In [48]:

```
pip install --upgrade pip
```

```
Requirement already satisfied: pip in c:\users\complab02\anaconda3\lib\site-
packages (22.2.2)
Note: you may need to restart the kernel to use updated packages.
```

In [46]:

```
pip install tensorflow
```

```
Requirement already satisfied: tensorflow in c:\users\complab02\anaconda3\li
b\site-packages (2.9.1)
Requirement already satisfied: six>=1.12.0 in c:\users\complab02\anaconda3\l
ib\site-packages (from tensorflow) (1.15.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\complab02\anaco
nda3\lib\site-packages (from tensorflow) (1.6.3)
Requirement already satisfied: tensorboard<2.10,>=2.9 in c:\users\complab02
\anaconda3\lib\site-packages (from tensorflow) (2.9.1)
Requirement already satisfied: libclang>=13.0.0 in c:\users\complab02\anacon
da3\lib\site-packages (from tensorflow) (14.0.6)
Requirement already satisfied: keras<2.10.0,>=2.9.0rc0 in c:\users\complab02
\anaconda3\lib\site-packages (from tensorflow) (2.9.0)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\complab02\anaconda3
\lib\site-packages (from tensorflow) (1.12.1)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\complab02\anacon
da3\lib\site-packages (from tensorflow) (1.1.0)
Requirement already satisfied: packaging in c:\users\complab02\anaconda3\lib
\site-packages (from tensorflow) (20.9)
Requirement already satisfied: protobuf<3.20,>=3.9.2 in c:\users\complab02\a
naconda3\lib\site-packages (from tensorflow) (3.19.4)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in c:\users\complab02\ana
conda3\lib\site-packages (from tensorflow) (0.4.0)
Requirement already satisfied: setuptools in c:\users\complab02\anaconda3\li
b\site-packages (from tensorflow) (52.0.0.post20210125)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\complab0
2\anaconda3\lib\site-packages (from tensorflow) (3.7.4.3)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\complab02\anaconda
3\lib\site-packages (from tensorflow) (1.2.0)
Requirement already satisfied: flatbuffers<2,>=1.12 in c:\users\complab02\an
aconda3\lib\site-packages (from tensorflow) (1.12)
Requirement already satisfied: keras-preprocessing>=1.1.1 in c:\users\compla
b02\anaconda3\lib\site-packages (from tensorflow) (1.1.2)
Requirement already satisfied: numpy>=1.20 in c:\users\complab02\anaconda3\l
ib\site-packages (from tensorflow) (1.20.1)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\complab02\ana
conda3\lib\site-packages (from tensorflow) (0.2.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\complab02\ana
conda3\lib\site-packages (from tensorflow) (1.47.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\complab02\anaco
nda3\lib\site-packages (from tensorflow) (3.3.0)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in c:\us
ers\complab02\anaconda3\lib\site-packages (from tensorflow) (0.26.0)
Requirement already satisfied: h5py>=2.9.0 in c:\users\complab02\anaconda3\l
ib\site-packages (from tensorflow) (2.10.0)
Requirement already satisfied: tensorflow-estimator<2.10.0,>=2.9.0rc0 in
c:\users\complab02\anaconda3\lib\site-packages (from tensorflow) (2.9.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\complab02\anac
onda3\lib\site-packages (from astunparse>=1.6.0->tensorflow) (0.36.2)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in c:\users\com
plab02\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow)
(1.8.1)
Requirement already satisfied: markdown>=2.6.8 in c:\users\complab02\anacond
a3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (3.4.1)
Requirement already satisfied: google-auth<3,>=1.6.3 in c:\users\complab02\a
naconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (2.11.
0)
Requirement already satisfied: werkzeug>=1.0.1 in c:\users\complab02\anacond
```

```
a3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (1.0.1)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in c:\u
sers\complab02\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->ten
sorflow) (0.6.1)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in c:\users
\complab02\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorf
low) (0.4.6)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\complab02\ana
conda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (2.25.1)
Requirement already satisfied: pyparsing>=2.0.2 in c:\users\complab02\anacon
da3\lib\site-packages (from packaging->tensorflow) (2.4.7)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in c:\users\complab02
\anaconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>
=2.9->tensorflow) (5.2.0)
Requirement already satisfied: pyasn1-modules>=0.2.1 in c:\users\complab02\a
naconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>=
2.9->tensorflow) (0.2.8)
Requirement already satisfied: rsa<5,>=3.1.4 in c:\users\complab02\anaconda3
\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>=2.9->tens
orflow) (4.9)
Requirement already satisfied: requests-oauthlib>=0.7.0 in c:\users\complab0
2\anaconda3\lib\site-packages (from google-auth-oauthlib<0.5,>=0.4.1->tensor
board<2.10,>=2.9->tensorflow) (1.3.1)
Requirement already satisfied: importlib-metadata>=4.4 in c:\users\complab02
\anaconda3\lib\site-packages (from markdown>=2.6.8->tensorboard<2.10,>=2.9->
tensorflow) (4.12.0)
Requirement already satisfied: idna<3,>=2.5 in c:\users\complab02\anaconda3
\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->tensor
flow) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\complab02\anac
onda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->t
ensorflow) (2020.12.5)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\complab02\a
naconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9
->tensorflow) (1.26.4)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\users\complab02\anaco
nda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->te
nsorflow) (4.0.0)
Requirement already satisfied: zipp>=0.5 in c:\users\complab02\anaconda3\lib
\site-packages (from importlib-metadata>=4.4->markdown>=2.6.8->tensorboard<
2.10,>=2.9->tensorflow) (3.4.1)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in c:\users\complab02\an
aconda3\lib\site-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3
->tensorboard<2.10,>=2.9->tensorflow) (0.4.8)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\complab02\anacond
a3\lib\site-packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<0.
5,>=0.4.1->tensorboard<2.10,>=2.9->tensorflow) (3.2.0)
Note: you may need to restart the kernel to use updated packages.
```

