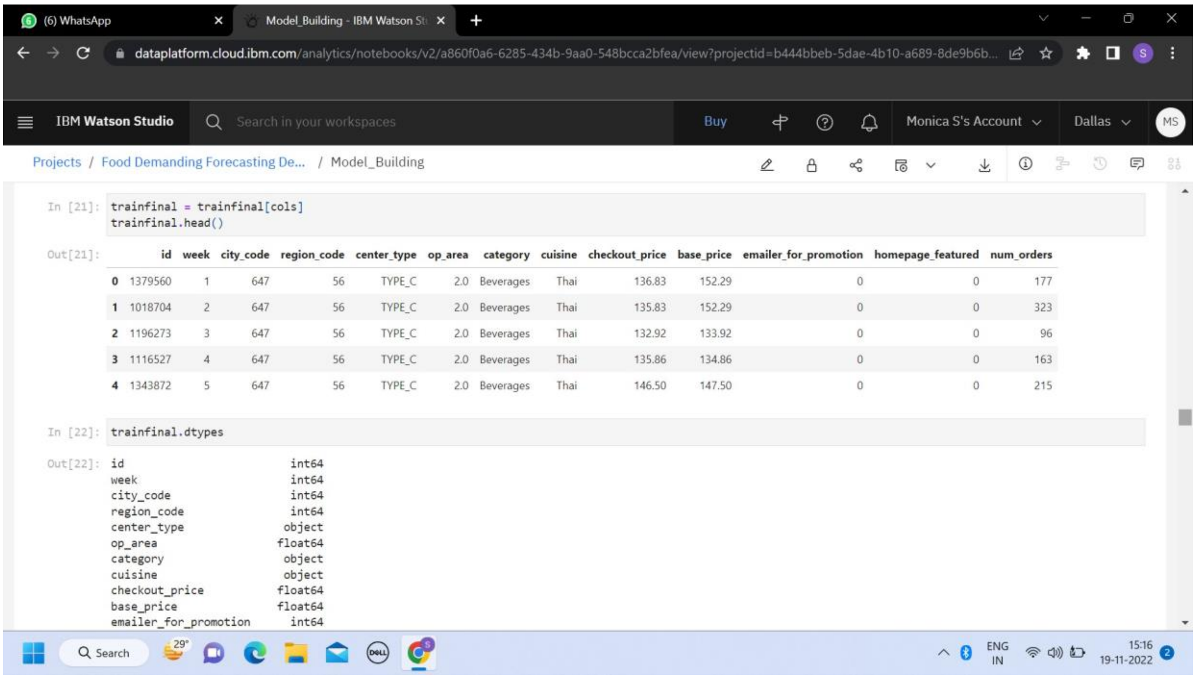
Store the changes of columns in trainfinal and display the datatypes of trainfinal using trainfinal.dtypes. Here, we can see that, we not only have numerical data but we also have object data.

```
In [21]: trainfinal = trainfinal[cols]
         trainfinal.head()
```

Out[21]:

| | id | week | city_code | region_code | center_type | op_area | category | cuisine | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1379560 | 1 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 136.83 | 152.29 | 0 | 0 | 177 |
| 1 | 1018704 | 2 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 135.83 | 152.29 | 0 | 0 | 323 |
| 2 | 1196273 | 3 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 132.92 | 133.92 | 0 | 0 | 96 |
| 3 | 1116527 | 4 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 135.86 | 134.86 | 0 | 0 | 163 |

Search
29°
ENG
IN
15:16
19-11-2022



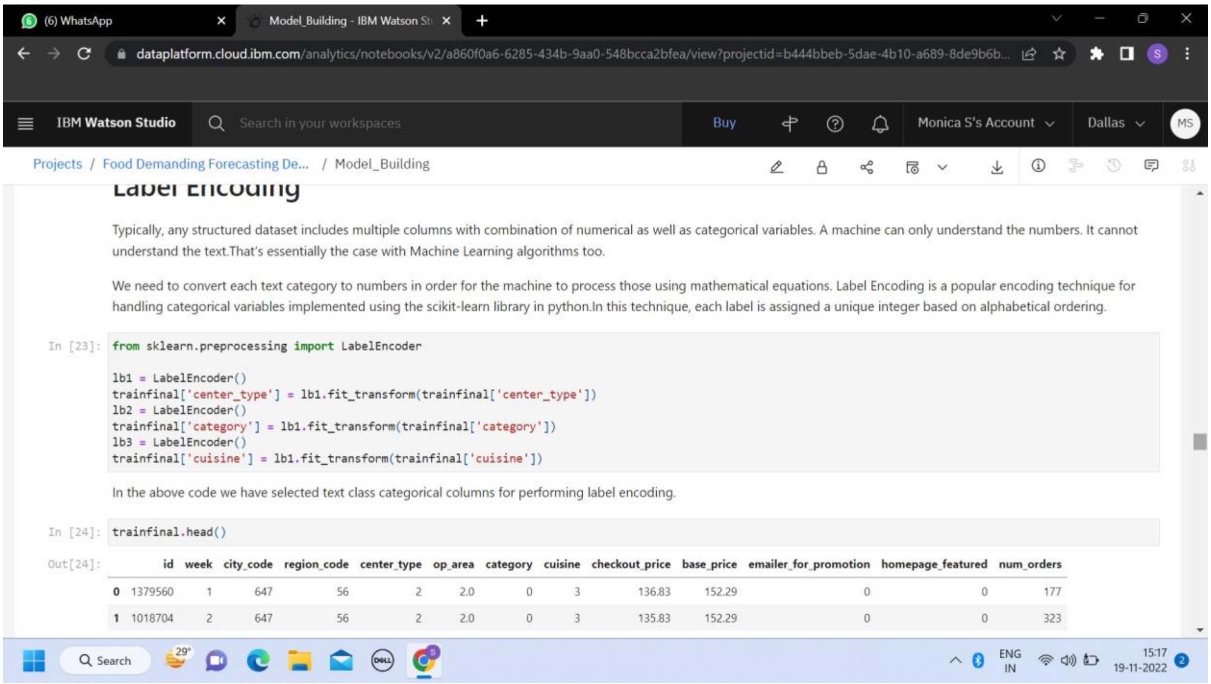
(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

```
In [21]: trainfinal = trainfinal[cols]
         trainfinal.head()
```

Out[21]:

| | id | week | city_code | region_code | center_type | op_area | category | cuisine | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1379560 | 1 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 136.83 | 152.29 | 0 | 0 | 177 |
| 1 | 1018704 | 2 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 135.83 | 152.29 | 0 | 0 | 323 |
| 2 | 1196273 | 3 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 132.92 | 133.92 | 0 | 0 | 96 |
| 3 | 1116527 | 4 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 135.86 | 134.86 | 0 | 0 | 163 |
| 4 | 1343872 | 5 | 647 | 56 | TYPE_C | 2.0 | Beverages | Thai | 146.50 | 147.50 | 0 | 0 | 215 |

```
In [22]: trainfinal.dtypes

Out[22]: id                         int64
         week                       int64
         city_code                  int64
         region_code                int64
         center_type               object
         op_area                  float64
         category                  object
         cuisine                   object
         checkout_price           float64
         base_price               float64
         emailer_for_promotion      int64
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

## Label Encoding

Typically, any structured dataset includes multiple columns with combination of numerical as well as categorical variables. A machine can only understand the numbers. It cannot understand the text.That's essentially the case with Machine Learning algorithms too.

We need to convert each text category to numbers in order for the machine to process those using mathematical equations. Label Encoding is a popular encoding technique for handling categorical variables implemented using the scikit-learn library in python.In this technique, each label is assigned a unique integer based on alphabetical ordering.

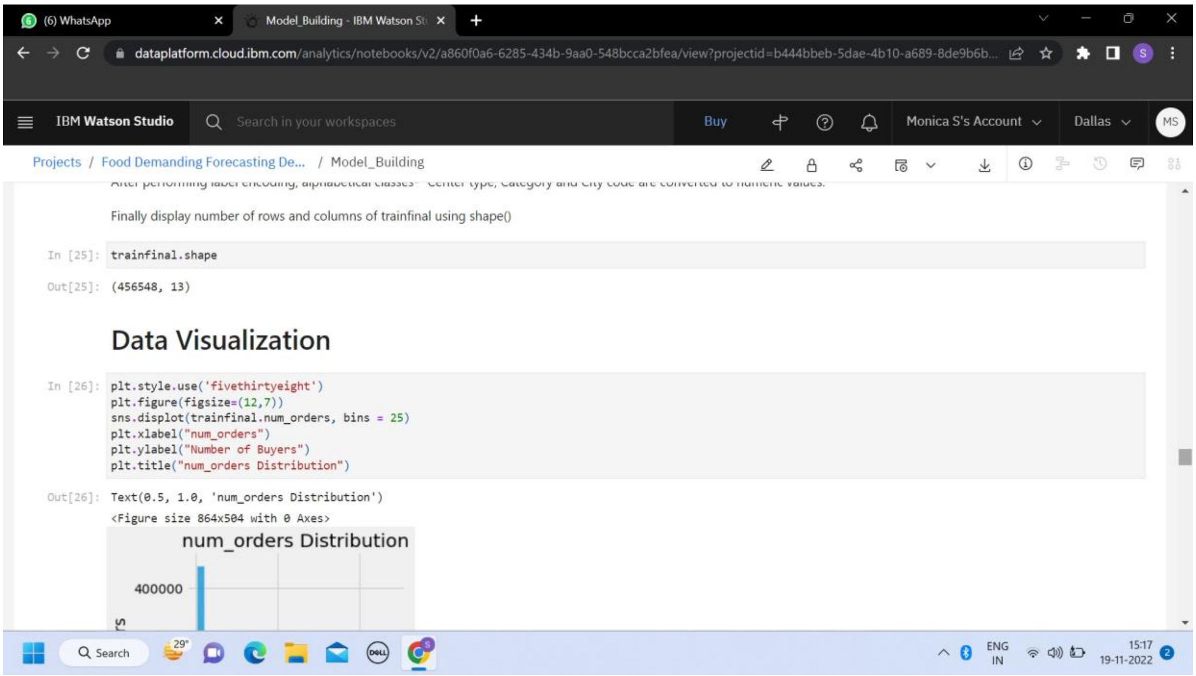
```
In [23]: from sklearn.preprocessing import LabelEncoder

         lb1 = LabelEncoder()
         trainfinal['center_type'] = lb1.fit_transform(trainfinal['center_type'])
         lb2 = LabelEncoder()
         trainfinal['category'] = lb1.fit_transform(trainfinal['category'])
         lb3 = LabelEncoder()
         trainfinal['cuisine'] = lb1.fit_transform(trainfinal['cuisine'])
```

In the above code we have selected text class categorical columns for performing label encoding.

