

## Final Code

Date	15 November 2022
Team ID	PNT2022TMID13870
Project Name	Project - AI-Powered Nutrition Analyzer for Fitness Enthusiasts

### HTML Page:

#### form.html

```
<!DOCTYPE html>
```

```
<html lang="en" dir="ltr">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>Login</title>
```

```
<link rel="stylesheet" href="style.css">
```

<style>

@import

url('https://fonts.googleapis.com/css2?family=Noto+Sans:wght@700&family=Poppins:wght@400;500;600&display=swap');

{

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: "Poppins", sans-serif;

}

body{

margin: 0;

padding: 0;

background: linear-gradient(120deg, #d7a4ed, #8e44ad);

```
height: 100vh;

overflow: hidden;

}

.center{

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

width: 400px;

background: white;

border-radius: 10px;

box-shadow: 10px 10px 15px rgba(0,0,0,0.05);

}

.center h1{
```

```
text-align: center;

padding: 20px 0;

border-bottom: 1px solid silver;
}

.center form{

padding: 0 40px;

box-sizing: border-box;
}

form .txt_field{

position: relative;

border-bottom: 2px solid #adadad;

margin: 30px 0;
}

.txt_field input{
```

```
width: 100%;  
  
padding: 0 5px;  
  
height: 40px;  
  
font-size: 16px;  
  
border: none;  
  
background: none;  
  
outline: none;  
  
}  
  
.txt_field label{  
  
    position: absolute;  
  
    top: 50%;  
  
    left: 5px;  
  
    color: #adadad;  
  
    transform: translateY(-50%);
```

```
font-size: 16px;

pointer-events: none;

transition: .5s;
}

.txt_field span::before{

content: "";

position: absolute;

top: 40px;

left: 0;

width: 0%;

height: 2px;

background: #8e44ad;

transition: .5s;
}
```

```
.txt_field input:focus ~ label,
```

```
.txt_field input:valid ~ label{
```

```
    top: -5px;
```

```
    color: #8e44ad;
```

```
}
```

```
.txt_field input:focus ~ span::before,
```

```
.txt_field input:valid ~ span::before{
```

```
    width: 100%;
```

```
}
```

```
.pass{
```

```
    margin: -5px 0 20px 5px;
```

```
    color: #a6a6a6;
```

```
    cursor: pointer;
```

```
}
```

```
.pass:hover{  
    text-decoration: underline;  
}
```

```
input[type="submit"]{  
    width: 100%;  
    height: 50px;  
    border: 1px solid;  
    background: #8e44ad;  
    border-radius: 25px;  
    font-size: 18px;  
    color: #e9f4fb;  
    font-weight: 700;  
    cursor: pointer;  
    outline: none;
```



```
}
```

```
input[type="submit"]:hover{
```

```
    border-color: #8e44ad;
```

```
    transition: .5s;
```

```
}
```

```
.signup_link{
```

```
    margin: 30px 0;
```

```
    text-align: center;
```

```
    font-size: 16px;
```

```
    color: #666666;
```

```
}
```

```
.signup_link a{
```

```
    color: #8e44ad;
```

```
    text-decoration: none;
```

```
}
```

```
.signup_link a:hover{
```

```
text-decoration: underline;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="center">
```

```
<h1>Login</h1>
```

```
<form method="post">
```

```
<div class="txt_field">
```

```
<input type="text" required>
```

```
<span></span>
```

```
<label>Username</label>

</div>

<div class="txt_field">

  <input type="password" required>

  <span></span>

  <label>Password</label>

</div>

<div class="pass">Forgot Password?</div>

<input type="submit" value="Login">

<div class="signup_link">

</div>

</form>

</div>

</body>
```

