```python
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import ImageDataGenerator

IMG_SIZE = 244
BATCH_SIZE = 32

train_datagen =
ImageDataGenerator(rescale=1./255,validation_split=0.2)
train_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Flowers/Ornamental Plants/train',
    target_size=(IMG_SIZE,IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='training'
)

val_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Flowers/Ornamental Plants/train',
    target_size=(IMG_SIZE,IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='validation'
)
```

```
Found 104 images belonging to 2 classes.
Found 26 images belonging to 2 classes.
```

```python
# Define the model
model = keras.Sequential([
    layers.Conv2D(32,
(3,3),activation='relu',input_shape=(IMG_SIZE,IMG_SIZE,3)),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(128,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Flatten(),
    layers.Dense(128,activation='relu'),
    layers.Dense(1,activation='sigmoid') #output layer
])

#compile the model
model.compile(optimizer='adam',loss='binary_crossentropy',metrics=['ac
curacy'])

model.fit(train_generator,validation_data=val_generator,epochs=1)#give
epocs 1 to check error
```

```
4/4 [==============================] - 39s 12s/step - loss: 1.3648 -
accuracy: 0.5000 - val_loss: 0.7517 - val_accuracy: 0.5000

<keras.src.callbacks.History at 0x7926c46363e0>
```

```python
model.save("ornamental plants.h5","label.txt")
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/engine/
training.py:3103: UserWarning: You are saving your model as an HDF5
file via `model.save()`. This file format is considered legacy. We
recommend using instead the native Keras format, e.g.
`model.save('my_model.keras')`.
  saving_api.save_model(
```

```python
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np

#load the saved model
model = load_model('/content/ornamental plants.h5')

#load and preprocess the test image
test_image_path ='/content/drive/MyDrive/Flowers/Ornamental
Plants/test/Damask Rose/Damask Rose (1).jpeg'
img = image.load_img(test_image_path,target_size=(244,244))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array,axis=0)

#add batch dimension
img_array /= 255. #normalize the pixel values
#make predictions
prediction = model.predict(img_array)
#print the prediction
print(prediction)
```

```
1/1 [==============================] - 0s 80ms/step
[[0.35046434]]
```

```python
if prediction < 0.5:
    print('It is a Damask Rose')
else:
    print('It is a Rain Lily')
```

```
It is a Damask Rose
```