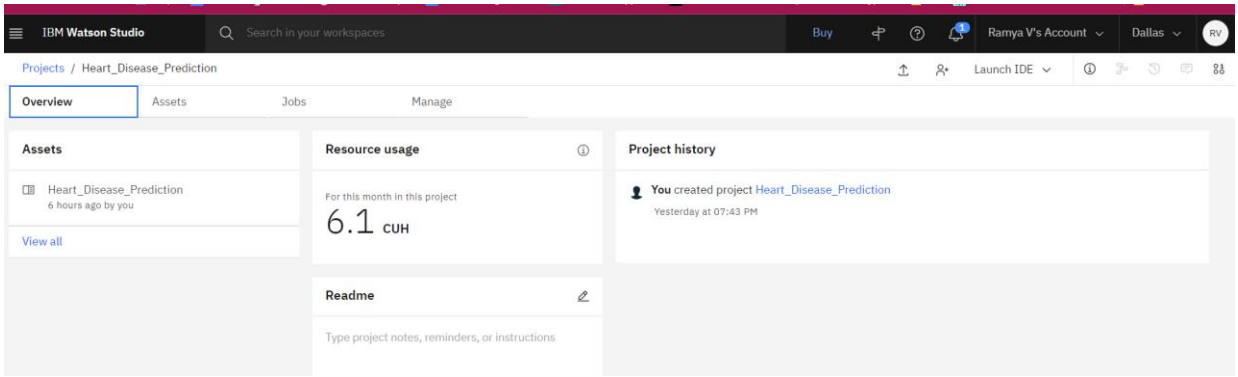
Sprint 4

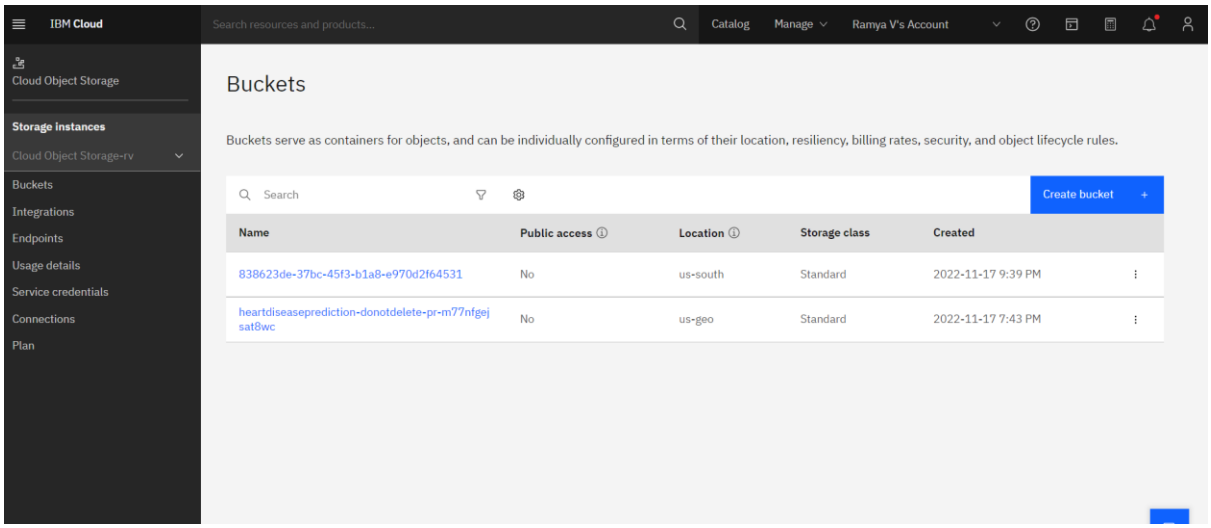
Heart Disease prediction Using ML Model in IBM Cloud

Date	18 November 2022
Team ID	PNT2022TMID52957
Project Name	Visualizing and Predicting Heart Diseases with an Interactive Dashboard