</html>

file:///C:/Users/harin/Desktop/FrontEnd/form.html

VPN

📷 🔍 📄 🔌 📶

## Login

Username

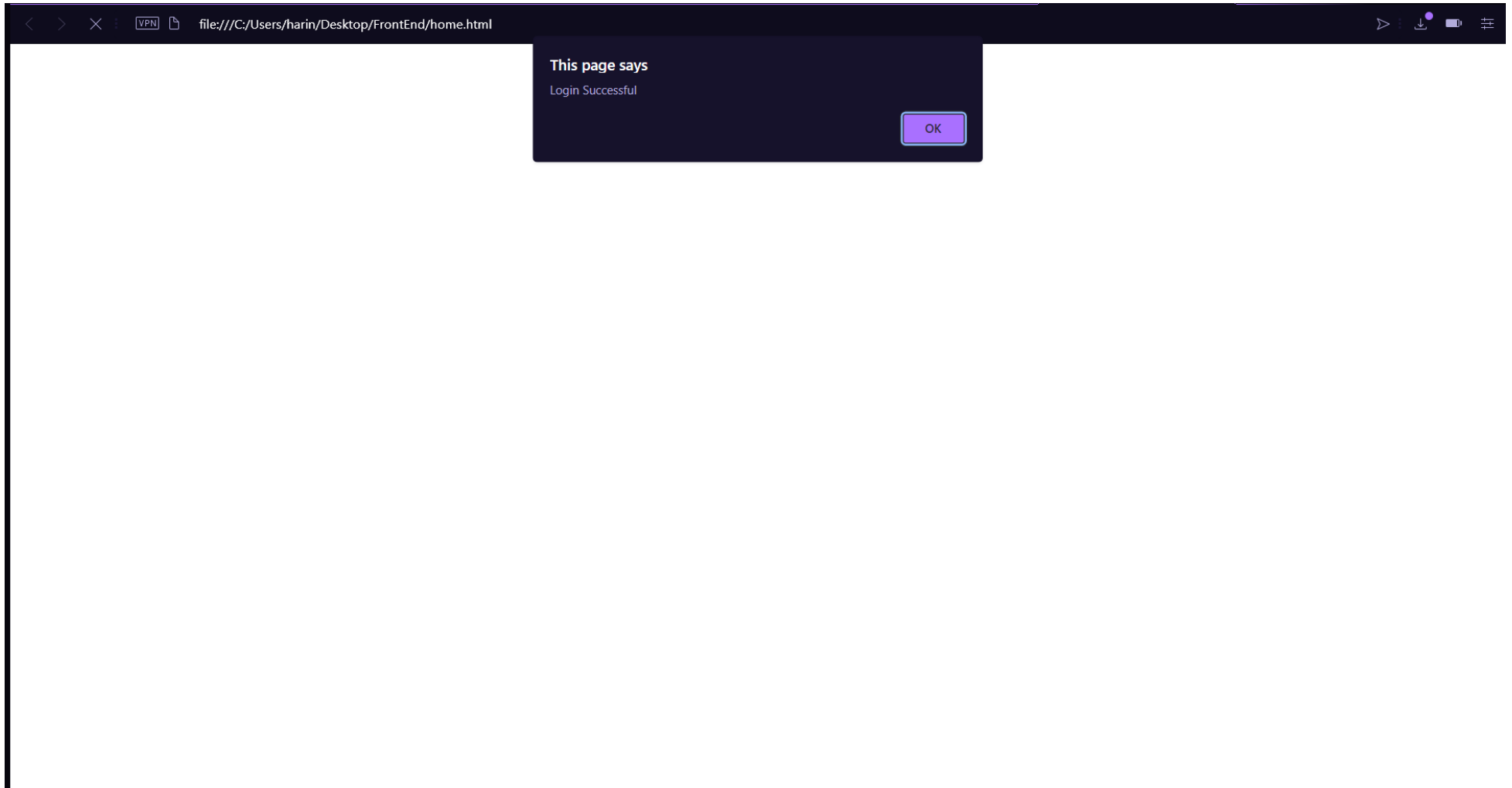
PNT2022TMID13870

Password

.....

Forgot Password?