```
In [24]: trainfinal.head()
```

Out[24]:

|  | id | week | city_code | region_code | center_type | op_area | category | cuisine | checkout_price | base_price | emailer_for_promotion | homepage_featured | num_orders |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1379560 | 1 | 647 | 56 | 2 | 2.0 | 0 | 3 | 136.83 | 152.29 | 0 | 0 | 177 |
| **1** | 1018704 | 2 | 647 | 56 | 2 | 2.0 | 0 | 3 | 135.83 | 152.29 | 0 | 0 | 323 |


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
num_orders Distribution
400000
Search
29°
ENG
IN
15:17
19-11-2022

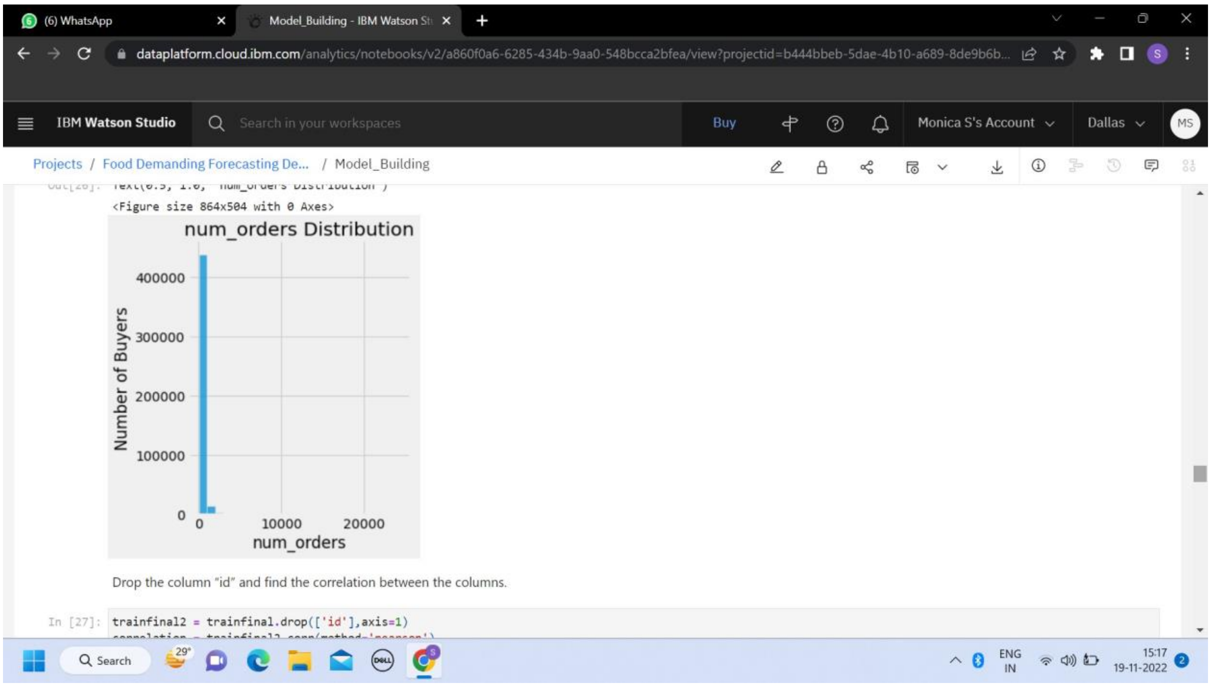
After performing label encoding, alphabetical classes- Center type, Category and City code are converted to numeric values.

Finally display number of rows and columns of trainfinal using shape()

```
In [25]: trainfinal.shape

Out[25]: (456548, 13)
```

## Data Visualization

```
In [26]: plt.style.use('fivethirtyeight')
         plt.figure(figsize=(12,7))
         sns.displot(trainfinal.num_orders, bins = 25)
         plt.xlabel("num_orders")
         plt.ylabel("Number of Buyers")
         plt.title("num_orders Distribution")

Out[26]: Text(0.5, 1.0, 'num_orders Distribution')
         <Figure size 864x504 with 0 Axes>
```



(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

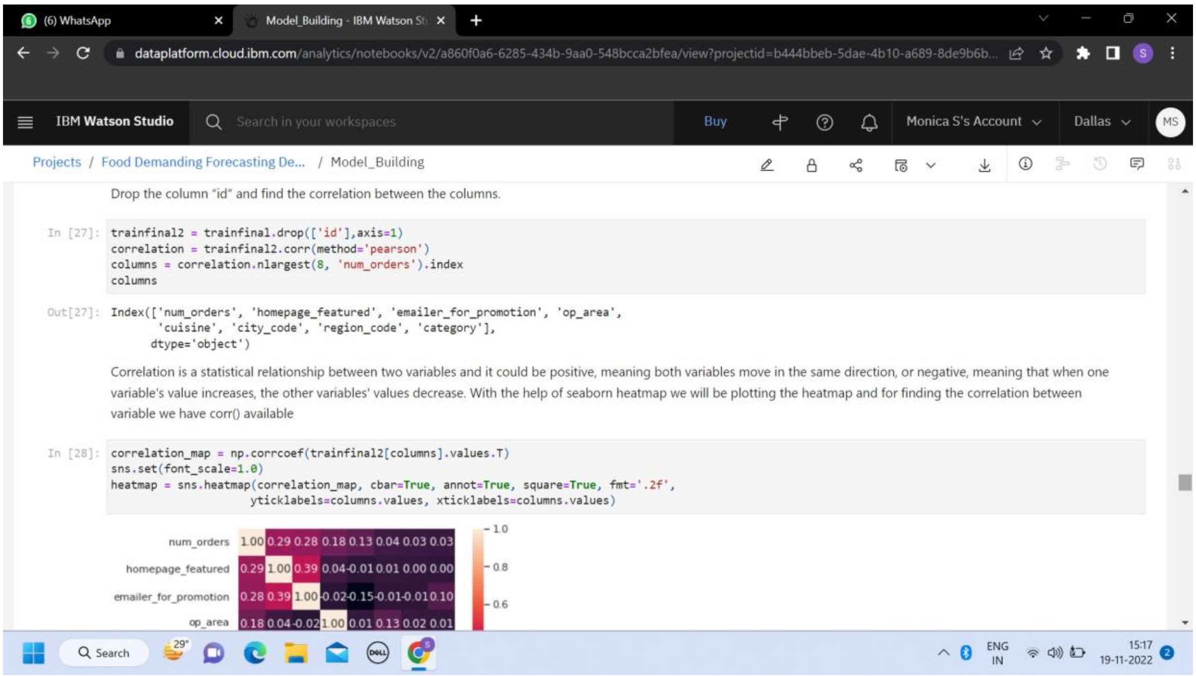
```
Out[26]: Text(0.5, 1.0, 'num_orders Distribution')
<Figure size 864x504 with 0 Axes>
```

num_orders Distribution
Number of Buyers
400000
300000
200000
100000
0
0
10000
20000
num_orders

Drop the column "id" and find the correlation between the columns.