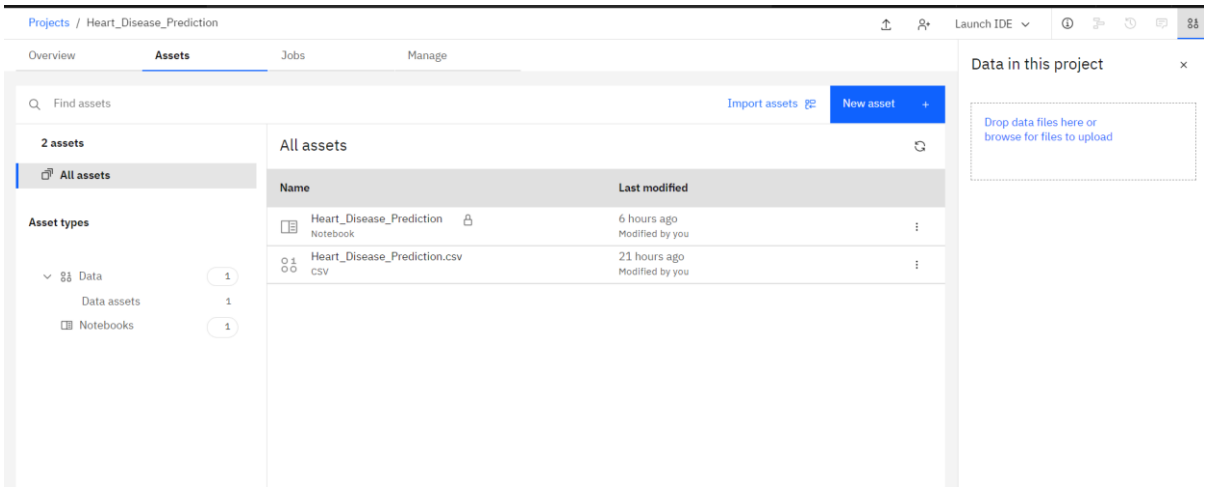
Step 1: Create Watson Studio project



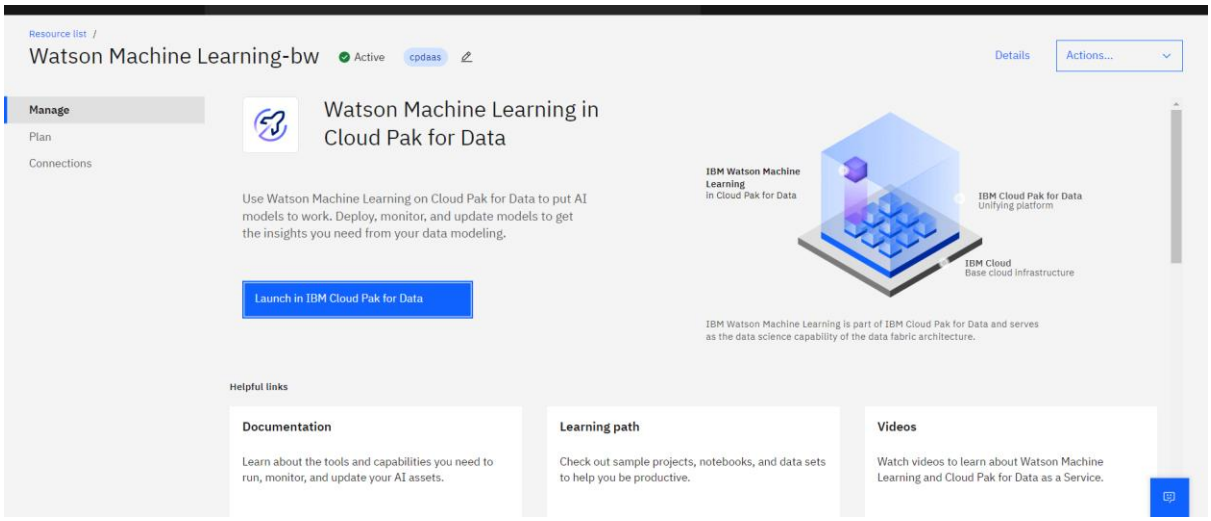
Step 2: Create Cloud Object Storage



Step 3: From project Upload Dataset and Machine Learning Model

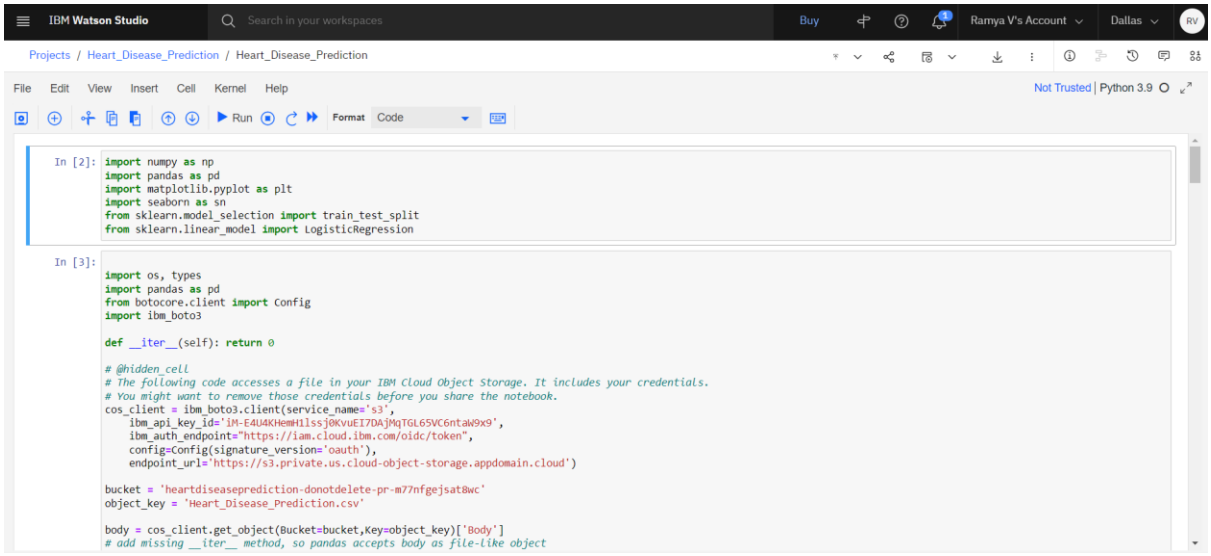


Step 4: Create Watson Machine Learning Account



Step 5: Train ML Model in IBM Cloud

Step 1: Import Libraries and Upload data in IBM Watson Studio And Connect it to ML Model



Projects / Heart_Disease_Prediction / Heart_Disease_Prediction

File Edit View Insert Cell Kernel Help

Not Trusted | Python 3.

Run Format Code

```
# @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='iM-E4U4KHemH1lssj0KvuEI7DAjMqTGL65VC6ntaW9x9',
    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 = 'heartdiseaseprediction-donotdelete-pr-m77nfgejsat8wc'
object_key = 'Heart_Disease_Prediction.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 )

data = pd.read_csv(body)
data.head()
```

Out[3]:

	Age	Sex	Chest pain type	BP	Cholesterol	FBS over 120	EKG results	Max HR	Exercise angina	ST depression	Slope of ST	Number of vessels fluro	Thallium	Heart Disease
0	70	1	4	130	322	0	2	109	0	2.4	2	3	3	Presence
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7	Absence
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	Presence
3	64	1	4	128	263	0	0	105	1	0.2	2	1	7	Absence

In [4]:
```
data
```

Out[4]:

	Age	Sex	Chest pain type	BP	Cholesterol	FBS over 120	EKG results	Max HR	Exercise angina	ST depression	Slope of ST	Number of vessels fluro	Thallium	Heart Disease