Login



## home.html

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <style>
```

```
    body{
```

```
margin: 0;
```

```
position: absolute;
```

```
top: 50%;
```

```
left: 50%;
```

```
-ms-transform: translate(-50%, -50%);
```

```
transform: translate(-50%, -50%);  
text-align: center;  
background-image: url(home3.jpg);  
background-position: center;  
background-repeat: no-repeat;  
background-attachment: fixed;  
text-transform: capitalize;  
color: aliceblue;
```

```
}
```

```
a{
```

```
text-decoration: none;  
cursor: pointer;  
color: rgb(3, 7, 11);
```

```
border: 2px black solid;  
  
padding: 5px;  
  
border-radius: 3px;  
  
background-color: #fff;  
}
```

```
</style>>
```

```
<title>Home</title>
```

```
</head>
```

```
<body>
```

```
<h2>AI-Powered Nutrition Analyser for Fitness Enthusiasts</h2>
```

```
<h4>Team ID: PNT2022TMID13870</h3>
```

```
<br><br>
```

```
<h3>“To ensure good health: eat lightly, breathe deeply, live moderately, cultivate cheerfulness and maintain an  
interest in life.” – William Londen</h3>
```

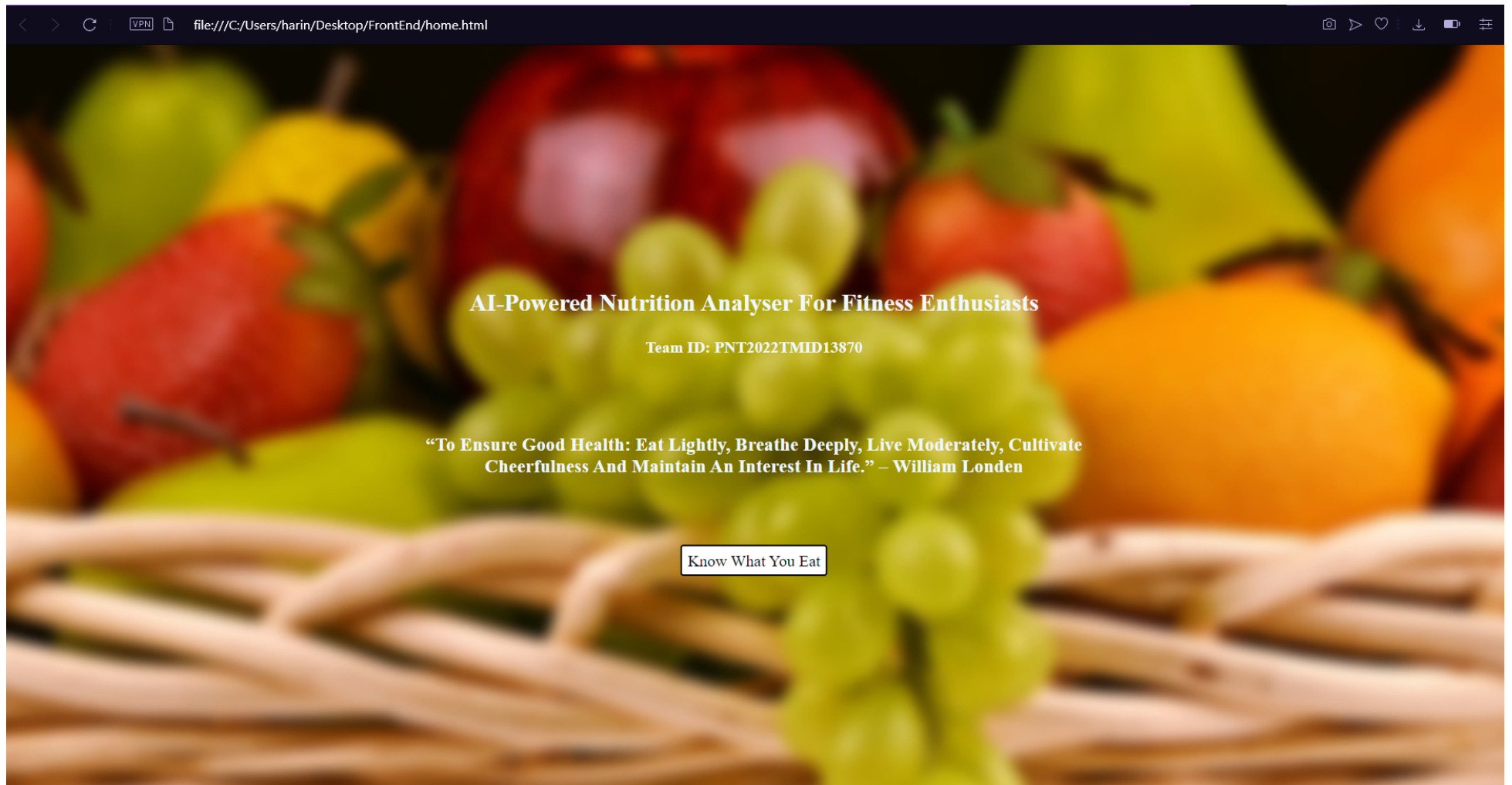


<br><br><br>

<a href="image.html">Know what you eat</a>

</body>

</html>



## image.html

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<!-- Required meta tags -->
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<!-- Bootstrap CSS --> <link rel="stylesheet"
```

```
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-
```

```
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJISAwIGgFAW/dAiS6JXm"
```

```
crossorigin="anonymous">
```

```
<title>Upload Image</title>
```

```
<style>
```

```
body {
```

```
background-color: #f2f7fb;
```

```
overflow: hidden;
```

```
}
```

```
.mt-100 {
```

```
    margin-top: 10px
```

```
}
```

```
.card {
```

```
border-radius: 5px;
```

```
-webkit-box-shadow: 0 0 5px 0 rgba(43, 43, 43, .1), 0 11px 6px -7px rgba(43, 43, 43, .1);
```

```
box-shadow: 0 0 5px 0 rgba(43, 43, 43, .1), 0 11px 6px -7px rgba(43, 43, 43, .1);
```

```
border: none;
```

```
