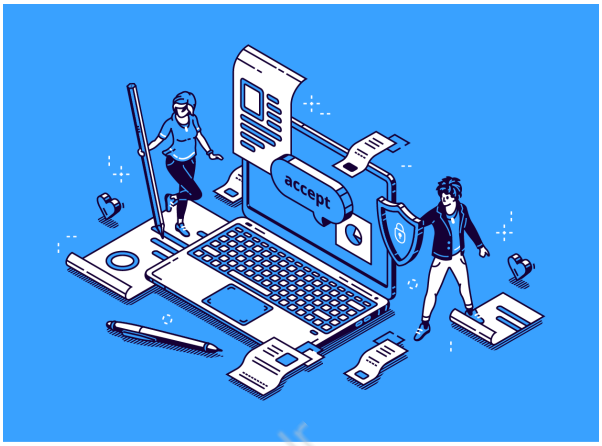
Model View Controller



What is our GOAL for this MODULE?

The goal of this module is to create a classifier model and provide the image to the classifier through an API.

What did we ACHIEVE in the class TODAY?

- We created a classifier model and sent an image to the classifier using POST Method of the API.

Which CONCEPTS/CODING BLOCKS did we cover today?

- Creating classifier
- Using postman
- Creating an API

How did we DO the activities?

1. We first created a classifier model.

```
import numpy as np
import pandas as pd
from sklearn.datasets import fetch_openml
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from PIL import Image
import PIL.ImageOps

X, y = fetch_openml('mnist_784', version=1, return_X_y=True)
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=9,
train_size=7500, test_size=2500)

X_train_scaled = X_train/255.0
X_test_scaled = X_test/255.0

clf = LogisticRegression(solver='saga',
multi_class='multinomial').fit(X_train_scaled, y_train)

def get_prediction(image):
   im_pil = Image.open(image)
   image_bw = im_pil.convert('L')
   image_bw_resized = image_bw.resize((28,28), Image.ANTIALIAS)
   pixel_filter = 20
   min_pixel = np.percentile(image_bw_resized, pixel_filter)
   image_bw_resized_inverted_scaled =
np.clip(image_bw_resized-min_pixel,0, 255)
   max_pixel = np.max(image_bw_resized)
   image_bw_resized_inverted_scaled =
np.asarray(image_bw_resized_inverted_scaled)/max_pixel
   test_sample = np.array(image_bw_resized_inverted_scaled).reshape(1,784)
   test_pred = clf.predict(test_sample)
   return test_pred[0]
```

2. We created an API to send Image in form of form data..

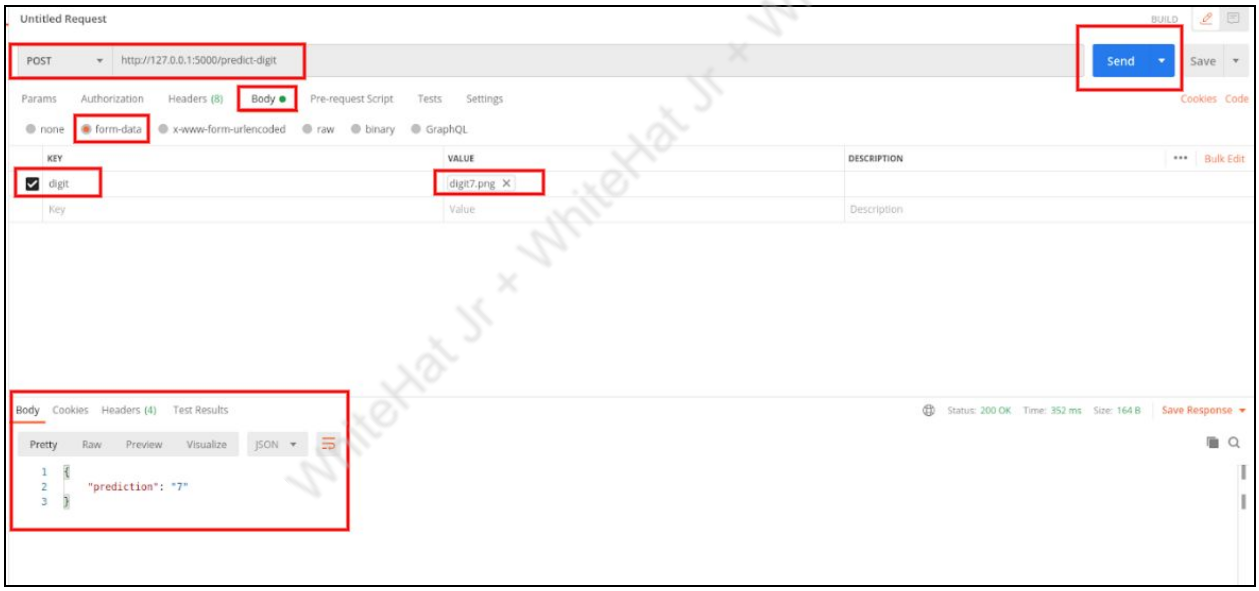
```
from flask import Flask, jsonify, request
app = Flask(__name__)
@app.route("/predict-digit", methods=["POST"])

def predict_data():
 image = request.files.get("digit")
 prediction = get_prediction(image)
 return jsonify({
   "prediction": prediction
 }), 200
```

```
if __name__ == "__main__":
 app.run(debug=True)
```

3. Then we started the server.

```
~/Desktop/API$ python3 app.py
/home/ashura/.local/lib/python3.8/site-packages/sklearn/linear_model/_sag.py:329: ConvergenceWarn
ched which means the coef_ did not converge
  warnings.warn("The max_iter was reached which means "
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

1. Opened the postman and tested the API and classification model by sending an image as a form data to it.



What's NEXT?

In the next class, we will learn to integrate Our own API with a react native app..

EXTEND YOUR KNOWLEDGE:

Learn about the usage of postman software from the following link.
https://www.guru99.com/postman-tutorial.html