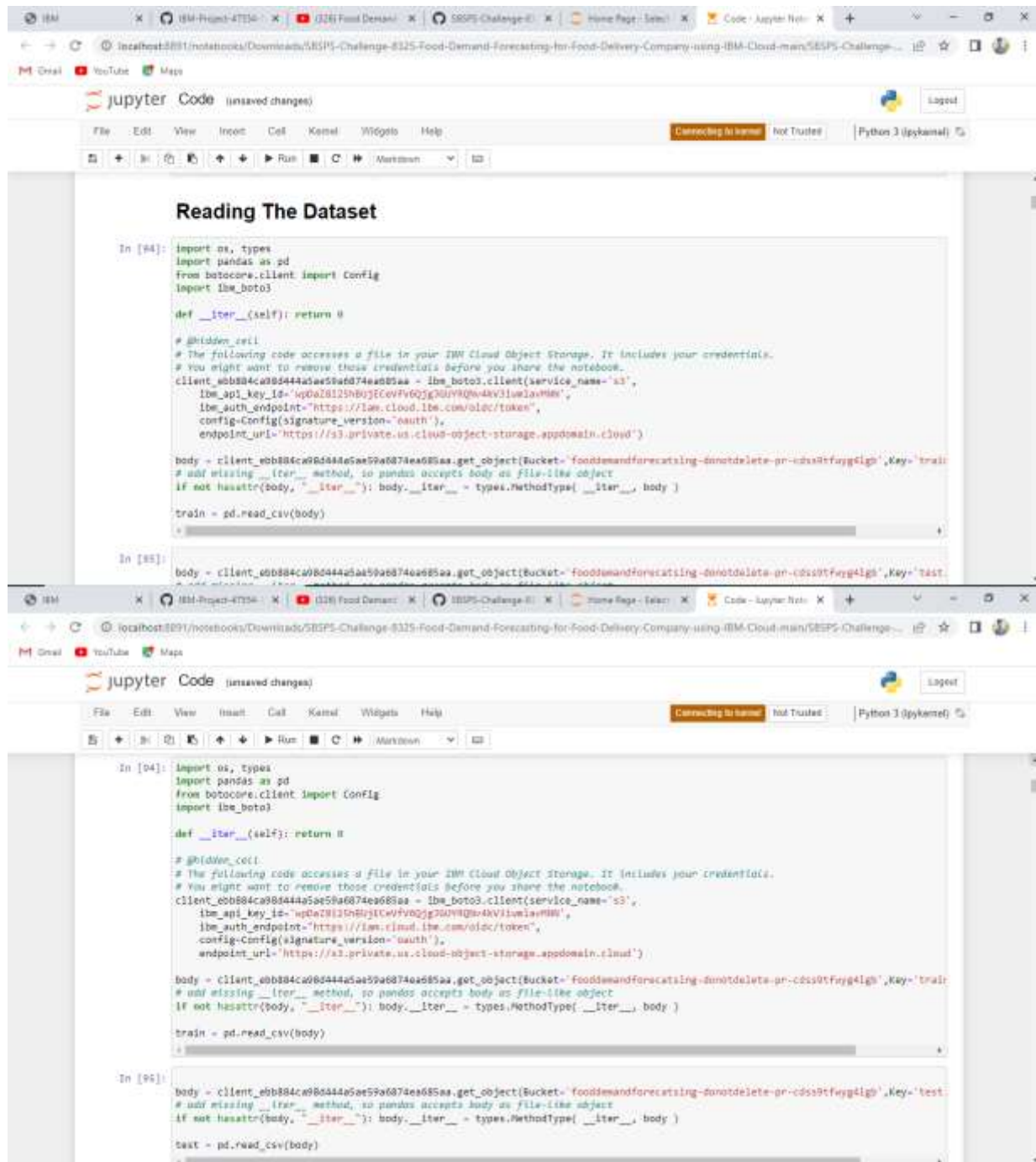


TEAM ID: PNT2022TMID32368

PROJECT NAME: DemandEst - AI powered Food Demand Forecaster

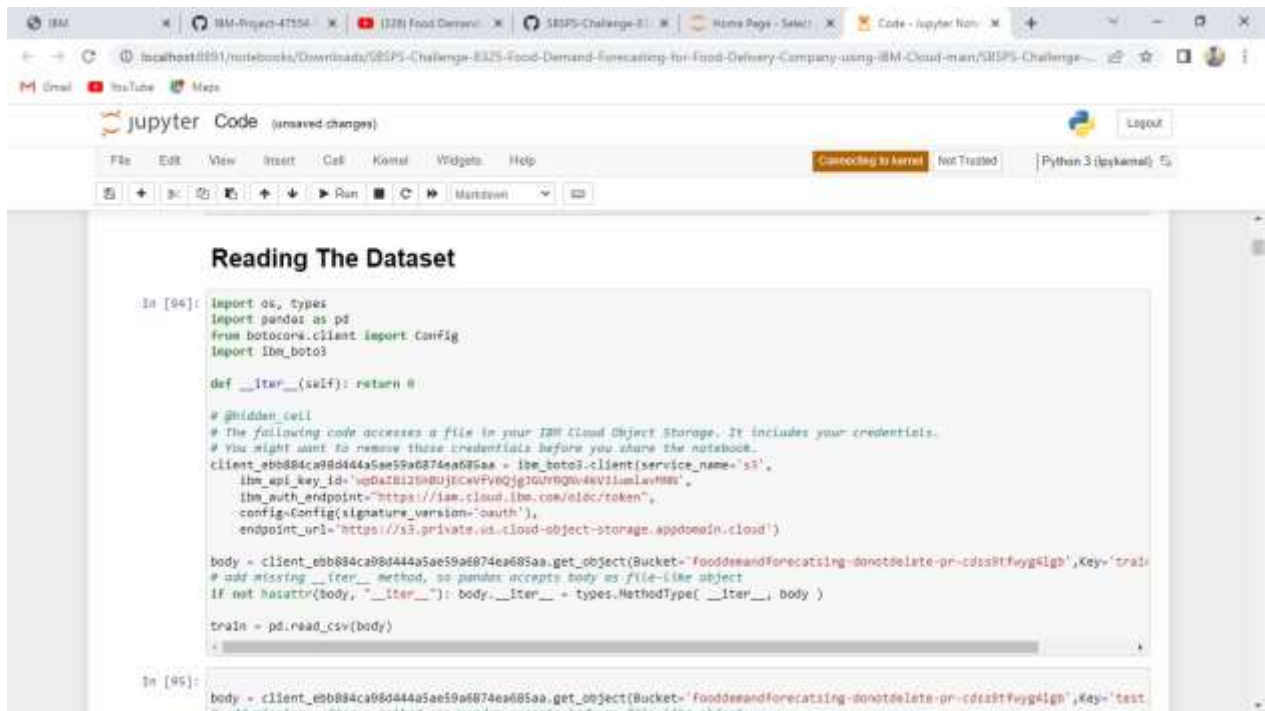


The image displays two screenshots of a Jupyter Notebook interface, showing the process of reading a dataset from IBM Cloud Object Storage.

Top Screenshot: The notebook is titled "Reading The Dataset". The code in cell [94] imports the necessary libraries: `os`, `types`, `pandas` as `pd`, `botocore.client` as `Config`, and `ibm_botoc`. It defines a custom `__iter__` method for the `body` object to return an iterator. The code then uses `client.get_object` to retrieve the dataset from the bucket `fooddemandforecasting-donotdelete-pr-cds9tfygg4lg` with the key `train`. The dataset is read into a `pandas` DataFrame using `pd.read_csv(body)`.

Bottom Screenshot: This screenshot shows the continuation of the code. In cell [95], the `body` object is used to retrieve the dataset from the bucket `fooddemandforecasting-donotdelete-pr-cds9tfygg4lg` with the key `test`. The dataset is read into a `pandas` DataFrame using `pd.read_csv(body)`.