margin-bottom: 30px;
```

```
-webkit-transition: all .3s ease-in-out;
```

```
transition: all .3s ease-in-out
```

```
}
```

```
.card .card-header {
```

```
background-color: transparent;
```

```
border-bottom: none;
```

```
padding: 20px; position:
```

```
relative
```

```
}
```

```
.card .card-header h5:after {
```

```
content: "";
```

```
background-color: #d2d2d2; width:
```

```
101px; height: 1px;
```

```
position: absolute; bottom:
```

```
6px; left: 20px
```

```
}
```

```
.card .card-block {
```

```
padding: 1.25rem
```

```
}
```

```
.dropzone.dz-clickable {
```

```
cursor: pointer
```

```
}
```

```
.dropzone { min-height: 150px;
```

```
border: 1px solid rgba(42, 42, 42, 0.05);
```

```
background: rgba(204, 204, 204, 0.15);
```

```
padding: 20px; border-radius: 5px;
```

```
-webkit-box-shadow: inset 0 0 5px 0 rgba(43, 43, 43, 0.1);
```

```
box-shadow: inset 0 0 5px 0 rgba(43, 43, 43, 0.1)
}
```

```
.m-t-20 {
    margin-top: 20px
}
```

```
.btn-primary,
.sweet-alert button.confirm,
.wizard>.actions a {
    background-color: #4099ff;
    border-color: #4099ff;
    color: #fff; cursor: pointer;
    -webkit-transition: all ease-in .3s;
    transition: all ease-in .3s
```

```
}
```

```
.btn {
```

```
border-radius: 2px;
```

```
text-transform: capitalize;
```

```
font-size: 15px; padding:
```

```
10px 19px;
```

```
cursor: pointer;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="row d-flex justify-content-center mt-100">
```

```
<div class="col-md-8">
```



```
<div class="card">
```

```
<div class="card-header">
```

```
<h3>Know Your Food Calorie</h4>
```

```
<h6>Know live food calories & nutrition information from a single food image</h6>
```

```
</div>
```

```
<div class="card-block">
```

```
<form action="/result" method = "POST" class="dropzone dz-clickable" enctype =  
"multipart/form-data">
```

```
<input type = "file" name = "file" />
```

```
<div class="text-center m-t-20">
```

```
<input class="btn btn-primary" type = "submit"/>
```

```
</div>
```

```
</form>
```

