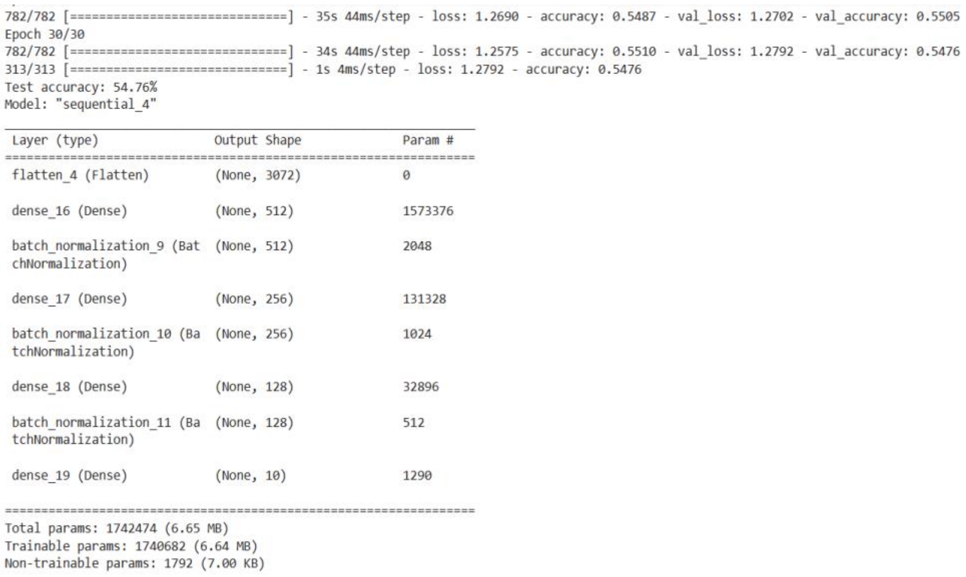
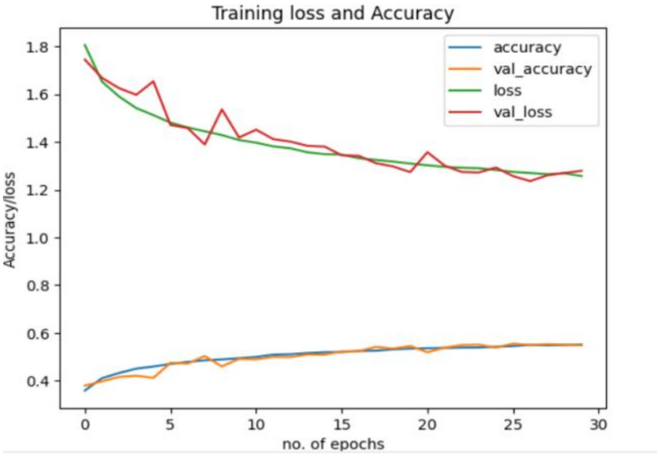
PROGRAM

