```
!mkdir -p ~/.kaggle
```

```
!cp kaggle.json ~/.kaggle/
```

```
!kaggle datasets download -d salader/dogs-vs-cats
```

```
Warning: Your Kaggle API key is readable by other users on this system! To fix this, you can run 'chmod 600 /root/.kaggle/kaggle.js
Downloading dogs-vs-cats.zip to /content
 98% 1.04G/1.06G [00:06<00:00, 258MB/s]
100% 1.06G/1.06G [00:07<00:00, 161MB/s]
```

```
import zipfile
!unzip /content/dogs-vs-cats.zip
```

```
  inflating: train/dogs/dog.9932.jpg
  inflating: train/dogs/dog.9933.jpg
  inflating: train/dogs/dog.9934.jpg
  inflating: train/dogs/dog.9935.jpg
  inflating: train/dogs/dog.9938.jpg
  inflating: train/dogs/dog.9940.jpg
  inflating: train/dogs/dog.9942.jpg
  inflating: train/dogs/dog.9943.jpg
  inflating: train/dogs/dog.9944.jpg
  inflating: train/dogs/dog.9945.jpg
  inflating: train/dogs/dog.9947.jpg
  inflating: train/dogs/dog.9948.jpg
  inflating: train/dogs/dog.9949.jpg
  inflating: train/dogs/dog.995.jpg
  inflating: train/dogs/dog.9950.jpg
  inflating: train/dogs/dog.9951.jpg
  inflating: train/dogs/dog.9952.jpg
  inflating: train/dogs/dog.9953.jpg
  inflating: train/dogs/dog.9954.jpg
  inflating: train/dogs/dog.9957.jpg
  inflating: train/dogs/dog.9958.jpg
  inflating: train/dogs/dog.9959.jpg
  inflating: train/dogs/dog.996.jpg
  inflating: train/dogs/dog.9960.jpg
  inflating: train/dogs/dog.9961.jpg
  inflating: train/dogs/dog.9962.jpg
  inflating: train/dogs/dog.9965.jpg
  inflating: train/dogs/dog.9966.jpg
  inflating: train/dogs/dog.9967.jpg
  inflating: train/dogs/dog.9968.jpg
  inflating: train/dogs/dog.9969.jpg
  inflating: train/dogs/dog.997.jpg
  inflating: train/dogs/dog.9970.jpg
  inflating: train/dogs/dog.9971.jpg
  inflating: train/dogs/dog.9972.jpg
  inflating: train/dogs/dog.9973.jpg
  inflating: train/dogs/dog.9975.jpg
  inflating: train/dogs/dog.9976.jpg
  inflating: train/dogs/dog.9977.jpg
  inflating: train/dogs/dog.9978.jpg
  inflating: train/dogs/dog.9979.jpg
  inflating: train/dogs/dog.9980.jpg
  inflating: train/dogs/dog.9981.jpg
  inflating: train/dogs/dog.9982.jpg
  inflating: train/dogs/dog.9983.jpg
  inflating: train/dogs/dog.9984.jpg
  inflating: train/dogs/dog.9985.jpg
  inflating: train/dogs/dog.9987.jpg
  inflating: train/dogs/dog.9988.jpg
  inflating: train/dogs/dog.999.jpg
  inflating: train/dogs/dog.9990.jpg
  inflating: train/dogs/dog.9992.jpg
  inflating: train/dogs/dog.9993.jpg
  inflating: train/dogs/dog.9994.jpg
  inflating: train/dogs/dog.9996.jpg
  inflating: train/dogs/dog.9997.jpg
  inflating: train/dogs/dog.9998.jpg
  inflating: train/dogs/dog.9999.jpg
```

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

```
train_dir="/content/dogs_vs_cats/train"
test_dir="/content/dogs_vs_cats/test"
```

```
train_gen=ImageDataGenerator(rescale=1/255.)
test_gen=ImageDataGenerator(rescale=1/255.)
```

```
train=train_gen.flow_from_directory(train_dir,target_size=(224,224),class_mode="binary",batch_size=32)
test=test_gen.flow_from_directory(test_dir,target_size=(224,224),class_mode="binary",batch_size=32)
```

```
Found 20000 images belonging to 2 classes.
Found 5000 images belonging to 2 classes.
```

```
import tensorflow as tf
```

```
base=tf.keras.applications.VGG16(include_top=False)
base.trainable=False
inputs=tf.keras.layers.Input((224,224,3))
x=base(inputs,training=False)
x=tf.keras.layers.GlobalAvgPool2D()(x)
outputs=tf.keras.layers.Dense(1,activation="sigmoid")(x)
model1=tf.keras.Model(inputs,outputs)
```

```
model1.compile(loss="binary_crossentropy",optimizer="adam",metrics=["accuracy"])
```

```
model1.fit(train,epochs=5,validation_data=test)
```

```
Epoch 1/5
625/625 [==============================] - 110s 176ms/step - loss: 0.2618 - accuracy: 0.9037 - val_loss: 0.2482 - val_accuracy: 0.9
Epoch 2/5
625/625 [==============================] - 128s 204ms/step - loss: 0.2483 - accuracy: 0.9061 - val_loss: 0.2371 - val_accuracy: 0.9
Epoch 3/5
625/625 [==============================] - 128s 205ms/step - loss: 0.2377 - accuracy: 0.9097 - val_loss: 0.2278 - val_accuracy: 0.9
Epoch 4/5
625/625 [==============================] - 108s 172ms/step - loss: 0.2295 - accuracy: 0.9118 - val_loss: 0.2202 - val_accuracy: 0.9
Epoch 5/5
625/625 [==============================] - 128s 204ms/step - loss: 0.2229 - accuracy: 0.9149 - val_loss: 0.2144 - val_accuracy: 0.9
<keras.src.callbacks.History at 0x7d4a927efc40>
```

```
def load(path):
  img=tf.io.read_file(path)
  image=tf.image.decode_image(img)
  image=tf.image.resize(image,size=(224,224))
  image=image/255
  return image
```

```
doggie=load("/content/doggo.jpeg")
```

```
pred=tf.round(model1.predict(tf.expand_dims(doggie,axis=0)))
pred
```

```
1/1 [==============================] - 0s 233ms/step
<tf.Tensor: shape=(1, 1), dtype=float32, numpy=array([[1.]], dtype=float32)>
```

```
classes=["cat","dog"]
```

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
dog=mpimg.imread("/content/doggo.jpeg")
plt.imshow(dog)
plt.title(classes[int(pred[0])])
```

```
Text(0.5, 1.0, 'dog')
```