<br>

<h4>Instructions:</h4>

<dl>

<dt>Limitations</dt>

<dd>- The image size must be under 1024KB.</dd>

<dd>- The image format must be in JPEG, JPG or PNG.</dd>

<dt>Do's</dt>

<dd>- Center the food on the picture.</dd>

<dd>- Upload squared images, meaning that height and width are the same.</dd>

<dt>Dont's</dt>

<dd>- Blurry images.</dd>

<dd>- Images that include multiple food items.</dd>

</dl>

<br>

</div>

</div>

</div>

</div>

</body>

</html>

## Know Your Food Calorie

Know live food calories & nutrition information from a single food image

Choose File

No file chosen

Submit

### Instructions:

#### Limitations

- The image size must be under 1024KB.
- The image format must be in JPEG, JPG or PNG.

#### Do's

- Center the food on the picture.
- Upload squared images, meaning that height and width are the same.

#### Dont's

- Blurry images.
- Images that include multiple food items.

## Data Collection

Drive Link : [https://drive.google.com/drive/folders/1Fs-MwaF5qmHZi6-xn\\_IHNMLiuBYuWVn0](https://drive.google.com/drive/folders/1Fs-MwaF5qmHZi6-xn_IHNMLiuBYuWVn0)

Download the dataset using the above given link

# Unzipping the dataset

```
!unzip '/content/Dataset.zip'
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/33_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/34_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/35_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/36_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/37_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/38_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/39_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/40_100.jpg
```

```
inflating: Dataset/TRAIN_SET/PINEAPPLE/41_100.jpg
```

inflating: Dataset/TRAIN\_SET/PINEAPPLE/42\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/43\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/44\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/45\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/46\_100.jpg inflating:

Dataset/TRAIN\_SET/PINEAPPLE/47\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/48\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/49\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/4\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/50\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/51\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/52\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/53\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/54\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/55\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/56\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/57\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/58\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/59\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/5\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/60\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/61\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/62\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/63\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/64\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/65\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/66\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/67\_100.jpg

inflating: Dataset/TRAIN\_SET/PINEAPPLE/68\_100.jpg

creating: Dataset/TRAIN\_SET/WATERMELON/

inflating: Dataset/TRAIN\_SET/WATERMELON/0\_100.jpg

inflating: Dataset/TRAIN\_SET/WATERMELON/100\_100.jpg

## **Image ProProcessing**

### **Importing the ImageDataGenerator Library**

```
import numpy as np
```

```
import tensorflow as tf
```

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras import layers
```

```
from tensorflow.keras.layers import Dense, Flatten
```

```
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
```

```
from keras.preprocessing.image import ImageDataGenerator
```



### **Config ImageDataGenerator Class**

```
train_datagen = ImageDataGenerator(rescale = 1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale = 1./255)
```

### **Applying Image DataGenerator Functionality To Trainset And Testset**

#Applying Image DataGenerator Functionality To Trainset And Testset

```
x_train = train_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet-IBM/TRAIN_SET',
target_size=(64, 64), batch_size=5, color_mode='rgb', class_mode='sparse')
```

#Applying Image DataGenerator Functionality To Testset

```
x_test = test_datagen.flow_from_directory( r'/content/drive/MyDrive/DataSet-IBM/TEST_SET',
target_size=(64, 64), batch_size=5, color_mode='rgb', class_mode='sparse')
```