In [47]:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import tensorflow as tf
from tensorflow.keras import models,layers
```

In [49]:

```
import tensorflow as tf
from tensorflow.keras import models,layers
```

In [78]:

```
IMAGE_SIZE=256
BATCH_SIZE=32
CHANNELS=3
EPOCHS=50
```

In [79]:

```
dataset= tf.keras.preprocessing.image_dataset_from_directory(
    "Orangedataset",
    shuffle=True,
    image_size=(IMAGE_SIZE,IMAGE_SIZE),
    batch_size=BATCH_SIZE

)
```

```
Found 1000 files belonging to 2 classes.
```

In [ ]:

```

```

In [80]:

```
class_names = dataset.class_names
class_names
```

Out[80]:

```
['fresh_orange', 'stale_orange']
```

In [81]:

```
len(dataset)
```

Out[81]:

```
32
```

In [82]:

```
for image_batch, label_batch in dataset.take(1):
    print(image_batch.shape)
    print(label_batch.numpy())
```

```
(32, 256, 256, 3)
[0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 1 1 1 1 1 0 0 1]
```

In [83]:

```
plt.figure(figsize=(10,10))
for image_batch, label_batch in dataset.take(1):
    for i in range(12):
        ax=plt.subplot(3,4,i+1)
        plt.imshow(image_batch[i].numpy().astype("uint8"))
        plt.title(class_names[label_batch[i]])
        plt.axis("off")
```

In [84]:

```
len(dataset)
```

Out[84]:

32

In [58]:

```
#80% ==> training
#20% ==> 10% validation, 10% test
```

In [59]:

```
train_size=0.8
len(dataset)*train_size
```

Out[59]:

25.6

In [60]:

```
train_ds=dataset.take(25)
len(train_ds)
```

Out[60]:

25

In [61]:

```
test_ds = dataset.skip(25)
len(test_ds)
```

Out[61]:

7

In [62]:

```
val_size=0.1
len(dataset)*val_size
```

Out[62]:

3.2

In [63]:

```
val_ds=test_ds.take(3)
len(val_ds)
```

Out[63]:

3

In [64]:

```
test_ds=test_ds.skip(3)
len(test_ds)
```

Out[64]:

4

In [65]:

```
def get_dataset_partitions_tf(ds,train_split=0.8,val_split=0.1,test_split=0.1,shuffle=True,

    ds_size=len(ds)
    if shuffle:
        ds=ds.shuffle(shuffle_size, seed=12)

    train_size=int(train_split*ds_size)
    val_size=int(val_split*ds_size)

    train_ds=ds.take(train_size)

    val_ds=ds.skip(train_size).take(val_size)
    test_ds=ds.skip(train_size).skip(val_size)

    return train_ds, val_ds, test_ds
```

In [66]:

```
train_ds, val_ds, test_ds= get_dataset_partitions_tf(dataset)
```

In [67]:

```
len(train_ds)
```

Out[67]:

```
25
```

In [68]:

```
len(val_ds)
```

Out[68]:

```
3
```

In [69]:

```
len(test_ds)
```

Out[69]:

```
4
```

In [85]:

```
train_ds=train_ds.cache().shuffle(500).prefetch(buffer_size=tf.data.AUTOTUNE)
val_ds=val_ds.cache().shuffle(500).prefetch(buffer_size=tf.data.AUTOTUNE)
test_ds=test_ds.cache().shuffle(500).prefetch(buffer_size=tf.data.AUTOTUNE)
```

In [86]:

```
resize_and_rescale= tf.keras.Sequential([
    layers.experimental.preprocessing.Resizing(IMAGE_SIZE,IMAGE_SIZE),
    layers.experimental.preprocessing.Rescaling(1.0/255)
])
```

In [87]:

```
data_augmentation=tf.keras.Sequential([
    layers.experimental.preprocessing.RandomFlip("horizontal_and_vertical"),
    layers.experimental.preprocessing.RandomRotation(0.2)
])
```

In [88]:

```
input_shape=(BATCH_SIZE, IMAGE_SIZE,IMAGE_SIZE,CHANNELS)
n_classes =2

model= models.Sequential([
    resize_and_rescale,
    data_augmentation,
    layers.Conv2D(16,(3,3),activation='relu',input_shape= input_shape),
    layers.MaxPooling2D((2,2)),

    layers.Conv2D(32,kernel_size=(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(32,kernel_size=(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),

    layers.Conv2D(32,(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(32,(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(32,(3,3),activation='relu'),
    layers.MaxPooling2D((2,2)),

    layers.Flatten(),
    layers.Dense(32,activation='relu'),
    layers.Dense(n_classes,activation='softmax'),


])

model.build(input_shape = input_shape)
```

In [89]:

```
model.summary()
```

```
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential_8 (Sequential)   (32, 256, 256, 3)         0

 sequential_9 (Sequential)   (32, 256, 256, 3)         0

 conv2d_28 (Conv2D)          (32, 254, 254, 16)        448

 max_pooling2d_28 (MaxPoolin  (32, 127, 127, 16)       0
 g2D)

 conv2d_29 (Conv2D)          (32, 125, 125, 32)        4640

 max_pooling2d_29 (MaxPoolin  (32, 62, 62, 32)         0
 g2D)

 conv2d_30 (Conv2D)          (32, 60, 60, 32)          9248

 max_pooling2d_30 (MaxPoolin  (32, 30, 30, 32)         0
 g2D)

 conv2d_31 (Conv2D)          (32, 28, 28, 32)          9248

 max_pooling2d_31 (MaxPoolin  (32, 14, 14, 32)         0
 g2D)

 conv2d_32 (Conv2D)          (32, 12, 12, 32)          9248

 max_pooling2d_32 (MaxPoolin  (32, 6, 6, 32)           0
 g2D)

 conv2d_33 (Conv2D)          (32, 4, 4, 32)            9248

 max_pooling2d_33 (MaxPoolin  (32, 2, 2, 32)           0
 g2D)

 flatten_5 (Flatten)         (32, 128)                 0

 dense_9 (Dense)             (32, 32)                  4128

 dense_10 (Dense)            (32, 2)                   66

=================================================================
Total params: 46,274
Trainable params: 46,274
Non-trainable params: 0
_________________________________________________________________
```

In [99]:

```
model.compile(
    optimizer='adam',
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False),
    metrics=['accuracy']
)
```

In [91]:

```
history=model.fit(
    train_ds,
    epochs=EPOCHS,
    batch_size=BATCH_SIZE,
    verbose=1,
    validation_data=val_ds
)
```

```
Epoch 1/50
25/25 [==============================] - 39s 1s/step - loss: 0.4722 - accu
racy: 0.7862 - val_loss: 0.3798 - val_accuracy: 0.8854
Epoch 2/50
25/25 [==============================] - 26s 1s/step - loss: 0.4318 - accu
racy: 0.8075 - val_loss: 0.3755 - val_accuracy: 0.8125
Epoch 3/50
25/25 [==============================] - 27s 1s/step - loss: 0.3491 - accu
racy: 0.8575 - val_loss: 0.3222 - val_accuracy: 0.8958
Epoch 4/50
25/25 [==============================] - 23s 894ms/step - loss: 0.3273 - a
ccuracy: 0.8775 - val_loss: 0.2971 - val_accuracy: 0.8750
Epoch 5/50
25/25 [==============================] - 20s 799ms/step - loss: 0.3089 - a
ccuracy: 0.8800 - val_loss: 0.2806 - val_accuracy: 0.8750
Epoch 6/50
25/25 [==============================] - 16s 624ms/step - loss: 0.2779 - a
ccuracy: 0.9013 - val_loss: 0.2842 - val_accuracy: 0.8646
Epoch 7/50
```

In [92]:

```
scores=model.evaluate(test_ds)
```

```
4/4 [==============================] - 1s 112ms/step - loss: 0.1145 - accura
cy: 0.9609
```

In [93]:

```
scores
```

Out[93]:

```
[0.11451883614063263, 0.9609375]
```

In [94]:

```
history
```

Out[94]:

```
<keras.callbacks.History at 0x2589f162a60>
```

In [95]:

```
history.params
```

Out[95]:

```
{'verbose': 1, 'epochs': 50, 'steps': 25}
```

In [96]:

```
history.history.keys()
```

Out[96]:

```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

In [97]:

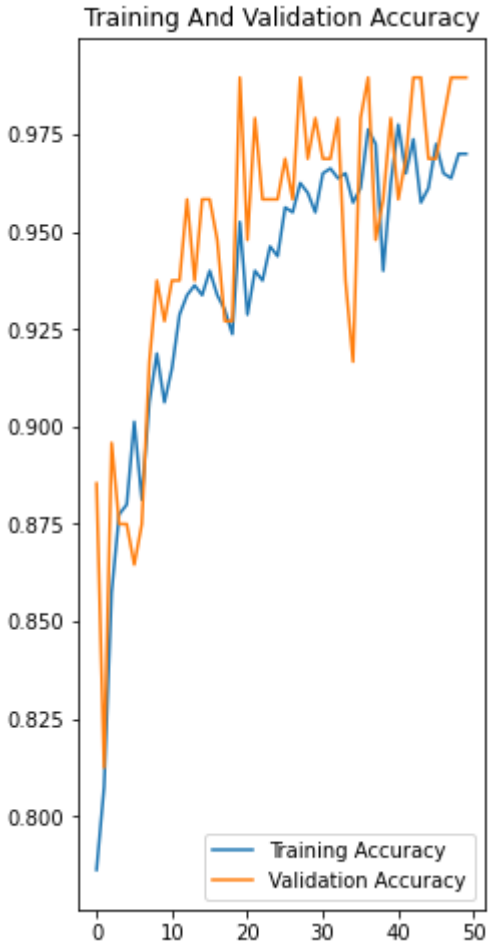
```
acc=history.history['accuracy']
val_acc=history.history['val_accuracy']

los = history.history['loss']
val_loss = history.history['val_loss']
```

In [102]:

```
plt.figure(figsize=(8,8))
plt.subplot(1,2,1)
plt.plot(range(EPOCHS),acc, label='Training Accuracy')
plt.plot(range(EPOCHS), val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training And Validation Accuracy')
```

Out[102]:

```
Text(0.5, 1.0, 'Training And Validation Accuracy')
```

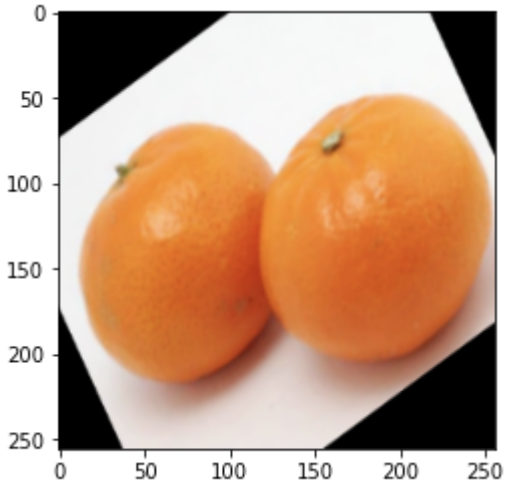


In [ ]:

In [107]:

```
for images_batch, labels_batch in test_ds.take(1):
    first_image = images_batch[0].numpy().astype("uint8")
    first_label = labels_batch[0].numpy()

    print("first Image to predict")
    plt.imshow(first_image)
    print("actual label:",class_names[first_label])

    batch_prediction = model.predict(images_batch)
    print("Predicted:",class_names[np.argmax(batch_prediction[0])])
```

```
first Image to predict
actual label: fresh_orange
1/1 [==============================] - 1s 564ms/step
Predicted: fresh_orange
```