Team Member 1



The screenshot shows a Jupyter Notebook interface with the title "Reading The Dataset". The code in the first cell (In [94]:) imports the necessary libraries (os, types, pandas, boto3) and defines a function to read a CSV file from IBM Cloud Object Storage. The function uses the boto3 client to get an object from the 'fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh' bucket. The code then reads the CSV file into a pandas DataFrame and returns it. The second cell (In [95]:) shows the same code being executed, with the output being a pandas DataFrame.

```
In [94]: import os, types
import pandas as pd
from boto3.client import Config
import boto3

def __iter__(self): return #

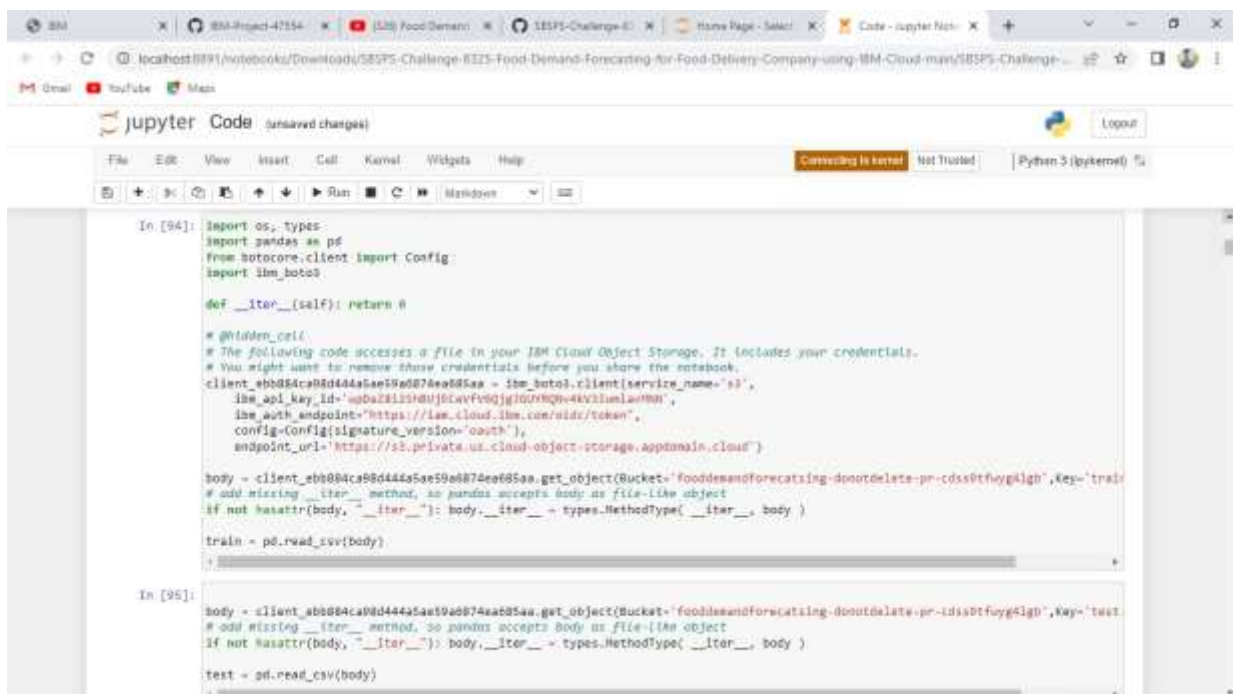
# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove these credentials before you share the notebook.
client_ebb884ca98d44a5ae59a6874ea685aa = boto3.client(service_name='s3',
    aws_api_key_id='wba28123f0u1b1avf0g10zmqn4kV11um1awm',
    aws_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')

body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='train')
# 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)

In [95]: body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='test')
# 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)
```