```
Found 4128 images belonging to 5 classes.  
Found 929 images belonging to 5 classes.
```

## **Image PreProcessing**

Model Building.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share Settings Profile

RAM  Disk  Editing ^

Files

{x}

drive

sample\_data

ainutrition.h5

<>

85.01 GB available

+ Code + Text

0s [2] test\_datagen = ImageDataGenerator(rescale = 1./255)

7s [3] #Applying Image DataGenerator Functionality To Trainset And Testset  
x\_train = train\_datagen.flow\_from\_directory(  
    r'/content/drive/MyDrive/DataSet-IBM/TRAIN\_SET',  
    target\_size=(64, 64),batch\_size=5,color\_mode='rgb',class\_mode='sparse')  
#Applying Image DataGenerator Functionality To Testset  
x\_test = test\_datagen.flow\_from\_directory(  
    r'/content/drive/MyDrive/DataSet-IBM/TEST\_SET',  
    target\_size=(64, 64),batch\_size=5,color\_mode='rgb',class\_mode='sparse')  
  
Found 4128 images belonging to 5 classes.  
Found 929 images belonging to 5 classes.

0s [4] #checking the number of classes  
print(x\_train.class\_indices)  
  
{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}

0s [5] #checking the number of classes  
print(x\_test.class\_indices)  
  
{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}

0s [6] from collections import Counter as c  
c(x\_train.labels)  
  
Counter({0: 995, 1: 1364, 2: 1019, 3: 275, 4: 475})

0s completed at 10:58 AM

## **Model Creation**

### **Importing libraries**

```
import numpy as np
```

```
import tensorflow as tf
```

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras import layers
```

```
from tensorflow.keras.layers import Dense, Flatten
```

```
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
```

```
from keras.preprocessing.image import ImageDataGenerator
```

### **Initializing the Model**

```
model = Sequential()
```

## **Adding CNN Layers**

```
classifier = Sequential()
```

```
# First convolution layer and pooling
```

```
classifier.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
```

```
classifier.add(MaxPooling2D(pool_size=(2, 2)))
```

```
# Second convolution layer and pooling
```

```
classifier.add(Conv2D(32, (3, 3), activation='relu'))
```

```
# input_shape is going to be the pooled feature maps from the previous convolution layer
```

```
classifier.add(MaxPooling2D(pool_size=(2, 2)))
```

```
# Flattening the layers
```

```
classifier.add(Flatten())
```

## **Adding Dense Layers**

```
classifier.add(Dense(units=128, activation='relu'))
```

```
classifier.add(Dense(units=5, activation='softmax'))
```

```
classifier.summary()
```

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
=====		
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 128)	802944
dense_1 (Dense)	(None, 5)	645

=====

Total params: 813,733

Trainable params: 813,733

Non-trainable params: 0

---

## **Configure the Learning Process**

# Compiling the CNN

# categorical\_crossentropy for more than 2

```
classifier.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

## **Train The Model**

```
classifier.fit_generator(generator=x_train, steps_per_epoch = len(x_train), epochs=20,
```

