```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
!unzip "/content/drive/MyDrive/Mask Detection/Face Mask Dataset.zip"
```

```
Streaming output truncated to the last 5000 lines.
  inflating: Face Mask Dataset/Train/WithoutMask/1858.png
  inflating: Face Mask Dataset/Train/WithoutMask/1859.png
  inflating: Face Mask Dataset/Train/WithoutMask/1860.png
  inflating: Face Mask Dataset/Train/WithoutMask/1861.png
  inflating: Face Mask Dataset/Train/WithoutMask/1862.png
  inflating: Face Mask Dataset/Train/WithoutMask/1863.png
  inflating: Face Mask Dataset/Train/WithoutMask/1864.png
  inflating: Face Mask Dataset/Train/WithoutMask/1865.png
  inflating: Face Mask Dataset/Train/WithoutMask/1866.png
  inflating: Face Mask Dataset/Train/WithoutMask/1867.png
  inflating: Face Mask Dataset/Train/WithoutMask/1868.png
  inflating: Face Mask Dataset/Train/WithoutMask/1869.png
  inflating: Face Mask Dataset/Train/WithoutMask/1870.png
  inflating: Face Mask Dataset/Train/WithoutMask/1871.png
  inflating: Face Mask Dataset/Train/WithoutMask/1872.png
  inflating: Face Mask Dataset/Train/WithoutMask/1873.png
  inflating: Face Mask Dataset/Train/WithoutMask/1874.png
  inflating: Face Mask Dataset/Train/WithoutMask/1875.png
  inflating: Face Mask Dataset/Train/WithoutMask/1876.png
  inflating: Face Mask Dataset/Train/WithoutMask/1877.png
  inflating: Face Mask Dataset/Train/WithoutMask/1878.png
  inflating: Face Mask Dataset/Train/WithoutMask/1879.png
  inflating: Face Mask Dataset/Train/WithoutMask/188.png
  inflating: Face Mask Dataset/Train/WithoutMask/1880.png
  inflating: Face Mask Dataset/Train/WithoutMask/1881.png
  inflating: Face Mask Dataset/Train/WithoutMask/1882.png
  inflating: Face Mask Dataset/Train/WithoutMask/1883.png
  inflating: Face Mask Dataset/Train/WithoutMask/1884.png
  inflating: Face Mask Dataset/Train/WithoutMask/1885.png
  inflating: Face Mask Dataset/Train/WithoutMask/1886.png
  inflating: Face Mask Dataset/Train/WithoutMask/1887.png
  inflating: Face Mask Dataset/Train/WithoutMask/1888.png
  inflating: Face Mask Dataset/Train/WithoutMask/1889.png
  inflating: Face Mask Dataset/Train/WithoutMask/189.png
  inflating: Face Mask Dataset/Train/WithoutMask/1890.png
  inflating: Face Mask Dataset/Train/WithoutMask/1891.png
  inflating: Face Mask Dataset/Train/WithoutMask/1892.png
  inflating: Face Mask Dataset/Train/WithoutMask/1893.png
  inflating: Face Mask Dataset/Train/WithoutMask/1894.png
  inflating: Face Mask Dataset/Train/WithoutMask/1895.png
  inflating: Face Mask Dataset/Train/WithoutMask/1896.png
  inflating: Face Mask Dataset/Train/WithoutMask/1897.png
  inflating: Face Mask Dataset/Train/WithoutMask/1898.png
  inflating: Face Mask Dataset/Train/WithoutMask/1899.png
  inflating: Face Mask Dataset/Train/WithoutMask/19.png
  inflating: Face Mask Dataset/Train/WithoutMask/1901.png
  inflating: Face Mask Dataset/Train/WithoutMask/1902.png
  inflating: Face Mask Dataset/Train/WithoutMask/1903.png
  inflating: Face Mask Dataset/Train/WithoutMask/1904.png
  inflating: Face Mask Dataset/Train/WithoutMask/1905.png
  inflating: Face Mask Dataset/Train/WithoutMask/1906.png
  inflating: Face Mask Dataset/Train/WithoutMask/1907.png
  inflating: Face Mask Dataset/Train/WithoutMask/1908.png
  inflating: Face Mask Dataset/Train/WithoutMask/1910.png
  inflating: Face Mask Dataset/Train/WithoutMask/1911.png
  inflating: Face Mask Dataset/Train/WithoutMask/1912.png
  inflating: Face Mask Dataset/Train/WithoutMask/1913.png
```

```
import numpy as np
import pandas as pd
import warnings
warnings.filterwarnings('ignore')

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Flatten,Conv2D,MaxPooling2D


import os

train_dir = '/content/Face Mask Dataset/Train'
test_dir = '/content/Face Mask Dataset/Test'
```