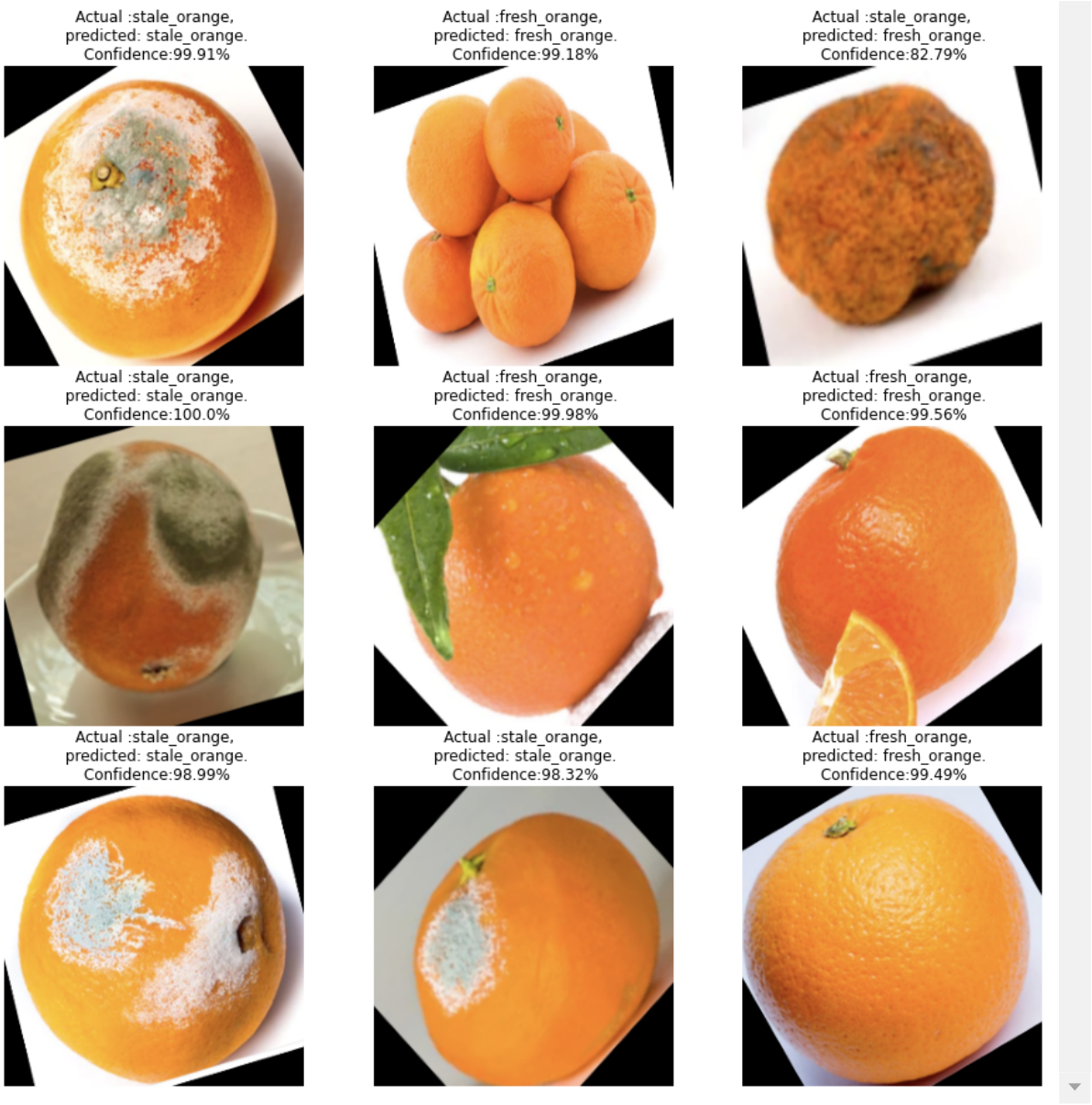
In [108]:

```
def predict(model,img):
    img_array = tf.keras.preprocessing.image.img_to_array(images[i].numpy())
    img_array= tf.expand_dims(img_array,0) # Create Batch

    predictions=model.predict(img_array)

    predicted_class =class_names[np.argmax(predictions[0])]
    confidence=round(100* (np.max(predictions[0])),2)
    return predicted_class, confidence
```

In [113]:

```
plt.figure(figsize=(15,15))
for images, labels in test_ds.take(1):
    for i in range(9):

        ax= plt.subplot(3,3,i+1)
        plt.imshow(images[i].numpy().astype("uint8"))

        predicted_class, confidence = predict(model, images[i].numpy())
        actual_class = class_names[labels[i]]

        plt.title(f"Actual :{actual_class},\n predicted: {predicted_class}.\n Confidence:{c

        plt.axis("off")
```

```
1/1 [==============================] - 0s 121ms/step
1/1 [==============================] - 0s 120ms/step
1/1 [==============================] - 0s 120ms/step
1/1 [==============================] - 0s 120ms/step
1/1 [==============================] - 0s 121ms/step
1/1 [==============================] - 0s 122ms/step
1/1 [==============================] - 0s 125ms/step
1/1 [==============================] - 0s 121ms/step
1/1 [==============================] - 0s 121ms/step
```



In [ ]:

```
model_version=1
model.save()
```