

```

from flask import Flask,render_template,request

# Flask-It is our framework which we are going to use to run/serve our application.

#request-for accessing file which was uploaded by the user on our application.

import os

import numpy as np #used for numerical analysis

from tensorflow.keras.models import load_model#to load our trained model

from tensorflow.keras.preprocessing import image

import requests


app = Flask(__name__,template_folder="templates") # initializing a flask app

# Loading the model

model=load_model('nutrition.h5')

print("Loaded model from disk")


@app.route('/')# route to display the home page

def home():

    return render_template('home.html')#rendering the home page


@app.route('/image1',methods=['GET','POST'])# routes to the index html

def image1():

    return render_template("image.html")


@app.route('/predict',methods=['GET', 'POST'])# route to show the predictions in a web UI

def launch():

    if request.method=='POST':

        f=request.files['file'] #requesting the file

        basepath=os.path.dirname('__file__')#storing the file directory

```

```

filepath=os.path.join(basepath,"uploads",f.filename)#storing the file in uploads folder
f.save(filepath)#saving the file

img=image.load_img(filepath,target_size=(64,64)) #load and reshaping the image
x=image.img_to_array(img)#converting image to an array
x=np.expand_dims(x,axis=0)#changing the dimensions of the image

pred=np.argmax(model.predict(x), axis=1)
print("prediction",pred)#printing the prediction
index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']

result=str(index[pred[0]])

x=result
print(x)
result=nutrition(result)
print(result)

return render_template("0.html",showcase=(result),showcase1=(x))
def nutrition(index):

url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"

querystring = {"query":tomato}

headers = {
    'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",
    'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
}

```

```
response = requests.request("GET", url, headers=headers, params=querystring)

print(response.text)

return response.json()['items']

if __name__ == "__main__":
    # running the app
    app.run(debug=False)
```