```
In [27]: trainfinal2 = trainfinal.drop(['id'],axis=1)
```

Search
29°
ENG
IN
15:17
19-11-2022



(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

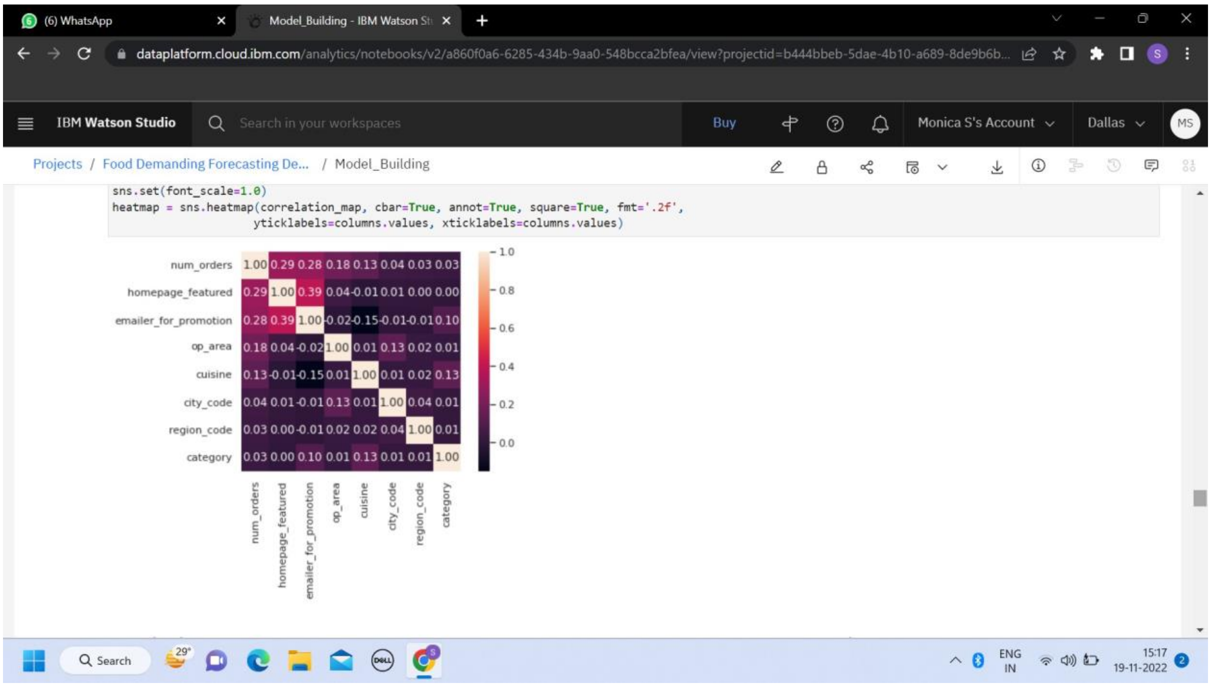
Drop the column "id" and find the correlation between the columns.

```
In [27]: trainfinal2 = trainfinal.drop(['id'],axis=1)
         correlation = trainfinal2.corr(method='pearson')
         columns = correlation.nlargest(8, 'num_orders').index
         columns

Out[27]: Index(['num_orders', 'homepage_featured', 'emailer_for_promotion', 'op_area',
                'cuisine', 'city_code', 'region_code', 'category'],
               dtype='object')
```

Correlation is a statistical relationship between two variables and it could be positive, meaning both variables move in the same direction, or negative, meaning that when one variable's value increases, the other variables' values decrease. With the help of seaborn heatmap we will be plotting the heatmap and for finding the correlation between variable we have corr() available

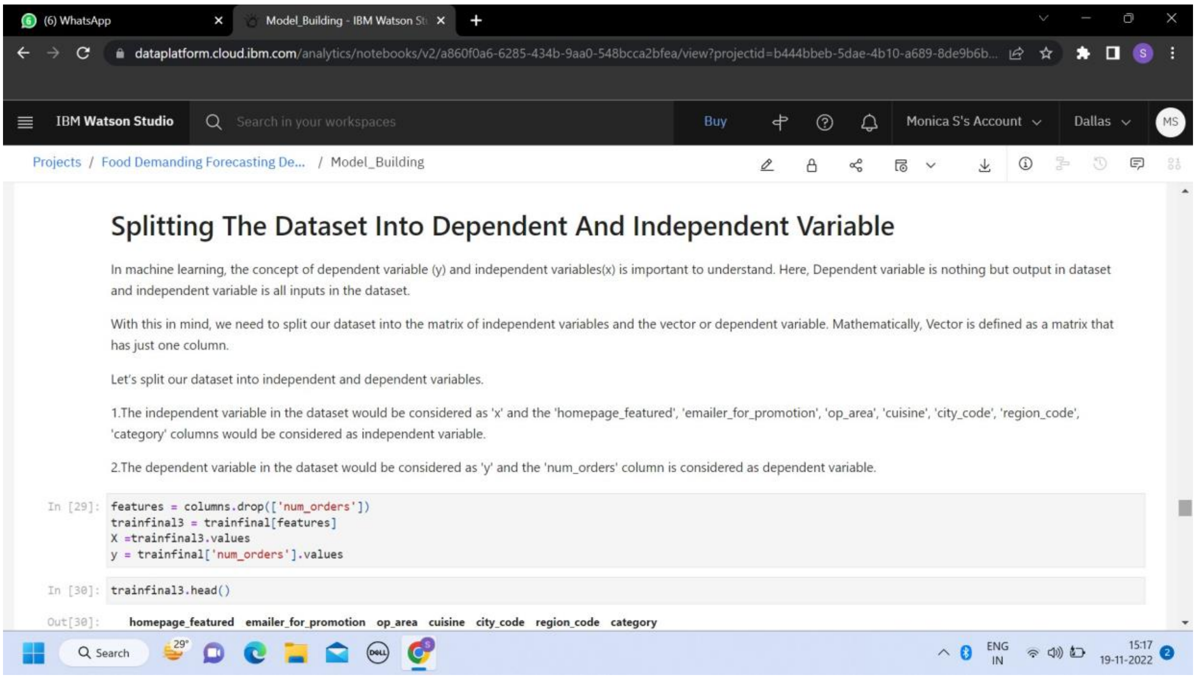
```
In [28]: correlation_map = np.corrcoef(trainfinal2[columns].values.T)
         sns.set(font_scale=1.0)
         heatmap = sns.heatmap(correlation_map, cbar=True, annot=True, square=True, fmt='.2f',
                               yticklabels=columns.values, xticklabels=columns.values)
```

num_orders 1.00 0.29 0.28 0.18 0.13 0.04 0.03 0.03
homepage_featured 0.29 1.00 0.39 0.04 -0.01 0.01 0.00 0.00
emailer_for_promotion 0.28 0.39 1.00 -0.02 -0.15 -0.01 -0.01 0.10
op_area 0.18 0.04 -0.02 1.00 0.01 0.13 0.02 0.01
1.0
0.8
0.6
Search
29°
ENG
IN
15:17
19-11-2022


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

```
sns.set(font_scale=1.0)
heatmap = sns.heatmap(correlation_map, cbar=True, annot=True, square=True, fmt='.2f',
                      yticklabels=columns.values, xticklabels=columns.values)
```

| | num_orders | homepage_featured | emailer_for_promotion | op_area | cuisine | city_code | region_code | category |
|---|---|---|---|---|---|---|---|---|
| num_orders | 1.00 | 0.29 | 0.28 | 0.18 | 0.13 | 0.04 | 0.03 | 0.03 |
| homepage_featured | 0.29 | 1.00 | 0.39 | 0.04 | -0.01 | 0.01 | 0.00 | 0.00 |
| emailer_for_promotion | 0.28 | 0.39 | 1.00 | -0.02 | -0.15 | -0.01 | -0.01 | 0.10 |
| op_area | 0.18 | 0.04 | -0.02 | 1.00 | 0.01 | 0.13 | 0.02 | 0.01 |
| cuisine | 0.13 | -0.01 | -0.15 | 0.01 | 1.00 | 0.01 | 0.02 | 0.13 |
| city_code | 0.04 | 0.01 | -0.01 | 0.13 | 0.01 | 1.00 | 0.04 | 0.01 |
| region_code | 0.03 | 0.00 | -0.01 | 0.02 | 0.02 | 0.04 | 1.00 | 0.01 |
| category | 0.03 | 0.00 | 0.10 | 0.01 | 0.13 | 0.01 | 0.01 | 1.00 |


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

## Splitting The Dataset Into Dependent And Independent Variable

In machine learning, the concept of dependent variable (y) and independent variables(x) is important to understand. Here, Dependent variable is nothing but output in dataset and independent variable is all inputs in the dataset.

With this in mind, we need to split our dataset into the matrix of independent variables and the vector or dependent variable. Mathematically, Vector is defined as a matrix that has just one column.

Let's split our dataset into independent and dependent variables.

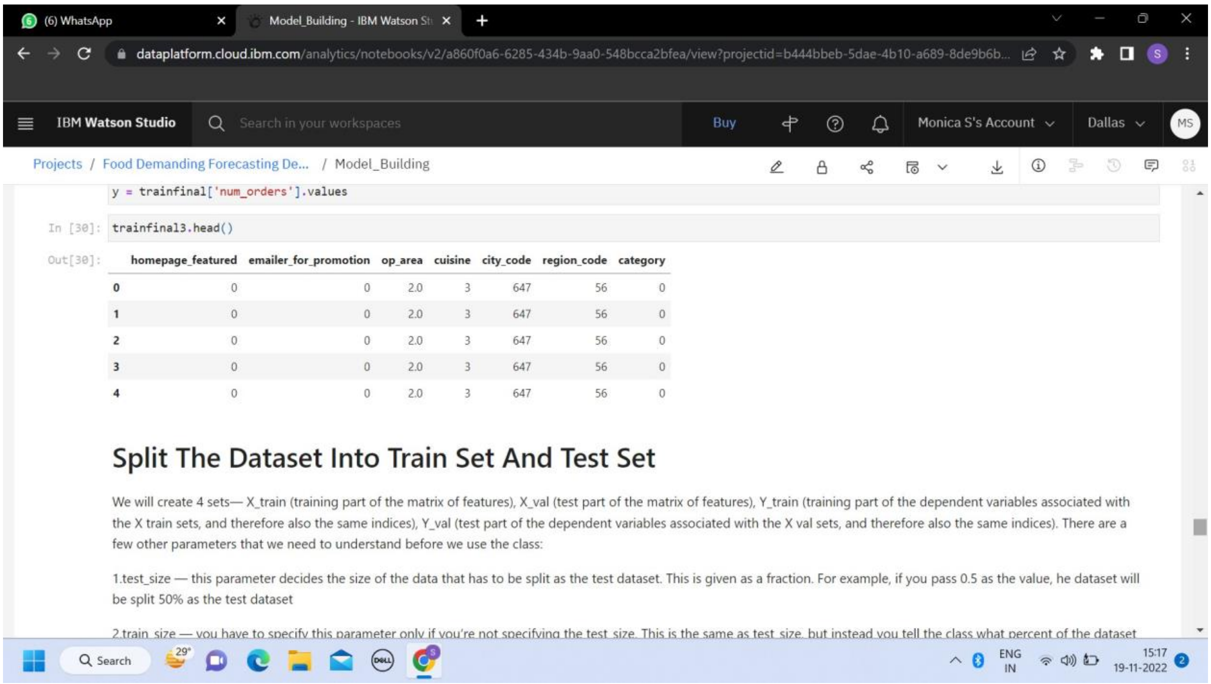
1.The independent variable in the dataset would be considered as 'x' and the 'homepage_featured', 'emailer_for_promotion', 'op_area', 'cuisine', 'city_code', 'region_code', 'category' columns would be considered as independent variable.

2.The dependent variable in the dataset would be considered as 'y' and the 'num_orders' column is considered as dependent variable.