The screenshot shows a Jupyter Notebook interface with the title "Reading The Dataset". The code in the first cell (In [94]:) imports the necessary libraries (os, types, pandas, boto3) and defines a function to read a CSV file from IBM Cloud Object Storage. The function uses the boto3 client to get an object from the 'fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh' bucket. The code then reads the CSV file into a pandas DataFrame and returns it. The second cell (In [95]:) shows the same code being executed, with the output being a pandas DataFrame.

```
In [94]: import os, types
import pandas as pd
from boto3.client import Config
import boto3

def __iter__(self): return #

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove these credentials before you share the notebook.
client_ebb884ca98d44a5ae59a6874ea685aa = boto3.client(service_name='s3',
    aws_api_key_id='wba28123f0u1b1avf0g10zmqn4kV11um1awm',
    aws_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')

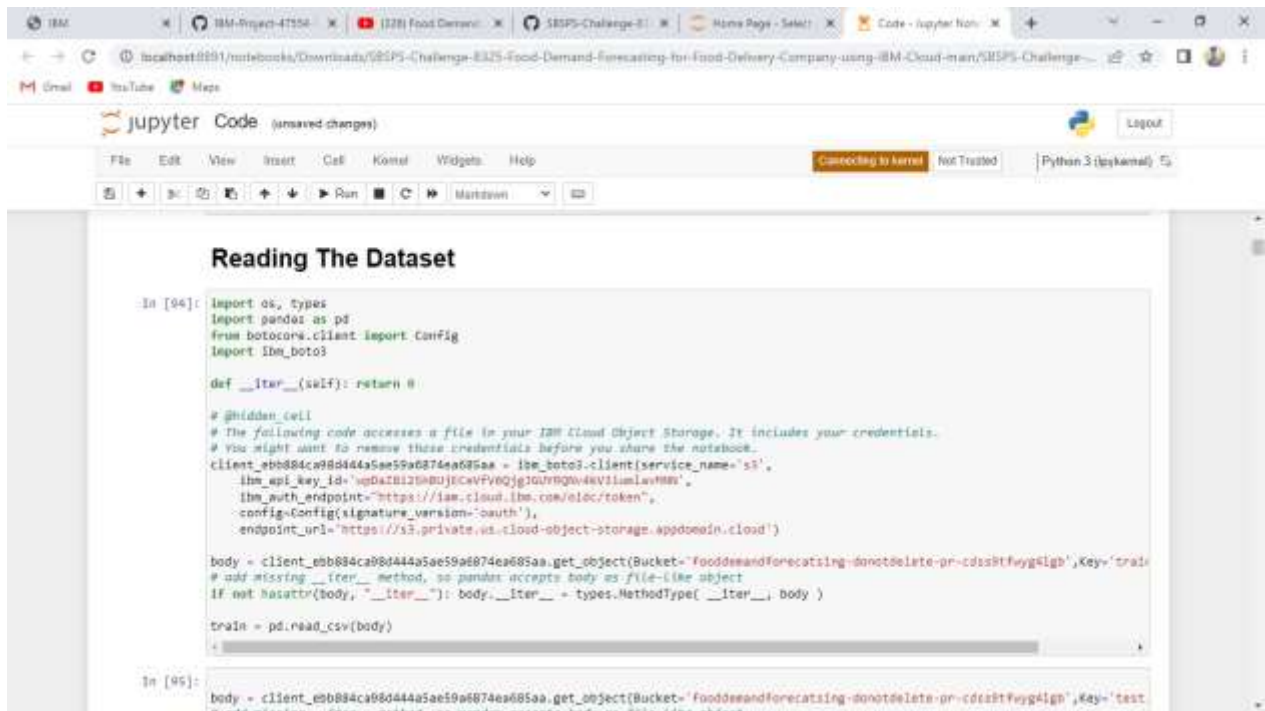
body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='train')
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, '__iter__'): body.__iter__ = types.MethodType(__iter__, body)

train = pd.read_csv(body)

In [95]: body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='test')
# 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)
```

Team Member 2



The screenshot shows a Jupyter Notebook interface with the title "Reading The Dataset". The code in cell [94] imports the necessary libraries and sets up the IBM Cloud Object Storage client. It then defines a function to read a CSV file from the storage and loads it into a pandas DataFrame.

