Project Development Phase
Delivery of Sprint-4
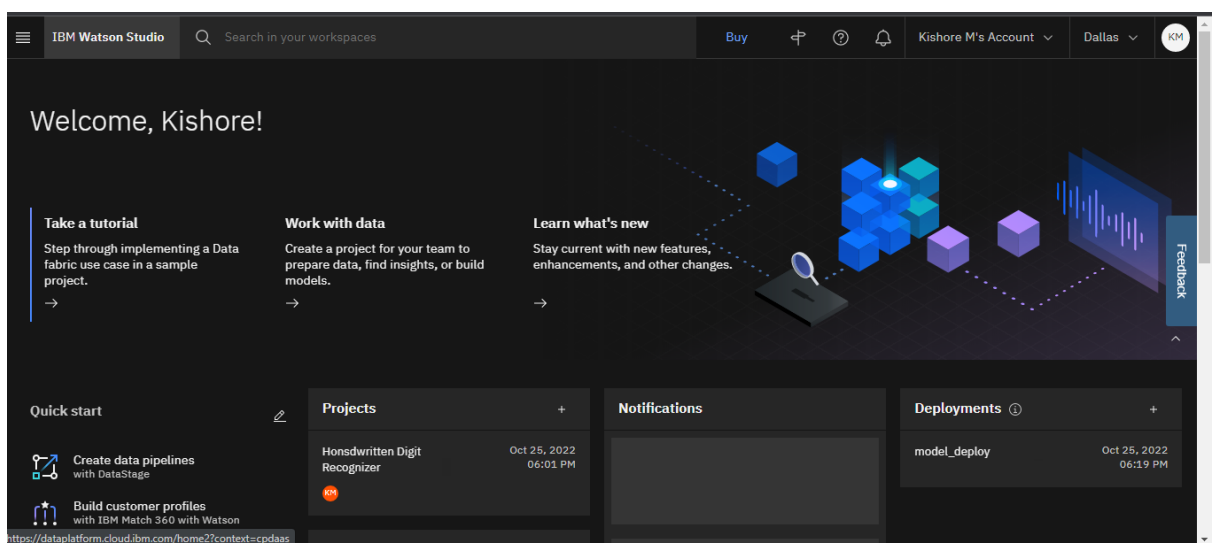
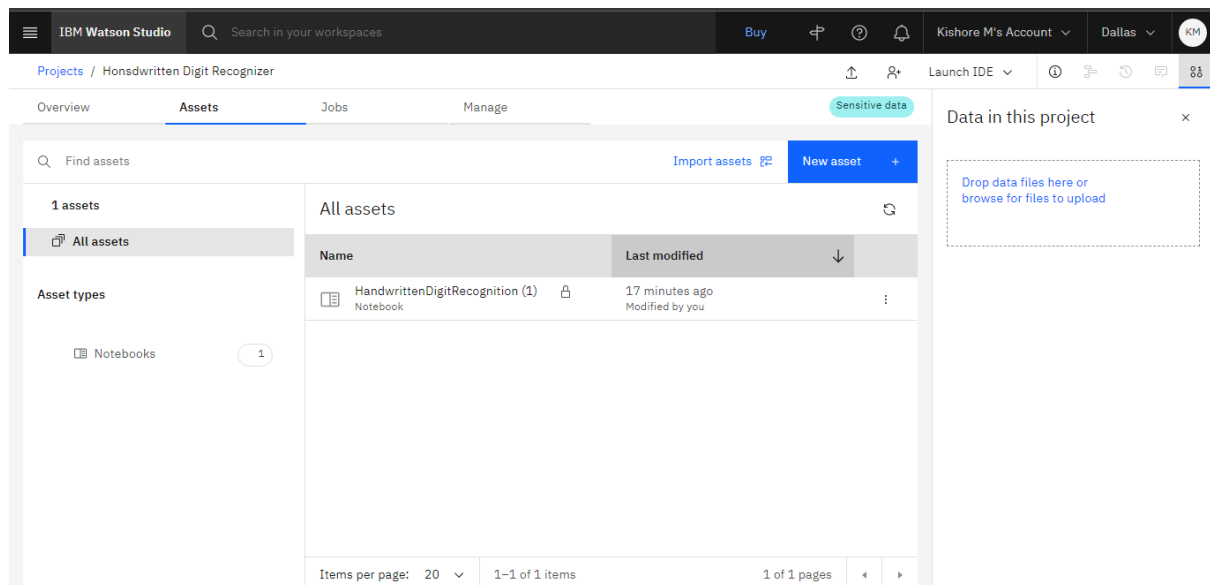| Date	15.11.2022
Team ID	PNT2022TMID53601
Project Name	Project - A Novel Method for Handwritten Digit Recognition System