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


train_datagen = ImageDataGenerator(rescale=1./255,
                                   rotation_range=40,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True,
                                   fill_mode='nearest')
test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(train_dir,
                                                    batch_size=666,
                                                    target_size=(150, 150),
                                                    class_mode = 'binary')

test_generator = test_datagen.flow_from_directory(test_dir,
                                                  batch_size=44,
                                                  target_size=(150, 150),
                                                  class_mode = 'binary')
```

```
Found 10000 images belonging to 2 classes.
Found 992 images belonging to 2 classes.
```

```
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dense,Flatten


from sklearn.metrics import classification_report


m = Sequential()


m.add(Conv2D(32,(4,4), activation = 'relu', input_shape = (150,150,3)))
m.add(MaxPooling2D(2,2))


m.add(Conv2D(64, (3, 3), activation='relu'))
m.add(MaxPooling2D((2, 2)))
m.add(Conv2D(128, (3, 3), activation='relu'))
m.add(MaxPooling2D((2, 2)))
m.add(Conv2D(128, (3, 3), activation='relu'))
m.add(MaxPooling2D((2, 2)))


m.add(Flatten())


m.add(Dense(512, activation='relu'))
m.add(Dense(1, activation='sigmoid'))



m.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
m.fit_generator(train_generator,epochs=3)
```

```
Epoch 1/3
16/16 [==============================] - 88s 5s/step - loss: 0.6643 - accuracy: 0.6198
Epoch 2/3
16/16 [==============================] - 76s 5s/step - loss: 0.3539 - accuracy: 0.8784
Epoch 3/3
16/16 [==============================] - 74s 5s/step - loss: 0.2948 - accuracy: 0.8870
<keras.callbacks.History at 0x7f12d422b850>
```

```
test_loss, test_accuracy = m.evaluate(test_generator)
print(test_loss)
print(test_accuracy)
```

```
23/23 [==============================] - 2s 83ms/step - loss: 0.2479 - accuracy: 0.9052
0.24793528020381927
0.9052419066429138
```

```
import tensorflow as tf
tf.keras.Model.save(m,filepath='/content/drive/MyDrive/model')
```

```
WARNING:absl:Found untraced functions such as _jit_compiled_convolution_op, _jit_compiled_convolution_op, _jit_compiled_convolution_op,
```

```
m.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 147, 147, 32)      1568

 max_pooling2d (MaxPooling2D  (None, 73, 73, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 71, 71, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 35, 35, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 33, 33, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 16, 16, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 14, 14, 128)       147584

 max_pooling2d_3 (MaxPooling  (None, 7, 7, 128)        0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 1)                 513

=================================================================
Total params: 3,453,793
Trainable params: 3,453,793
Non-trainable params: 0
_________________________________________________________________
```

```
v = tf.keras.models.load_model(filepath='/content/drive/MyDrive/model')
```

```
v.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 147, 147, 32)      1568

 max_pooling2d (MaxPooling2D  (None, 73, 73, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 71, 71, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 35, 35, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 33, 33, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 16, 16, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 14, 14, 128)       147584

 max_pooling2d_3 (MaxPooling  (None, 7, 7, 128)        0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 1)                 513

=================================================================
Total params: 3,453,793
Trainable params: 3,453,793
Non-trainable params: 0
_________________________________________________________________
```