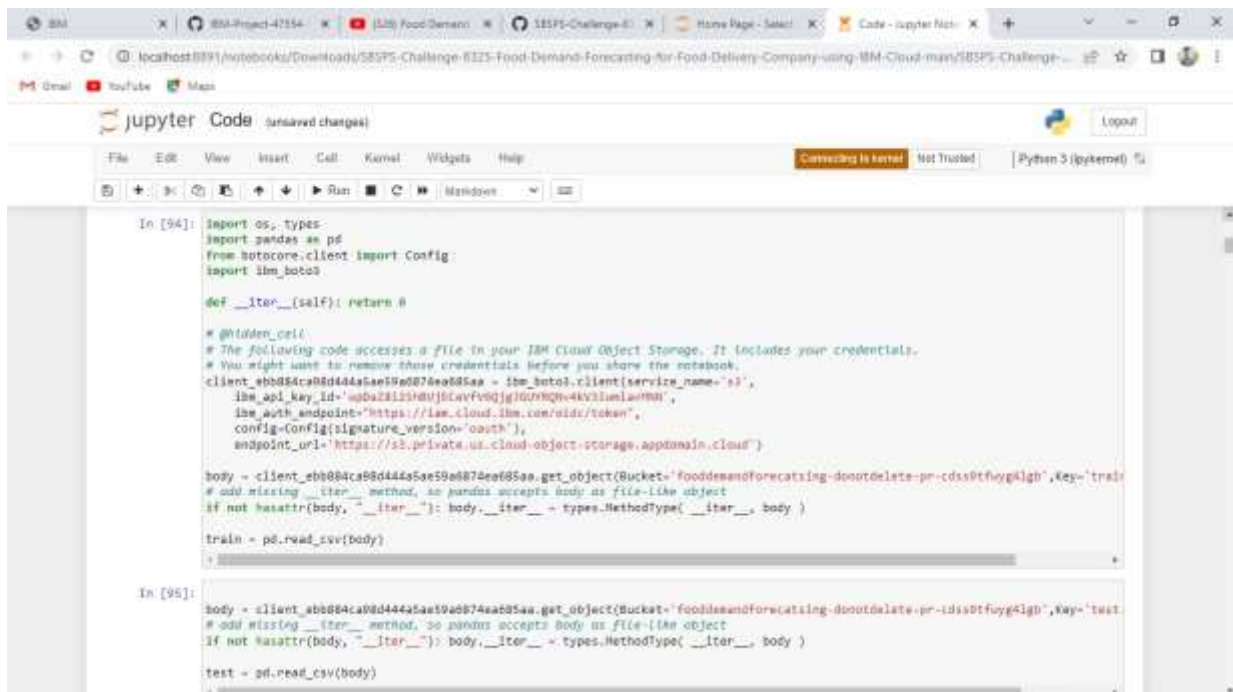
```
In [94]: import os, types
import pandas as pd
from boto3.client import Config
import ibm_boto3

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 these credentials before you share the notebook.
client_ebb884ca98d44a5ae59a6874ea685aa = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='wba28123f0u1b1avf0g10zmqn4kv3iim1awm',
    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')

body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='train')
# 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)
```



The screenshot shows a Jupyter Notebook interface with the title "Reading The Dataset". The code in cell [94] imports the necessary libraries and sets up the IBM Cloud Object Storage client. It then defines a function to read a CSV file from the storage and loads it into a pandas DataFrame. Cell [95] shows the next step, where the test data is loaded into a pandas DataFrame.

```
In [94]: import os, types
import pandas as pd
from boto3.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 these credentials before you share the notebook.
client_ebb884ca98d44a5ae59a6874ea685aa = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='wba28123f0u1b1avf0g10zmqn4kv3iim1awm',
    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')

body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='train')
# 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)
```

```
In [95]: body = client_ebb884ca98d44a5ae59a6874ea685aa.get_object(Bucket='fooddemandforecasting-donotdelete-pr-cdss8tfuyg4lgh',Key='test')
# 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)
```

Team Member 3

[illegible]

```
BM-Project-47154 X [L2] Food-Dem... X [L2] SESPS-Challenge-6125 Food-Demand-Forecasting for Food-Delivery Company using IBM Cloud-man/SESPS-Challenge-... X Home Page - Select X Code - Jupyter Note... X Logout
```

localhost8891/notebooks/Downloads/SESPS-Challenge-6125-Food-Demand-Forecasting-for-Food-Delivery-Company-using-IBM-Cloud-man/SESPS-Challenge-...

Gmail YouTube Maps

Jupyter Code (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Connecting to kernel Not Trusted Python 3 (ipykernel) 7.0

```
In [94]: import os, types
import pandas as pd
from ibmcloud.client import Config
import ibm_botoc

def __iter__(self): return #

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove these credentials before you share the notebook.
client_abb884ca9b8444a5ae59a6874ea685aa = ibm_botoc.client(service_name='s3',
    ibm_api_key_id='wpa281258Uj8TavFv0qj30Uv8Qv4kV3lunjaRMH',
    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')

body = client_abb884ca9b8444a5ae59a6874ea685aa.get_object(bucket='fooddemandforecasting-doondelete-pr-cds8tfuygkgn',key='train')
# 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)

In [95]: body = client_abb884ca9b8444a5ae59a6874ea685aa.get_object(bucket='fooddemandforecasting-doondelete-pr-cds8tfuygkgn',key='test')
# 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)
```