0	70	1	4	130	322	0	2	109	0	2.4	2	3	3	Presence
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7	Absence
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	Presence
3	64	1	4	128	263	0	0	105	1	0.2	2	1	7	Absence
4	74	0	2	120	269	0	2	121	1	0.2	1	1	3	Absence
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
265	52	1	3	172	199	1	0	162	0	0.5	1	0	7	Absence
266	44	1	2	120	263	0	0	173	0	0.0	1	0	7	Absence
267	56	0	2	140	294	0	2	153	0	1.3	2	0	3	Absence
268	57	1	4	140	192	0	0	148	0	0.4	2	0	6	Absence
269	67	1	4	160	286	0	2	108	1	1.5	2	3	3	Presence

270 rows × 14 columns

In [29]:
```
X= data[['Age','Sex','Chest pain type','BP','Cholesterol','FBS over 120','EKG results','Max HR','Exercise angina','ST depression','Slope of ST','Number of vessels fluro',
        'Thallium']].to_numpy()
Y= data['Heart Disease'].to_numpy()
```

Step 2: Train and Test ML Model and Check For Accuracy

IBM Watson Studio Search in your workspaces Buy Ramya V's Account Dallas

Projects / Heart_Disease_Prediction / Heart_Disease_Prediction

File Edit View Insert Cell Kernel Help

Not Trusted | Python 3.9

Run Format Code

In [31]:
```
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.05)
```

In [32]:
```
X_train.shape
```

Out[32]: (256, 13)

In [33]:
```
log_reg= LogisticRegression()
log_reg.fit(X_train,Y_train)
Y_pred=log_reg.predict(X_test)
```

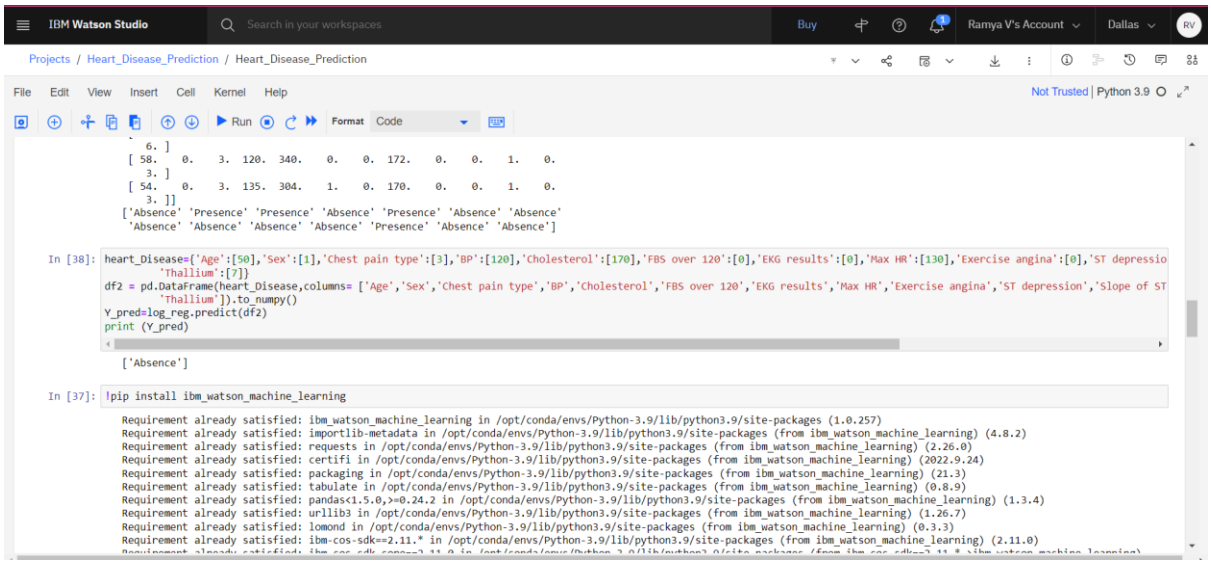
In [34]:
```
from sklearn.metrics import accuracy_score
print ("Accuracy : ", accuracy_score(Y_test,Y_pred))
```

```
Accuracy :  0.9285714285714286
```