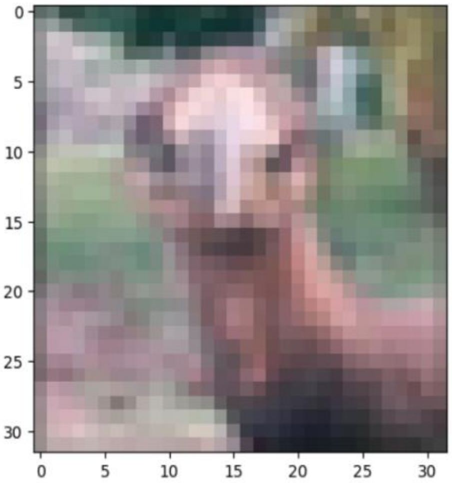
```
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.preprocessing.image import
ImageDataGenerator
(train_images, train_labels), (test_images, test_labels) =
cifar10.load_data()
train_images, test_images = train_images / 255.0, test_images
/ 255.0
train_labels = to_categorical(train_labels, 10)
test_labels = to_categorical(test_labels, 10)
datagen = ImageDataGenerator(
    rotation_range=15,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True,
    )
datagen.fit(train_images)

model = models.Sequential()
model.add(layers.Flatten(input_shape=(32, 32, 3)))
model.add(layers.Dense(512, activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Dense(256, activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Dense(128, activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Dense(10, activation='softmax'))

model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

history = model.fit(datagen.flow(train_images, train_labels,
batch_size=64),epochs=30,
               validation_data=(test_images,test_labels))

test_loss, test_acc = model.evaluate(test_images, test_labels)
print(f'Test accuracy: {test_acc * 100:.2f}%')
model.summary()
history.history.keys()
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Training loss and Accuracy')
plt.xlabel('no. of epochs')
plt.ylabel('Accuracy/loss')
plt.legend(['accuracy','val_accuracy','loss','val_loss'])
plt.show()

import random
import numpy as np
n= random.randint(0,9999)
plt.imshow(test_images[n])
plt.show()
predictions = model.predict(test_images)
predicted_label = np.argmax(predictions)
print(f"Predicted Label: {predicted_label}")
import matplotlib.pyplot as plt
plt.plot(history.history['accuracy'], label='Training Accuracy')
plt.plot(history.history['val_accuracy'], label='Validation Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend()
plt.show()
```

OUTPUT

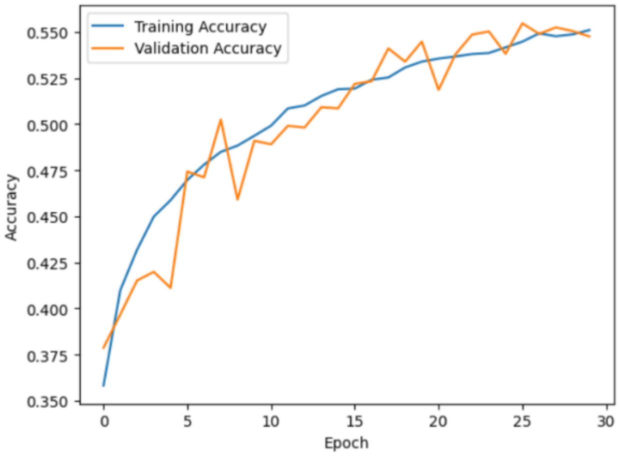


```
782/782 [==============================] - 35s 44ms/step - loss: 1.2690 - accuracy: 0.5487 - val_loss: 1.2702 - val_accuracy: 0.5505
Epoch 30/30
782/782 [==============================] - 34s 44ms/step - loss: 1.2575 - accuracy: 0.5510 - val_loss: 1.2792 - val_accuracy: 0.5476
313/313 [==============================] - 1s 4ms/step - loss: 1.2792 - accuracy: 0.5476
Test accuracy: 54.76%
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 flatten_4 (Flatten)         (None, 3072)              0         
                                                                 
 dense_16 (Dense)            (None, 512)               1573376   
                                                                 
 batch_normalization_9 (Bat  (None, 512)               2048      
 chNormalization)                                                
                                                                 
 dense_17 (Dense)            (None, 256)               131328    
                                                                 
 batch_normalization_10 (Ba  (None, 256)               1024      
 tchNormalization)                                               
                                                                 
 dense_18 (Dense)            (None, 128)               32896     
                                                                 
 batch_normalization_11 (Ba  (None, 128)               512       
 tchNormalization)                                               
                                                                 
 dense_19 (Dense)            (None, 10)                1290      
                                                                 
=================================================================
Total params: 1742474 (6.65 MB)
Trainable params: 1740682 (6.64 MB)
Non-trainable params: 1792 (7.00 KB)
```





```
313/313 [==============================] - 1s 2ms/step
```


Training Accuracy
Validation Accuracy
Accuracy
Epoch