```
from keras.preprocessing.image import ImageDataGenerator

train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_r
ange=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)

x_train=train_datagen.flow_from_directory(
    r'C:\Users\pavan\Desktop\AI Image_Processing\Data_Set',

target_size=(64,64),batch_size=5,color_mode='rgb',class_mode='sparse')

x_test=test_datagen.flow_from_directory(
    r'C:\Users\pavan\Desktop\AI Image_Processing\Data_Set',

target_size=(64,64),batch_size=5,color_mode='rgb',class_mode='sparse')

print(x_train.class_indices)

print(x_test.class_indices)

from collections import Counter as c
c(x_train.labels)

import numpy as np
import tensorflow
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

model=Sequential()
```

Creating the model

```
classifier=Sequential()
classifier.add(Conv2D(32,
(3,3),input_shape=(64,64,3),activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))
classifier.add(Cov2D(32,(3,3),activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))
classifier.add(Flatten())

classifier.summary()
```

Compiling the model

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

Fitting 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))
```

saving model

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

Predicting results

```
from tesorflow.keras.models import load_model
from keras.preprocessing import image
model =load_model("nutrition.h5")

img=image.load_img(r"C:\Users\pavan\Desktop\AI Image_Processing\
Data_Set\",
                   grayscale=False,target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred=model.predict_classes(x)
pred

index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']
result=str(index[pred[0]])
result
```