In [35]:
```
print (X_test)
print (Y_pred)
```

```
[[ 43.   1.   4. 110. 211.   0.   0. 161.   0.   0.   1.   0.
   7. ]
 [ 65.   1.   4. 110. 248.   0.   2. 158.   0.   0.6  1.   2.
   6. ]
 [ 64.   1.   4. 145. 212.   0.   2. 132.   0.   2.   2.   2.
   6. ]
 [ 70.   1.   2. 156. 245.   0.   2. 143.   0.   0.   1.   0.
   3. ]
 [ 54.   1.   4. 124. 266.   0.   2. 109.   1.   2.2  2.   1.
   7. ]
```

Step 3: Check with varies sample data



Step 6: Connecting a Machine Learning Service as an API Client

Step 1: Create API Key in IBM Cloud

```
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_learning) (3.0.4)
```

In [39]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials={
    "url":"https://us-south.ml.cloud.ibm.com",
    "apikey":"YgTKlJaXJfw0Buhk6_FCGdcAAtzUwBNBznleRI9fJ-KB"
}
client=APIClient(wml_credentials)
```

In [40]:
```
def guid_from_space_name(client,space_name):
    space=client.spaces.get_details()
    return(next(item for item in space['resources'] if item['entity']['name']==space_name)['metadata']['id'])
```

In [41]:
```
space_uid=guid_from_space_name(client,'Heart Diseases Prediction Model')
print("Space UID="+space_uid)
```

```
Space UID=d18c6095-1931-4722-9c35-d10688009ffa
```

In [42]:
```
client.set.default_space(space_uid)
```

Out[42]: 'SUCCESS'

In [43]:
```
client.software_specifications.list()
```

```
------------------------------  ------------------------------------  ----
```

Step 2: Deploy Model


```
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_learning) (3.0.4)
```

In [39]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials={
    "url":"https://us-south.ml.cloud.ibm.com",
    "apikey":"YgTKlJaXJfw0Buhk6_FCGdcAAtzUwBNBznleRI9fJ-KB"
}
client=APIClient(wml_credentials)
```

In [40]:
```
def guid_from_space_name(client,space_name):
    space=client.spaces.get_details()
    return(next(item for item in space['resources'] if item['entity']['name']==space_name)['metadata']['id'])
```

In [41]:
```
space_uid=guid_from_space_name(client,'Heart Diseases Prediction Model')
print("Space UID="+space_uid)
```

```
Space UID=d18c6095-1931-4722-9c35-d10688009ffa
```

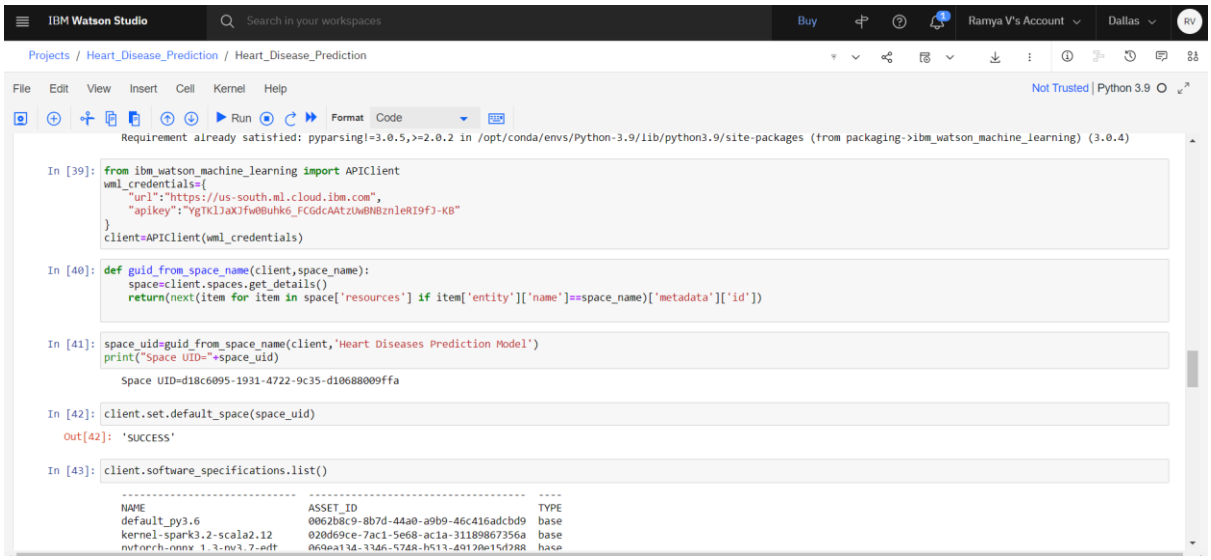
In [42]:
```
client.set.default_space(space_uid)
```