```
In [29]: features = columns.drop(['num_orders'])
         trainfinal3 = trainfinal[features]
         X =trainfinal3.values
         y = trainfinal['num_orders'].values
```

```
In [30]: trainfinal3.head()
```

```
Out[30]:    homepage_featured   emailer_for_promotion   op_area   cuisine   city_code   region_code   category
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

```
y = trainfinal['num_orders'].values
```

In [30]:
```
trainfinal3.head()
```

Out[30]:

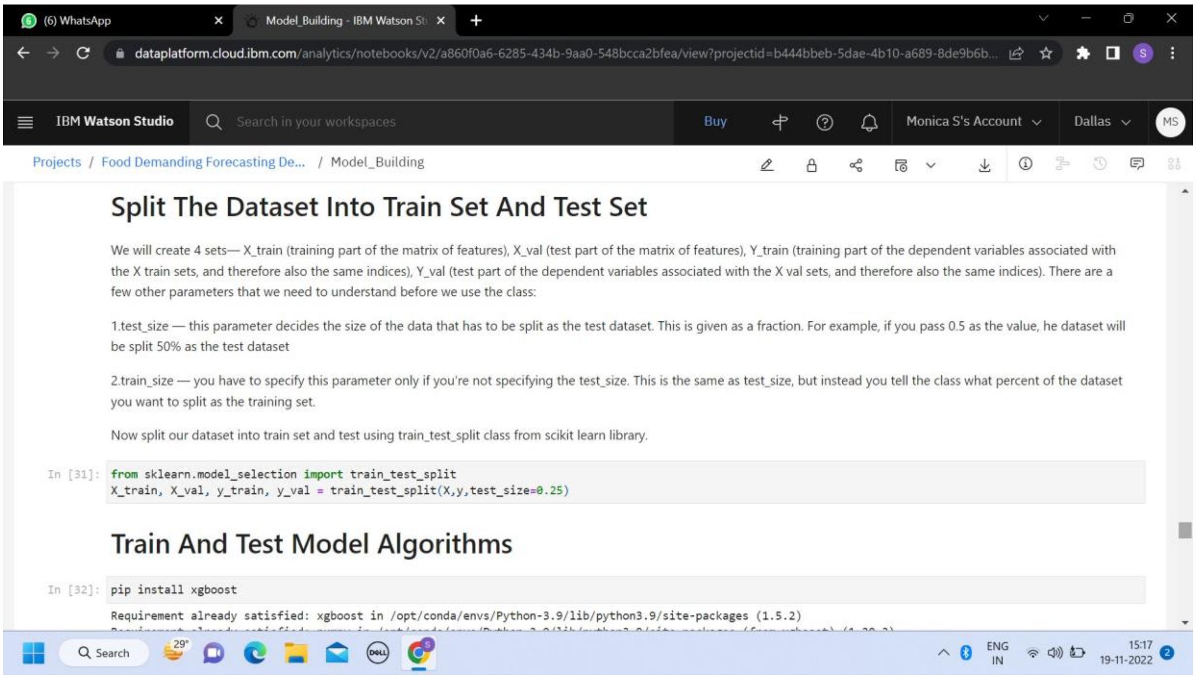
| | homepage_featured | emailer_for_promotion | op_area | cuisine | city_code | region_code | category |
|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 2.0 | 3 | 647 | 56 | 0 |
| **1** | 0 | 0 | 2.0 | 3 | 647 | 56 | 0 |
| **2** | 0 | 0 | 2.0 | 3 | 647 | 56 | 0 |
| **3** | 0 | 0 | 2.0 | 3 | 647 | 56 | 0 |
| **4** | 0 | 0 | 2.0 | 3 | 647 | 56 | 0 |

## Split The Dataset Into Train Set And Test Set

We will create 4 sets— X_train (training part of the matrix of features), X_val (test part of the matrix of features), Y_train (training part of the dependent variables associated with the X train sets, and therefore also the same indices), Y_val (test part of the dependent variables associated with the X val sets, and therefore also the same indices). There are a few other parameters that we need to understand before we use the class:

1.test_size — this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, he dataset will be split 50% as the test dataset

2.train_size — you have to specify this parameter only if you're not specifying the test_size. This is the same as test_size, but instead you tell the class what percent of the dataset


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

## Split The Dataset Into Train Set And Test Set

We will create 4 sets— X_train (training part of the matrix of features), X_val (test part of the matrix of features), Y_train (training part of the dependent variables associated with the X train sets, and therefore also the same indices), Y_val (test part of the dependent variables associated with the X val sets, and therefore also the same indices). There are a few other parameters that we need to understand before we use the class:

1.test_size — this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, he dataset will be split 50% as the test dataset

2.train_size — you have to specify this parameter only if you're not specifying the test_size. This is the same as test_size, but instead you tell the class what percent of the dataset you want to split as the training set.

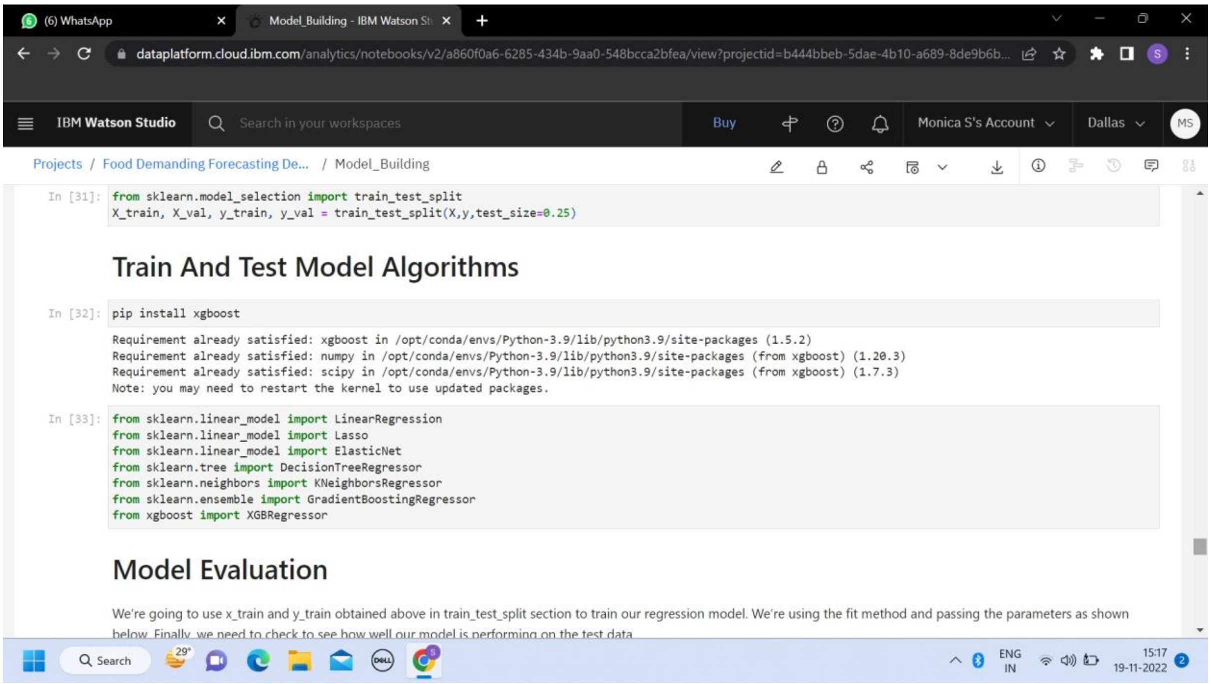
Now split our dataset into train set and test using train_test_split class from scikit learn library.