```
validation_data=x_test, validation_steps = len(x_test))
```



Model Building.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment

Share

Files

Connecting to a runtime to enable file browsing.

{x}

<>

+ Code + Text

Reconnect

Editing

"""Entry point for launching an IPython kernel.

Epoch 1/20

826/826 [=====] - 1315s 2s/step - loss: 0.6087 - accuracy: 0.7599 - val\_loss: 0.5782 - val\_accuracy: 0.7847

Epoch 2/20

826/826 [=====] - 47s 56ms/step - loss: 0.4062 - accuracy: 0.8454 - val\_loss: 0.4071 - val\_accuracy: 0.8558

Epoch 3/20

826/826 [=====] - 46s 56ms/step - loss: 0.3754 - accuracy: 0.8607 - val\_loss: 0.4110 - val\_accuracy: 0.8471

Epoch 4/20

826/826 [=====] - 45s 55ms/step - loss: 0.3381 - accuracy: 0.8728 - val\_loss: 0.4668 - val\_accuracy: 0.8170

Epoch 5/20

826/826 [=====] - 49s 59ms/step - loss: 0.3282 - accuracy: 0.8769 - val\_loss: 0.4054 - val\_accuracy: 0.8493

Epoch 6/20

826/826 [=====] - 47s 56ms/step - loss: 0.3081 - accuracy: 0.8854 - val\_loss: 0.4247 - val\_accuracy: 0.8418

Epoch 7/20

826/826 [=====] - 45s 54ms/step - loss: 0.2897 - accuracy: 0.8900 - val\_loss: 0.4057 - val\_accuracy: 0.8590

Epoch 8/20

826/826 [=====] - 46s 56ms/step - loss: 0.2746 - accuracy: 0.8932 - val\_loss: 0.4180 - val\_accuracy: 0.8741

Epoch 9/20

826/826 [=====] - 45s 54ms/step - loss: 0.2689 - accuracy: 0.8995 - val\_loss: 0.4639 - val\_accuracy: 0.8418

Epoch 10/20

826/826 [=====] - 46s 55ms/step - loss: 0.2456 - accuracy: 0.9092 - val\_loss: 0.3555 - val\_accuracy: 0.8773

Epoch 11/20

826/826 [=====] - 46s 56ms/step - loss: 0.2278 - accuracy: 0.9104 - val\_loss: 0.3919 - val\_accuracy: 0.8622

Epoch 12/20

826/826 [=====] - 46s 56ms/step - loss: 0.2104 - accuracy: 0.9213 - val\_loss: 0.3689 - val\_accuracy: 0.8751

Epoch 13/20

826/826 [=====] - 46s 56ms/step - loss: 0.2100 - accuracy: 0.9191 - val\_loss: 0.3579 - val\_accuracy: 0.8827

Epoch 14/20

826/826 [=====] - 47s 57ms/step - loss: 0.1906 - accuracy: 0.9319 - val\_loss: 0.4280 - val\_accuracy: 0.8611

Epoch 15/20

826/826 [=====] - 46s 55ms/step - loss: 0.1827 - accuracy: 0.9329 - val\_loss: 0.3347 - val\_accuracy: 0.9031

Epoch 16/20

826/826 [=====] - 43s 52ms/step - loss: 0.1636 - accuracy: 0.9394 - val\_loss: 0.4189 - val\_accuracy: 0.8579

Epoch 17/20

826/826 [=====] - 47s 57ms/step - loss: 0.1609 - accuracy: 0.9397 - val\_loss: 0.3509 - val\_accuracy: 0.8967

Epoch 18/20

826/826 [=====] - 46s 56ms/step - loss: 0.1363 - accuracy: 0.9479 - val\_loss: 0.3901 - val\_accuracy: 0.8924

Epoch 19/20

826/826 [=====] - 46s 55ms/step - loss: 0.1339 - accuracy: 0.9537 - val\_loss: 0.4557 - val\_accuracy: 0.8730

Epoch 20/20

826/826 [=====] - 45s 55ms/step - loss: 0.1179 - accuracy: 0.9566 - val\_loss: 0.3902 - val\_accuracy: 0.9042

<keras.callbacks.History at 0x7fc096200250>

0s completed at 10:58 AM

## **Save the Model**

```
classifier.save('ainutrition.h5')
```

## **Test the Model**

```
#Predict the results
```

```
from tensorflow.keras.models import load_model
```

```
from keras.preprocessing import image
```

```
from keras_preprocessing.image import load_img
```

```
model = load_model("ainutrition.h5")
```

```
from tensorflow.keras.utils import img_to_array
```

```
#loading of the image
```

```
img = load_img(r'/content/drive/MyDrive/DataSet-IBM/TEST_SET/ORANGE/n07749192_1251.jpg', grayscale=False,  
target_size= (64,64))
```