Out[42]: 'SUCCESS'

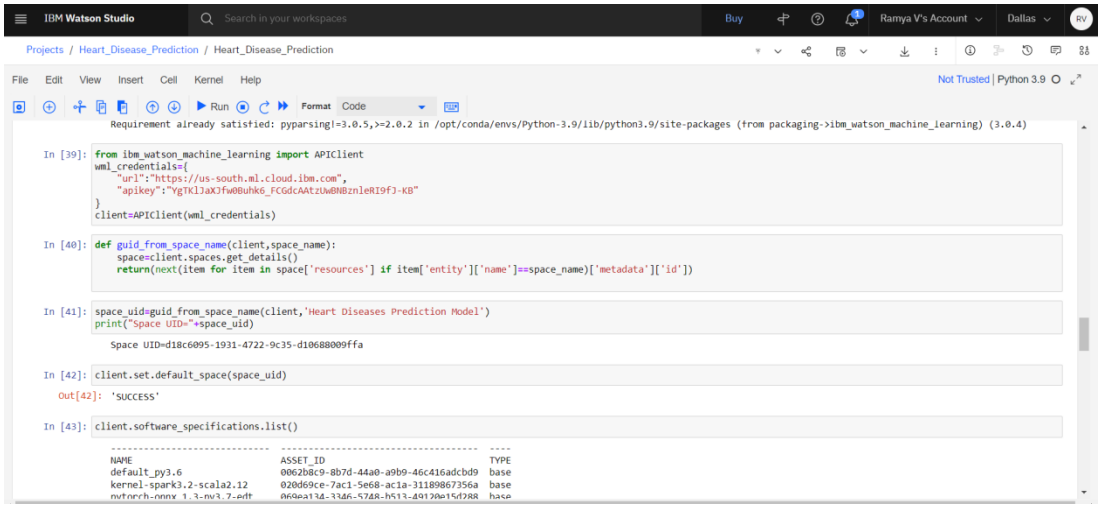
In [43]:
```
client.software_specifications.list()
```

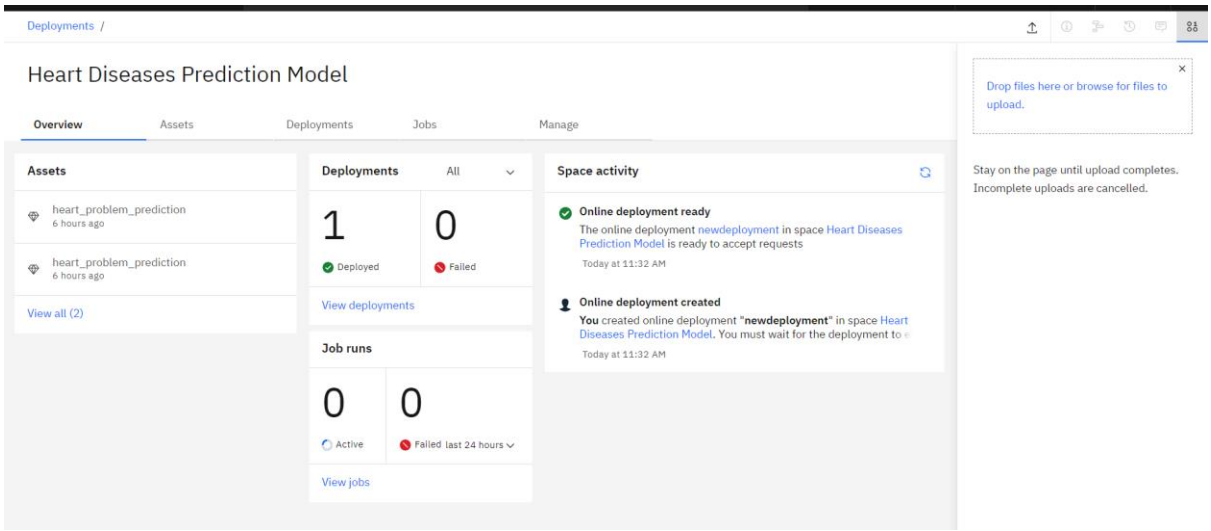
```
------------------------------  ------------------------------------  ----
```