In [31]:
```
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split(X,y,test_size=0.25)
```

## Train And Test Model Algorithms

In [32]:
```
pip install xgboost
```

```
Requirement already satisfied: xgboost in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.5.2)
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/a860f0a6-6285-434b-9aa0-548bcca2bfea/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6b...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:17
19-11-2022

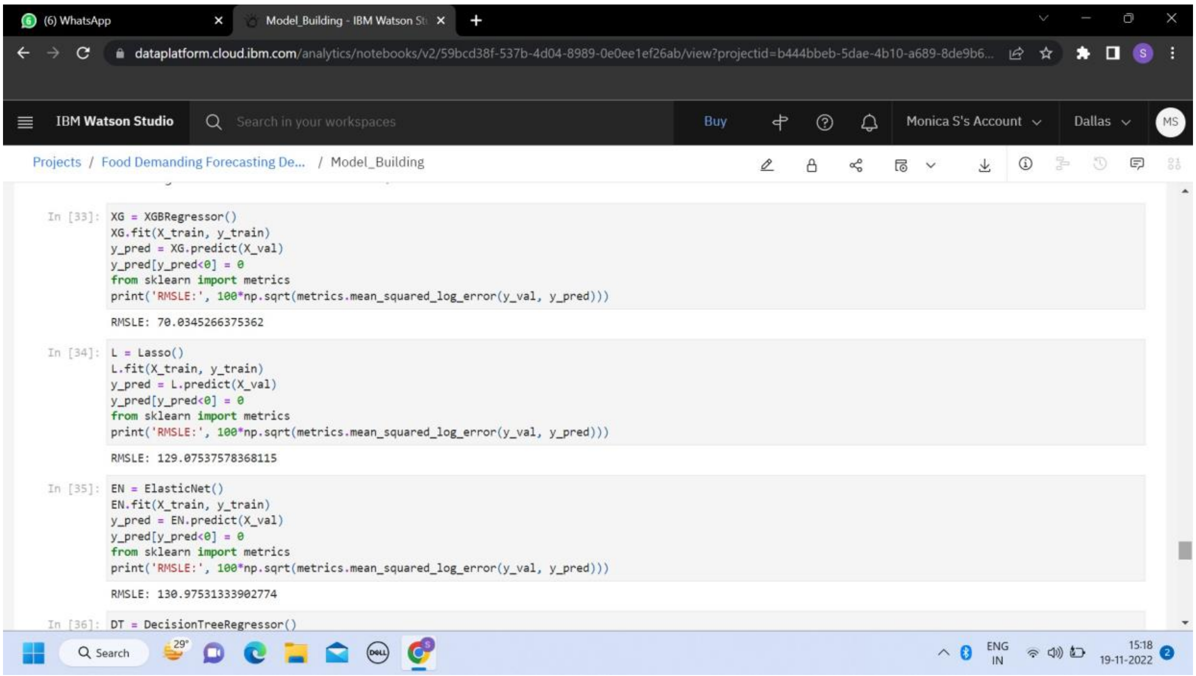
```
In [31]: from sklearn.model_selection import train_test_split
         X_train, X_val, y_train, y_val = train_test_split(X,y,test_size=0.25)
```

## Train And Test Model Algorithms

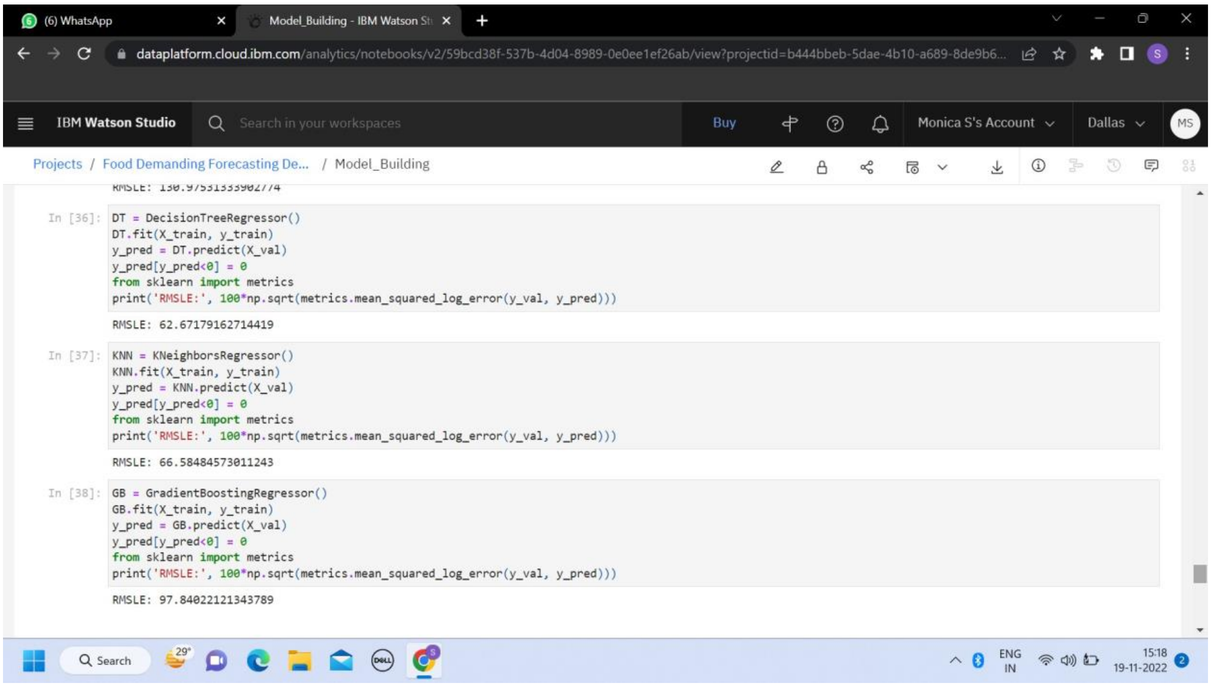
```
In [32]: pip install xgboost

         Requirement already satisfied: xgboost in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.5.2)
         Requirement already satisfied: numpy in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from xgboost) (1.20.3)
         Requirement already satisfied: scipy in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from xgboost) (1.7.3)
         Note: you may need to restart the kernel to use updated packages.

In [33]: from sklearn.linear_model import LinearRegression
         from sklearn.linear_model import Lasso
         from sklearn.linear_model import ElasticNet
         from sklearn.tree import DecisionTreeRegressor
         from sklearn.neighbors import KNeighborsRegressor
         from sklearn.ensemble import GradientBoostingRegressor
         from xgboost import XGBRegressor
```

## Model Evaluation

We're going to use x_train and y_train obtained above in train_test_split section to train our regression model. We're using the fit method and passing the parameters as shown below. Finally, we need to check to see how well our model is performing on the test data.


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:18
19-11-2022

```
In [33]: XG = XGBRegressor()
         XG.fit(X_train, y_train)
         y_pred = XG.predict(X_val)
         y_pred[y_pred<0] = 0
         from sklearn import metrics
         print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))

         RMSLE: 70.0345266375362

In [34]: L = Lasso()
         L.fit(X_train, y_train)
         y_pred = L.predict(X_val)
         y_pred[y_pred<0] = 0
         from sklearn import metrics
         print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))

         RMSLE: 129.07537578368115

In [35]: EN = ElasticNet()
         EN.fit(X_train, y_train)
         y_pred = EN.predict(X_val)
         y_pred[y_pred<0] = 0
         from sklearn import metrics
         print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))

         RMSLE: 130.97531333902774

In [36]: DT = DecisionTreeRegressor()
```



(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building
Search
29°
ENG
IN
15:18
19-11-2022

```
RMSLE: 130.97531333902774
```

In [36]:

```
DT = DecisionTreeRegressor()
DT.fit(X_train, y_train)
y_pred = DT.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 62.67179162714419
```

In [37]:

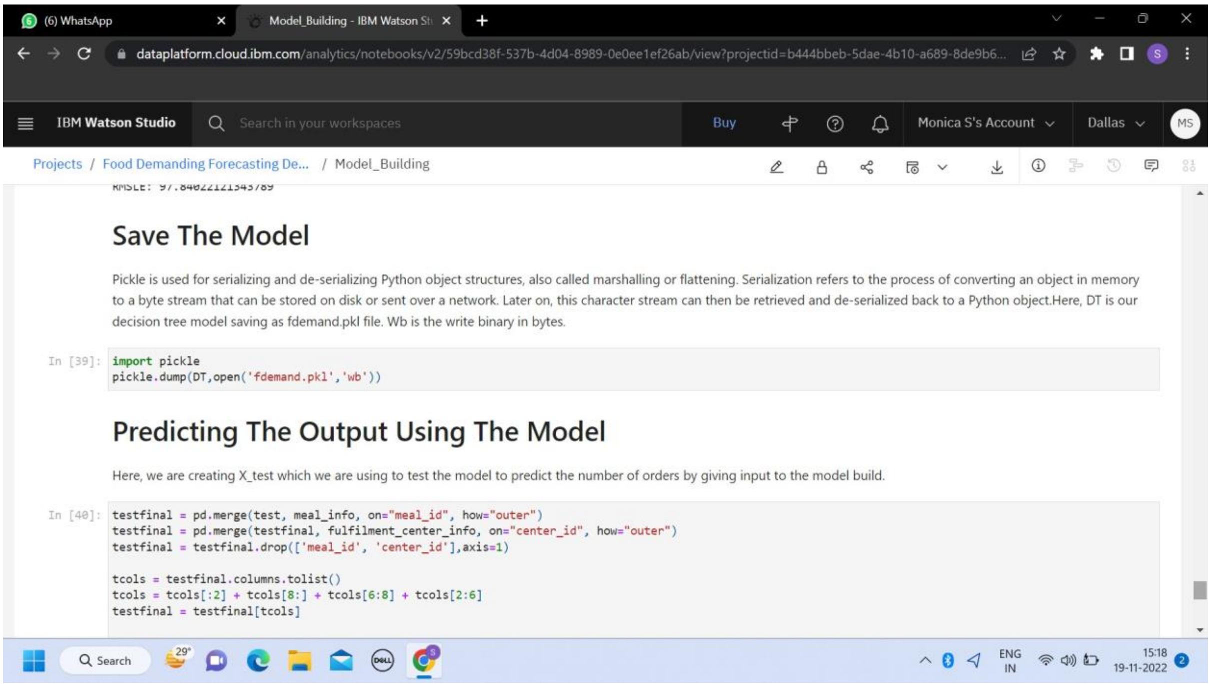
```
KNN = KNeighborsRegressor()
KNN.fit(X_train, y_train)
y_pred = KNN.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 66.58484573011243
```

In [38]:

```
GB = GradientBoostingRegressor()
GB.fit(X_train, y_train)
y_pred = GB.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 97.84022121343789
```


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

```
RMSLE: 97.84022121343789
```

## Save The Model

Pickle is used for serializing and de-serializing Python object structures, also called marshalling or flattening. Serialization refers to the process of converting an object in memory to a byte stream that can be stored on disk or sent over a network. Later on, this character stream can then be retrieved and de-serialized back to a Python object.Here, DT is our decision tree model saving as fdemand.pkl file. Wb is the write binary in bytes.

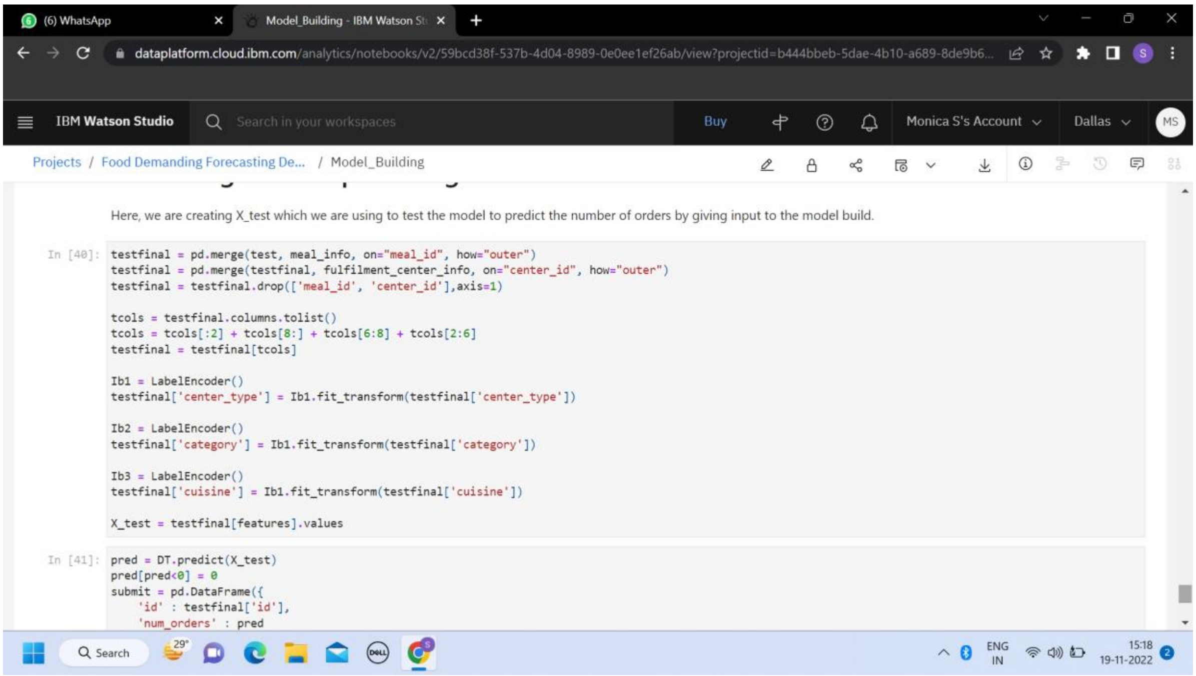
```
In [39]: import pickle
         pickle.dump(DT,open('fdemand.pkl','wb'))
```

## Predicting The Output Using The Model

Here, we are creating X_test which we are using to test the model to predict the number of orders by giving input to the model build.

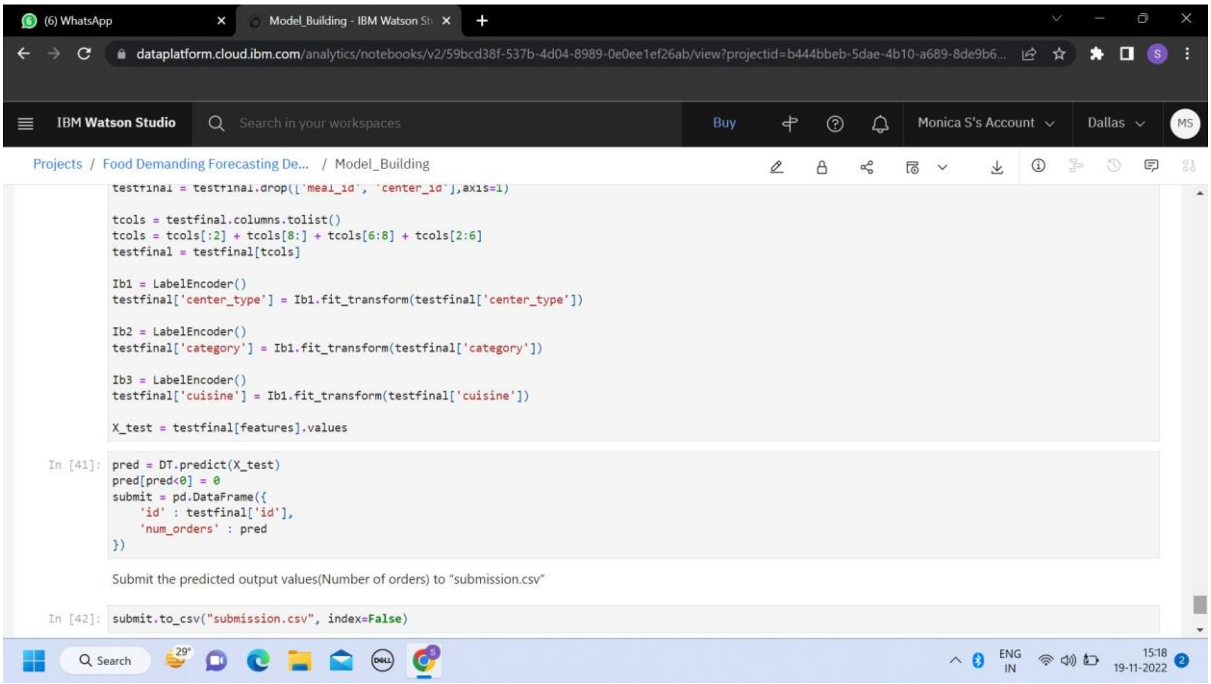
```
In [40]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
         testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
         testfinal = testfinal.drop(['meal_id', 'center_id'],axis=1)

         tcols = testfinal.columns.tolist()
         tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
         testfinal = testfinal[tcols]
```

Search
29°
ENG
IN
15:18
19-11-2022


(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

Here, we are creating X_test which we are using to test the model to predict the number of orders by giving input to the model build.

```
In [40]: testfinal = pd.merge(test, meal_info, on="meal_id", how="outer")
         testfinal = pd.merge(testfinal, fulfilment_center_info, on="center_id", how="outer")
         testfinal = testfinal.drop(['meal_id', 'center_id'],axis=1)

         tcols = testfinal.columns.tolist()
         tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
         testfinal = testfinal[tcols]

         Ib1 = LabelEncoder()
         testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

         Ib2 = LabelEncoder()
         testfinal['category'] = Ib1.fit_transform(testfinal['category'])

         Ib3 = LabelEncoder()
         testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