```
#image to array
```

```
x = img_to_array(img)
```

```
#changing the shape
```

```
x = np.expand_dims(x,axis = 0)
```

```
predict_x=model.predict(x)
```

```
classes_x=np.argmax(predict_x,axis=-1)
```

```
classes_x
```

```
1/1 [=====] - 0s 107ms/step  
array([2])
```

```
index=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']
```

```
result=str(index[classes_x[0]])
```

```
result
```



```
'ORANGE'
```

```
print(result)
```

```
if result == 'APPLES':
```

```
    print("One serving, or one medium apple, provides about 95 calories, 0 gram fat, 1 gram protein, 25 grams  
carbohydrate, 19 grams sugar (naturally occurring), and 3 grams fiber.")
```

```
elif result == 'BANANA':
```

```
print("One serving, or one medium ripe banana, provides about 110 calories, 0 gram fat, 1 gram protein, 28 grams carbohydrate, 15 grams sugar (naturally occurring), 3 grams fiber, and 450 mg potassium.")
```

```
elif result == 'ORANGE':
```

```
print("60 calories, No fat or sodium, 3 grams of fiber, 12 grams of sugar, 1 gram of protein, 14 micrograms of vitamin A, 70 milligrams of vitamin C, 6% of your daily recommended amount of calcium.")
```

```
elif result == 'PINEAPPLE':
```

```
print("Calories: 83, Fat: 1.7 grams, Protein: 1 gram, Carbs: 21.6 grams, Fiber: 2.3 grams, Vitamin C: 88% of the Daily Value (DV), Manganese: 109% of the DV, Vitamin B6: 11% of the DV.")
```

```
elif result == 'WATERMELON':
```

```
print("Calories: 46, Carbs: 11.5 grams, Fiber: 0.6 grams, Sugar: 9.4 grams, Protein: 0.9 grams, Fat: 0.2 grams, Vitamin A: 5% of the Daily Value (DV), Vitamin C: 14% of the DV.")
```

ORANGE

60 calories, No fat or sodium, 3 grams of fiber, 12 grams of sugar, 1 gram of protein, 14 micrograms of vitamin A, 70 milligrams of vitamin C, 6% of your daily recommended amount

<

>

## Model Building



+ Code + Text

✓ RAM  
Disk

Editing



✓ [13] classifier.save('ainutrition.h5')

{x}

✓ [14] #Predict the results  
from tensorflow.keras.models import load\_model  
from keras.preprocessing import image  
from keras\_preprocessing.image import load\_img  
model = load\_model("ainutrition.h5")✓ [15] from tensorflow.keras.utils import img\_to\_array  
#loading of the image  
img = load\_img(r'/content/drive/MyDrive/DataSet-IBM/TEST\_SET/ORANGE/n07749192\_1251.jpg', grayscale=False, target\_size= (64,64))  
#image to array  
x = img\_to\_array(img)  
#changing the shape  
x = np.expand\_dims(x,axis = 0)  
predict\_x=model.predict(x)  
classes\_x=np.argmax(predict\_x,axis=-1)  
classes\_x

1s

1/1 [=====] - 0s 107ms/step  
array([2])

&lt;&gt;

✓ [16] index=['APPLES', 'BANANA', 'ORANGE', 'PINEAPPLE', 'WATERMELON']  
result=str(index[classes\_x[0]])  
result

0s

'ORANGE'



## Webpage

### Know Your Food Calorie

Know live food calories & nutrition information from a single food image

Choose File n07749192\_1251.jpg

Submit

#### Instructions:

##### Limitations

- The image size must be under 1024KB.
- The image format must be in JPEG, JPG or PNG.

##### Do's

- Center the food on the picture.
- Upload squared images, meaning that height and width are the same.

##### Dont's

- Blurry images.
- Images that include multiple food items.





### **Fruit: ORANGE**

Nutrition: 60 calories, No fat or sodium, 3 grams of fiber, 12 grams of sugar, 1 gram of protein, 14 micrograms of vitamin A, 70 milligrams of vitamin C, 6% of your daily recommended amount of calcium.