Step 7: Create Deployment Space Where Model will be deployed

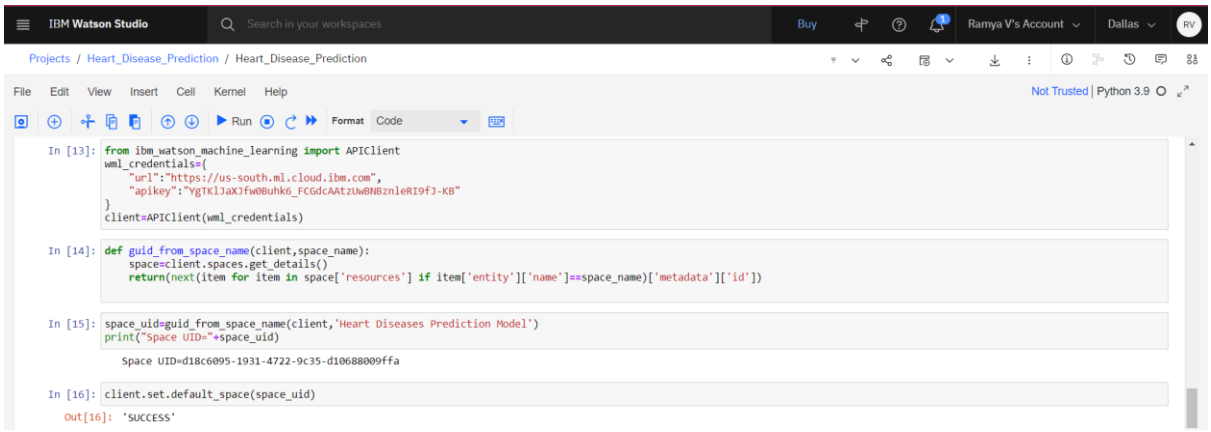


Step 8: Create Deployment Space as Heart Diseases Prediction Model

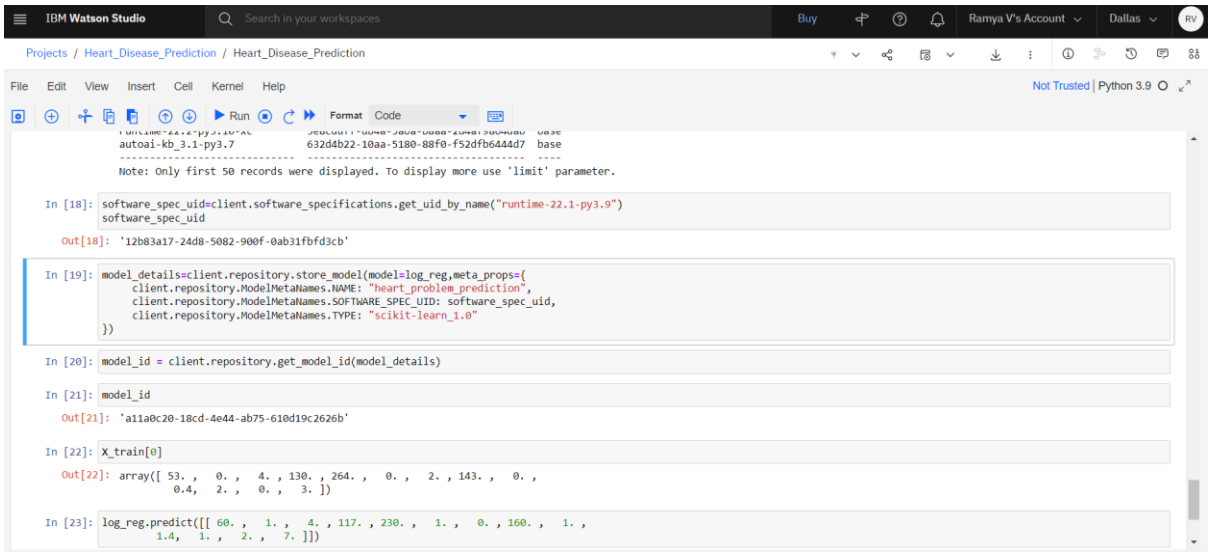