         X_test = testfinal[features].values

In [41]: pred = DT.predict(X_test)
         pred[pred<0] = 0
         submit = pd.DataFrame({
             'id' : testfinal['id'],
             'num_orders' : pred
```

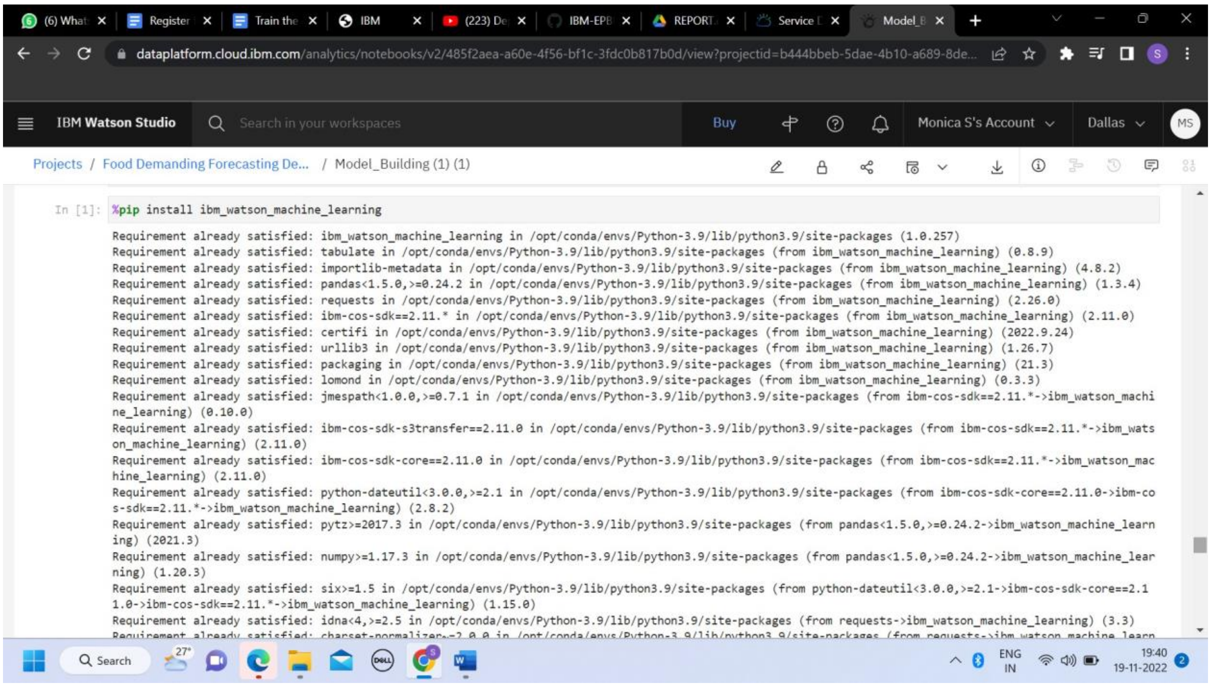

(6) WhatsApp
Model_Building - IBM Watson St
dataplatform.cloud.ibm.com/analytics/notebooks/v2/59bcd38f-537b-4d04-8989-0e0ee1ef26ab/view?projectid=b444bbeb-5dae-4b10-a689-8de9b6...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building

```
testfinal = testfinal.drop(['meal_id', 'center_id'],axis=1)

tcols = testfinal.columns.tolist()
tcols = tcols[:2] + tcols[8:] + tcols[6:8] + tcols[2:6]
testfinal = testfinal[tcols]

Ib1 = LabelEncoder()
testfinal['center_type'] = Ib1.fit_transform(testfinal['center_type'])

Ib2 = LabelEncoder()
testfinal['category'] = Ib1.fit_transform(testfinal['category'])

Ib3 = LabelEncoder()
testfinal['cuisine'] = Ib1.fit_transform(testfinal['cuisine'])

X_test = testfinal[features].values
```

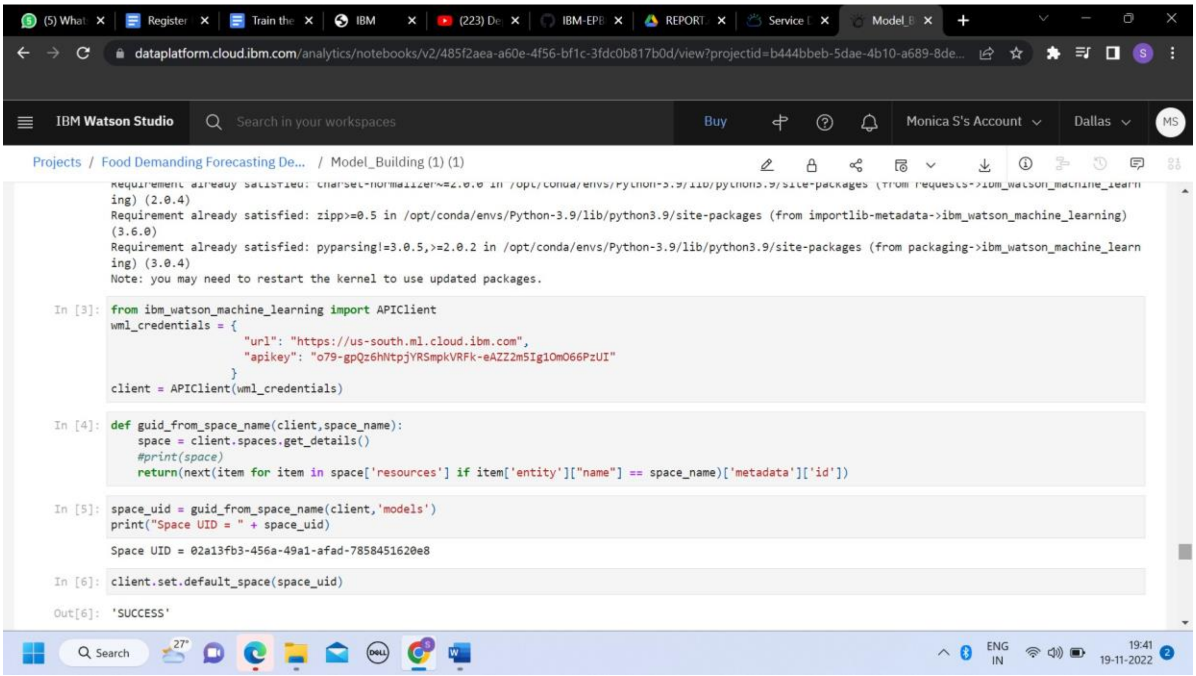
```
In [41]: pred = DT.predict(X_test)
         pred[pred<0] = 0
         submit = pd.DataFrame({
             'id' : testfinal['id'],
             'num_orders' : pred
         })
```

Submit the predicted output values(Number of orders) to "submission.csv"

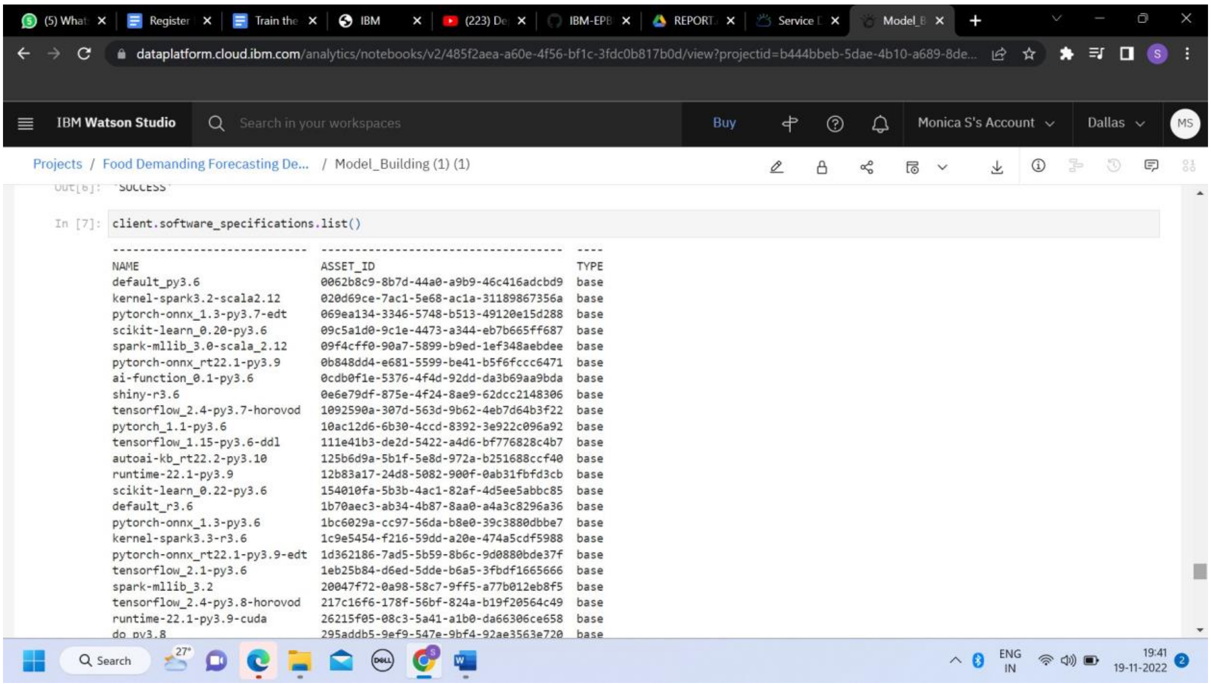
```
In [42]: submit.to_csv("submission.csv", index=False)
```


(6) What
Register
Train the
IBM
(223) De
IBM-EPB
REPORT
Service
Model_B
dataplatform.cloud.ibm.com/analytics/notebooks/v2/485f2aea-a60e-4f56-bf1c-3fdc0b817b0d/view?projectid=b444bbeb-5dae-4b10-a689-8de...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building (1) (1)
Search
27°
ENG
IN
19:40
19-11-2022

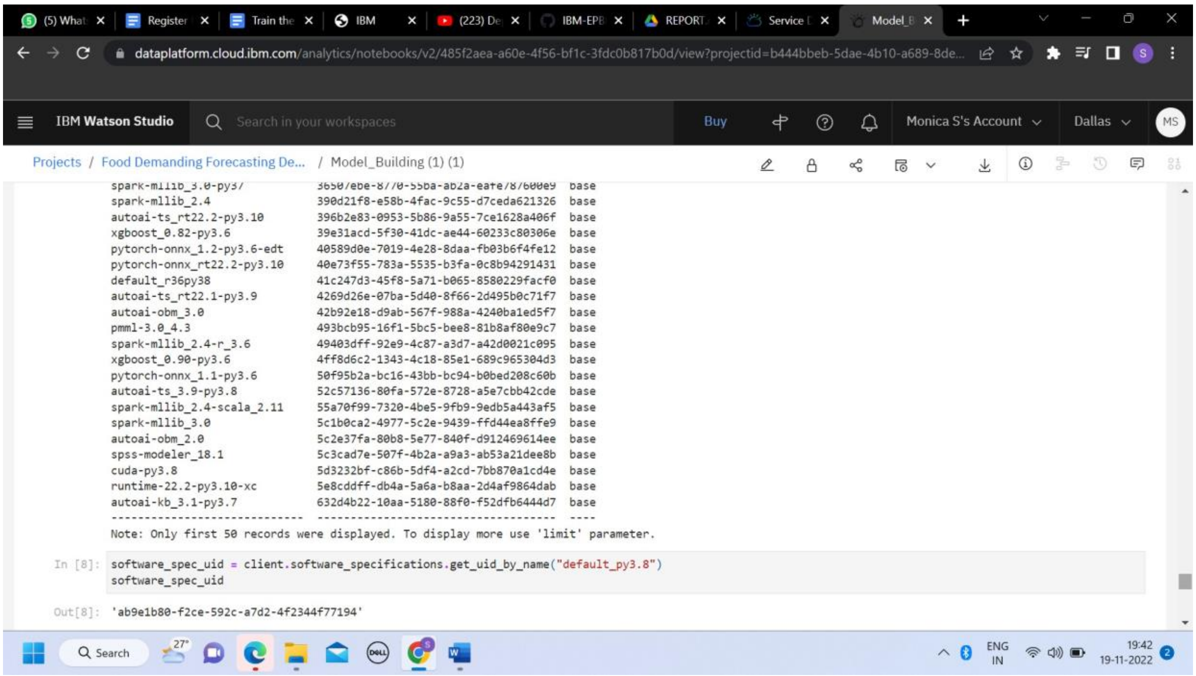
```
In [1]: %pip install ibm_watson_machine_learning