Create IBM Cloud Account

1. Login into the IBM Cloud Account using your credentials.
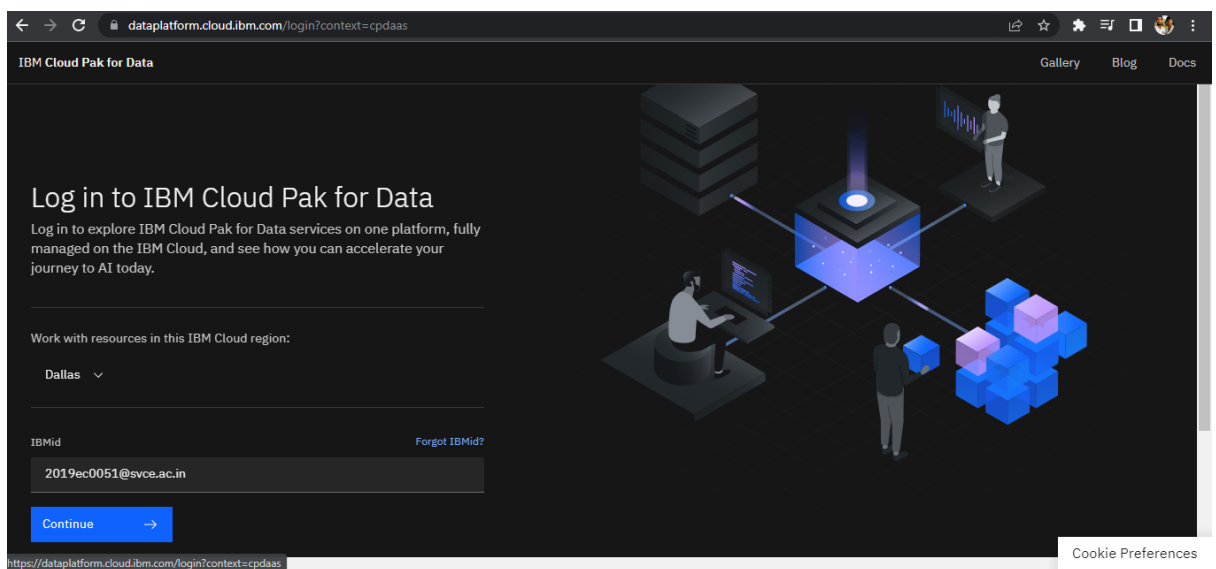
2. Go to your project. If you don't have any project create a new project
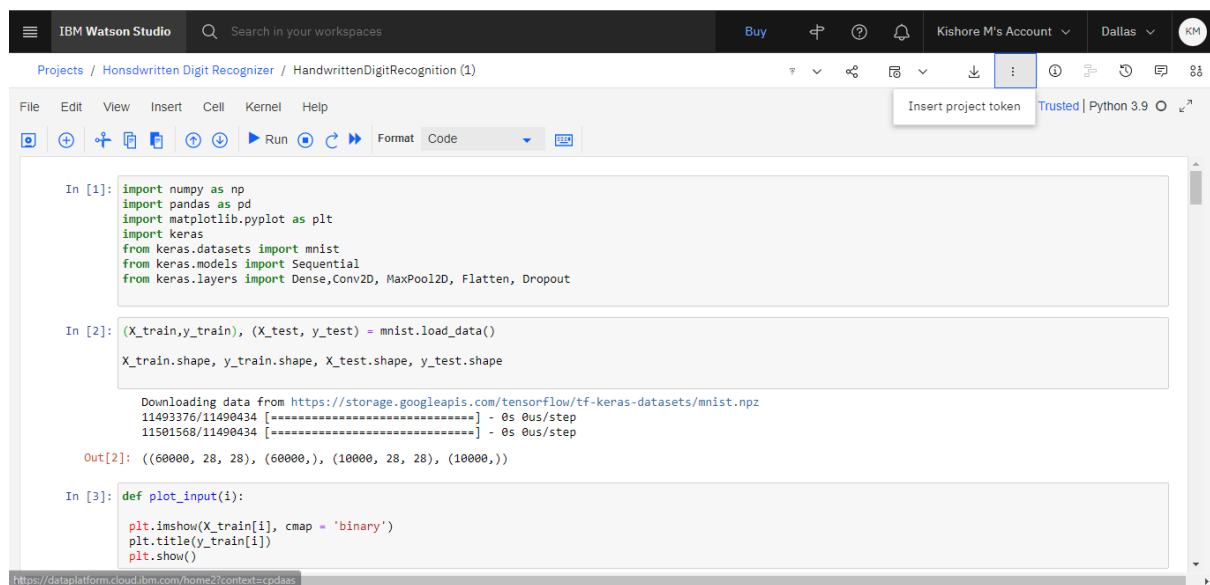
3. Inside the project upload a new **Jupyter Notebook** file as an asset and click the file.