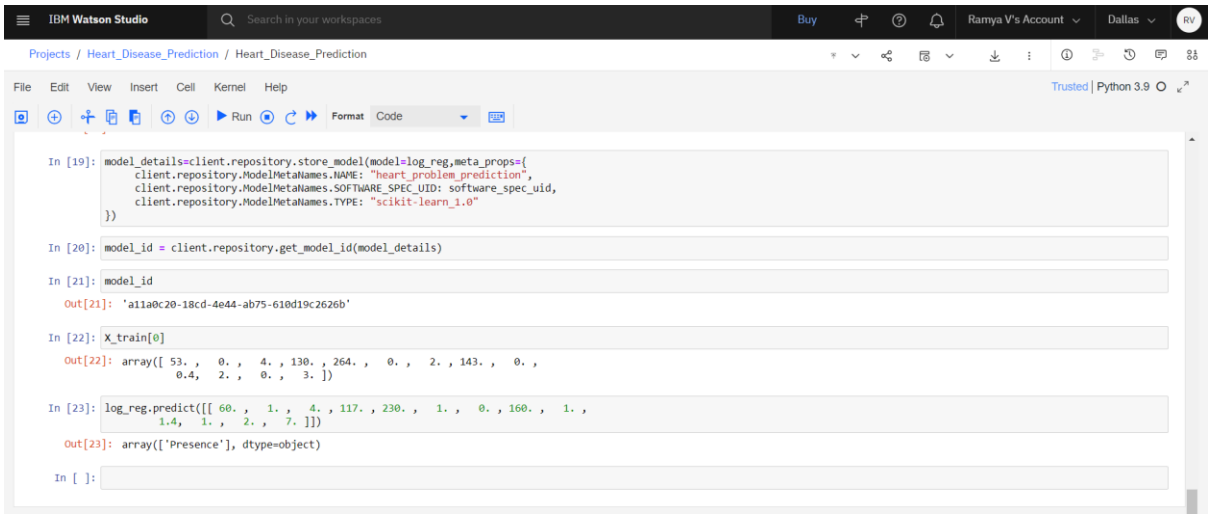
Step 9: Make the Space id as default one



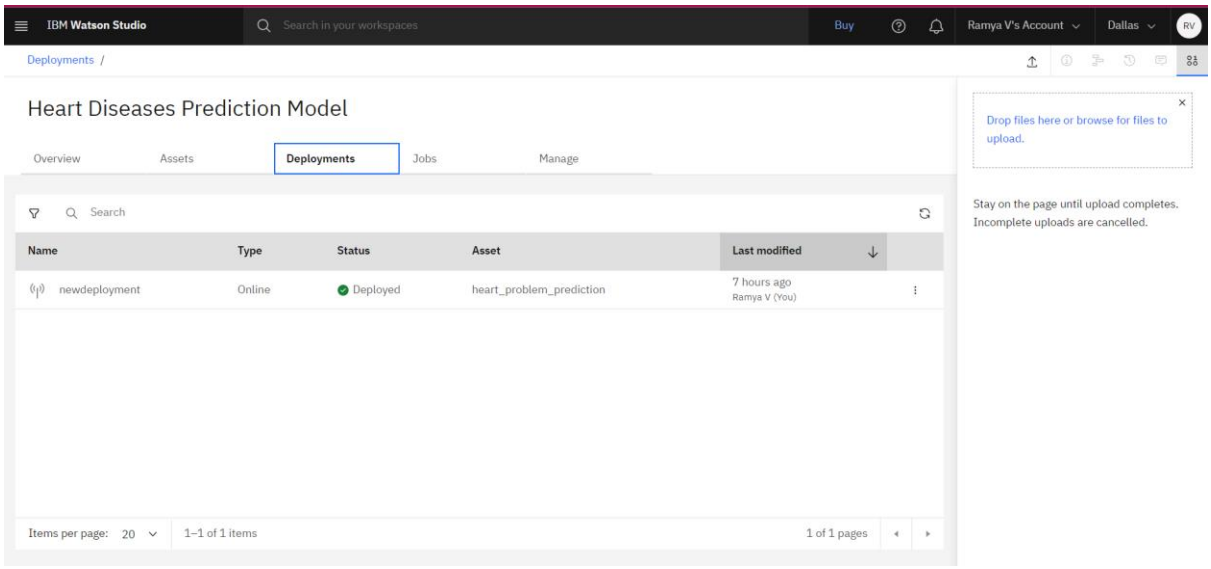
Step 10: Using Space id Deploy an scikit-learn