Requirement already satisfied: ibm_watson_machine_learning in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.257)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (0.8.9)
Requirement already satisfied: importlib-metadata in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (4.8.2)
Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (1.3.4)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2.26.0)
Requirement already satisfied: ibm-cos-sdk==2.11.* in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2.11.0)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2022.9.24)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (1.26.7)
Requirement already satisfied: packaging in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (21.3)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (0.3.3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_watson_machi
ne_learning) (0.10.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_wats
on_machine_learning) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_watson_mac
hine_learning) (2.11.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk-core==2.11.0->ibm-co
s-sdk==2.11.*->ibm_watson_machine_learning) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm_watson_machine_learn
ing) (2021.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm_watson_machine_lear
ning) (1.20.3)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->ibm-cos-sdk-core==2.1
1.0->ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (1.15.0)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm_watson_machine_learning) (3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm_watson_machine_learn
```


(5) What
Register
Train the
IBM
(223) De
IBM-EPB
REPORT
Service
Model_B
dataplatform.cloud.ibm.com/analytics/notebooks/v2/485f2aea-a60e-4f56-bf1c-3fdc0b817b0d/view?projectid=b444bbeb-5dae-4b10-a689-8de...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
MS
Projects / Food Demanding Forecasting De... / Model_Building (1) (1)
Search
27°
ENG
IN
19:41
19-11-2022

```
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm_watson_machine_learn
ing) (2.0.4)
Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from importlib-metadata->ibm_watson_machine_learning)
(3.6.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from packaging->ibm_watson_machine_learn
ing) (3.0.4)
Note: you may need to restart the kernel to use updated packages.

In [3]: from ibm_watson_machine_learning import APIClient
        wml_credentials = {
                            "url": "https://us-south.ml.cloud.ibm.com",
                            "apikey": "o79-gpQz6hNtpjYRSmpkVRFk-eAZZ2m5Ig1OmO66PzUI"
                          }
        client = APIClient(wml_credentials)

In [4]: def guid_from_space_name(client,space_name):
            space = client.spaces.get_details()
            #print(space)
            return(next(item for item in space['resources'] if item['entity']["name"] == space_name)['metadata']['id'])

In [5]: space_uid = guid_from_space_name(client,'models')
        print("Space UID = " + space_uid)

        Space UID = 02a13fb3-456a-49a1-afad-7858451620e8

In [6]: client.set.default_space(space_uid)

Out[6]: 'SUCCESS'
```


dataplatform.cloud.ibm.com/analytics/notebooks/v2/485f2aea-a60e-4f56-bf1c-3fdc0b817b0d/view?projectid=b444bbeb-5dae-4b10-a689-8de...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
Projects / Food Demanding Forecasting De... / Model_Building (1) (1)
19:41
19-11-2022

```
Out[6]: 'SUCCESS'

In [7]: client.software_specifications.list()

        ----------------------------  ------------------------------------  ----
        NAME                          ASSET_ID                              TYPE
        default_py3.6                 0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
        kernel-spark3.2-scala2.12     020d69ce-7ac1-5e68-ac1a-31189867356a  base
        pytorch-onnx_1.3-py3.7-edt    069ea134-3346-5748-b513-49120e15d288  base
        scikit-learn_0.20-py3.6       09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
        spark-mllib_3.0-scala_2.12    09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
        pytorch-onnx_rt22.1-py3.9     0b848dd4-e681-5599-be41-b5f6fccc6471  base
        ai-function_0.1-py3.6         0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
        shiny-r3.6                    0e6e79df-875e-4f24-8ae9-62dcc2148306  base
        tensorflow_2.4-py3.7-horovod  1092590a-307d-563d-9b62-4eb7d64b3f22  base
        pytorch_1.1-py3.6             10ac12d6-6b30-4ccd-8392-3e922c096a92  base
        tensorflow_1.15-py3.6-ddl     111e41b3-de2d-5422-a4d6-bf776828c4b7  base
        autoai-kb_rt22.2-py3.10       125b6d9a-5b1f-5e8d-972a-b251688ccf40  base
        runtime-22.1-py3.9            12b83a17-24d8-5082-900f-0ab31fbfd3cb  base
        scikit-learn_0.22-py3.6       154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
        default_r3.6                  1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
        pytorch-onnx_1.3-py3.6        1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
        kernel-spark3.3-r3.6          1c9e5454-f216-59dd-a20e-474a5cdf5988  base
        pytorch-onnx_rt22.1-py3.9-edt 1d362186-7ad5-5b59-8b6c-9d0880bde37f  base
        tensorflow_2.1-py3.6          1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
        spark-mllib_3.2               20047f72-0a98-58c7-9ff5-a77b012eb8f5  base
        tensorflow_2.4-py3.8-horovod  217c16f6-178f-56bf-824a-b19f20564c49  base
        runtime-22.1-py3.9-cuda       26215f05-08c3-5a41-a1b0-da66306ce658  base
        do_py3.8                      295addb5-9ef9-547e-9bf4-92ae3563e720  base
```


dataplatform.cloud.ibm.com/analytics/notebooks/v2/485f2aea-a60e-4f56-bf1c-3fdc0b817b0d/view?projectid=b444bbeb-5dae-4b10-a689-8de...
IBM Watson Studio
Search in your workspaces
Buy
Monica S's Account
Dallas
Projects / Food Demanding Forecasting De... / Model_Building (1) (1)
19:42
19-11-2022

```
        spark-mllib_3.0-py37          36507ebe-8770-55ba-ab2a-eafe787600e9  base
        spark-mllib_2.4               390d21f8-e58b-4fac-9c55-d7ceda621326  base
        autoai-ts_rt22.2-py3.10       396b2e83-0953-5b86-9a55-7ce1628a406f  base
        xgboost_0.82-py3.6            39e31acd-5f30-41dc-ae44-60233c80306e  base
        pytorch-onnx_1.2-py3.6-edt    40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
        pytorch-onnx_rt22.2-py3.10    40e73f55-783a-5535-b3fa-0c8b94291431  base
        default_r36py38               41c247d3-45f8-5a71-b065-8580229facf0  base
        autoai-ts_rt22.1-py3.9        4269d26e-07ba-5d40-8f66-2d495b0c71f7  base
        autoai-obm_3.0                42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
        pmml-3.0_4.3                  493bcb95-16f1-5bc5-bee8-81b8af80e9c7  base
        spark-mllib_2.4-r_3.6         49403dff-92e9-4c87-a3d7-a42d0021c095  base
        xgboost_0.90-py3.6            4ff8d6c2-1343-4c18-85e1-689c965304d3  base
        pytorch-onnx_1.1-py3.6        50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
        autoai-ts_3.9-py3.8           52c57136-80fa-572e-8728-a5e7cbb42cde  base
        spark-mllib_2.4-scala_2.11    55a70f99-7320-4be5-9fb9-9edb5a443af5  base
        spark-mllib_3.0               5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9  base
        autoai-obm_2.0                5c2e37fa-80b8-5e77-840f-d912469614ee  base
        spss-modeler_18.1             5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
        cuda-py3.8                    5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e  base
        runtime-22.2-py3.10-xc        5e8cddff-db4a-5a6a-b8aa-2d4af9864dab  base
        autoai-kb_3.1-py3.7           632d4b22-10aa-5180-88f0-f52dfb6444d7  base
        ----------------------------  ------------------------------------  ----
        Note: Only first 50 records were displayed. To display more use 'limit' parameter.

In [8]: software_spec_uid = client.software_specifications.get_uid_by_name("default_py3.8")
        software_spec_uid

Out[8]: 'ab9e1b80-f2ce-592c-a7d2-4f2344f77194'
```