DATE	07-11-2022
TEAM ID	IBM-Project-PNT2022TMID22678
PROJECT NAME	AI-POWERED NUTRITION ANALYSER FOR FITNESS ENTHUSIASTICS

MODEL BUILDING

```
import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A layer consists of a tensor-in tensor-out computation function
#Dense layer is the regular deeply connected neural network layer
from tensorflow.keras.layers import Dense,Flatten
#Falten-used fot flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dropout #Convolutional layer
#MaxPooling2D-for downsampling the image
from keras.preprocessing.image import ImageDataGenerator
```

```
# Initializing the CNN
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 a fully connected layer
classifier.add(Dense(units=128, activation='relu')) classifier.add(Dense(units=5, activation='softmax')) # softmax for more than 2



classifier.summary()#summary of our model
Model: "sequential"
_________________________________________________________________


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             (None, 14, 14, 32)         0
 (MaxPooling2D)

 flatten (Flatten)           (None, 6272)           0

 dense (Dense)               (None, 128)                80294
                                                        4
 dense_1 (Dense)             (None, 5)                  645

=================================================================
```

```
Total params: 813,733
Trainable params: 813,733
Non-trainable params: 0
```