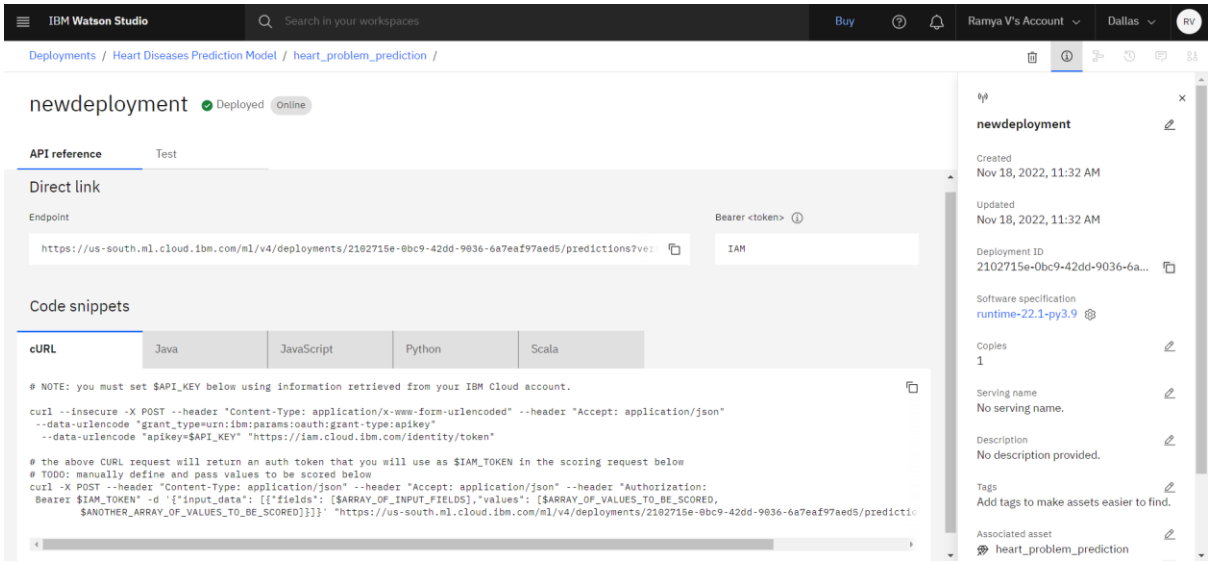


Step 11: Test And predict the Model



Step 12: Create Deployment as newdeployment





Step 13: Heart Disease prediction in python using Deployed Model in IBM cloud

Code

test_cloud_api.py - C:\Users\SSN\OneDrive - SSN Trust\Documents\SEM 7\IBM\test_cloud_api.py (3.10.3)

File Edit Format Run Options Window Help

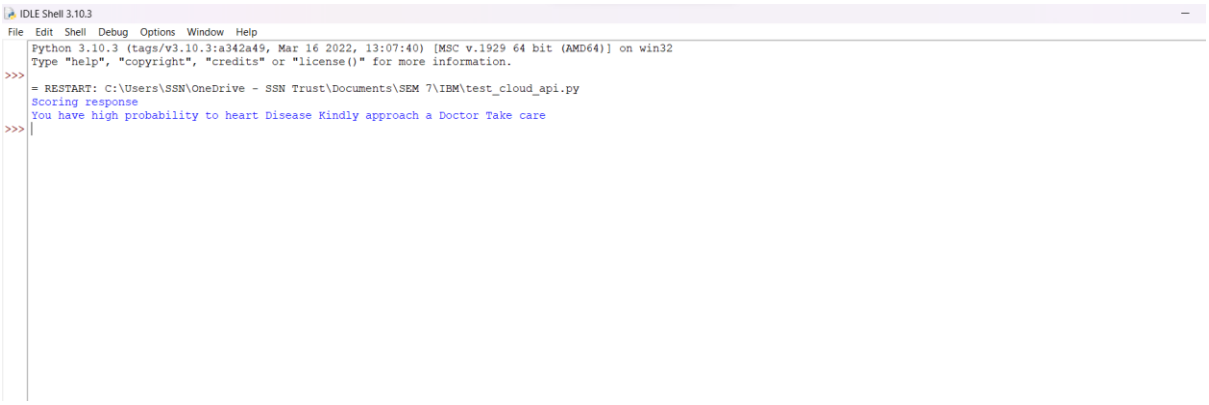
```
import requests
import json
# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "YgTKlJaXJfw0Buhk6_FCGdcAAtzUWBNBznleRI9fJ-KB"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [{"field": ['Age','Sex','Chest pain type','BP','Cholesterol','FBS over 120','EKG results','Max HR','Exercise angina','ST depression','Slope of ST'
                                             ,'Number of vessels fluro','Thallium'], "values": [[ 60, 1, 4, 117, 230, 1, 0, 160, 1, 1.4, 1, 2, 7]]}]}

response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/2102715e-0bc9-42dd-9036-6a7eaf97aed5/predictions?version=2022-11-18', json=payload_scoring,
 headers={'Authorization': 'Bearer ' + mltoken})
print("Scoring response")
predictions=response_scoring.json()
pred=predictions['predictions'][0]['values'][0][0]
if(pred =='Presence'):
    print("You have high probability to heart Disease Kindly approach a Doctor Take care")
else:
    print("Hey! Your Normal Take care")
```

Output:


IDLE Shell 3.10.3
File Edit Shell Debug Options Window Help
Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\SSN\OneDrive - SSN Trust\Documents\SEM 7\IBM\test_cloud_api.py
Scoring response
You have high probability to heart Disease Kindly approach a Doctor Take care
>>>