4. Now, perform the model training by running the codes of the Jupyter notebook file

File Edit View Insert Cell Kernel Help

Format Code

```
In [4]: for i in range (2):
            plot_input(i)
```

File Edit View Insert Cell Kernel Help

Format Code

```
In [8]: model.summary()
```

```
Model: "sequential"
_____
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 32)        320

 max_pooling2d (MaxPooling2D  (None, 13, 13, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 11, 11, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 5, 5, 64)         0
 2D)

 flatten (Flatten)           (None, 1600)              0

 dropout (Dropout)           (None, 1600)              0

 dense (Dense)               (None, 10)                16010

=================================================================
Total params: 34,826
Trainable params: 34,826
Non-trainable params: 0
_____
```

File Edit View Insert Cell Kernel Help

Format Code

```
In [ ]:

In [11]: his = model.fit(X_train, y_train, epochs = 5,validation_split = 0.3)
         model.save('bestmodel.h5')
```

```
Epoch 1/5
1313/1313 [==============================] - 22s 17ms/step - loss: 0.2118 - accuracy: 0.9382 - val_loss: 0.0797 - val_accuracy: 0.9746
Epoch 2/5
1313/1313 [==============================] - 21s 16ms/step - loss: 0.0724 - accuracy: 0.9780 - val_loss: 0.0603 - val_accuracy: 0.9818
Epoch 3/5
1313/1313 [==============================] - 21s 16ms/step - loss: 0.0545 - accuracy: 0.9829 - val_loss: 0.0516 - val_accuracy: 0.9834
Epoch 4/5
1313/1313 [==============================] - 21s 16ms/step - loss: 0.0430 - accuracy: 0.9865 - val_loss: 0.0539 - val_accuracy: 0.9842
Epoch 5/5
1313/1313 [==============================] - 21s 16ms/step - loss: 0.0356 - accuracy: 0.9890 - val_loss: 0.0474 - val_accuracy: 0.9857
```

```
In [12]: model_S = keras.models.load_model('bestmodel.h5')
```

```
In [13]: score = model_S.evaluate(X_test,y_test)
         print(score[1])
```

```
313/313 [==============================] - 1s 4ms/step - loss: 0.0366 - accuracy: 0.9884
0.9883999824523926
```

```
In [ ]:
```