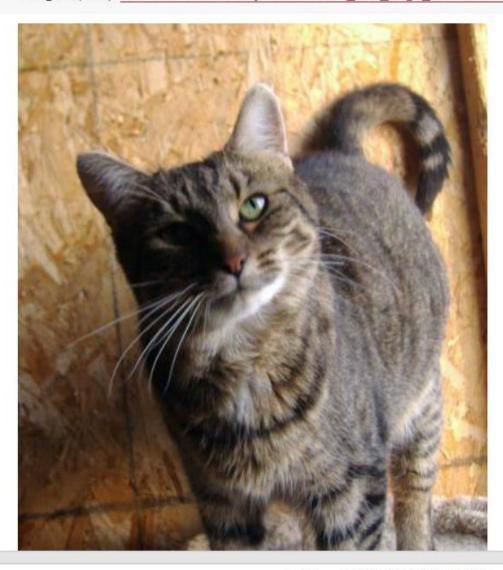


```
from skimage.io import imread
from PIL import Image
categories=["cats", "dogs"]
for category in categories:
  for file in os.listdir (os.path.join(df1, category)):
    img path=os.path.join(df1, category,file)
    print(img path)
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.123.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.160.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.17.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.182.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.164.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.200.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.122.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.112.jpg
/content/drive/MvDrive/Cats and Dogs dataset/dogs/dog.127.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.11.ipg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.134.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.148.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.114.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.135.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.174.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.116.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.13.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.117.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.145.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.130.jpg
/content/drive/MyDrive/Cats and Dogs dataset/dogs/dog.104.jpg
```

T V CO E W N

[18] #Taking any random path of the previous output to display the image
Image.open("/content/drive/MyDrive/Cats_and_Dogs_dataset/cats/cat.21.jpg")



Found 1600 images belonging to 2 classes.

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score

dt = DecisionTreeClassifier() 
dt.fit(x_train, y_train)

y_pred = dt.predict(x_test)
accuracy = accuracy_score(y_test, y_pred)
print("The accuracy of the decision tree classifier is:",accuracy)
```

The accuracy of the decision tree classifier is: 0.4

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

rfc = RandomForestClassifier(n_estimators=1000) 
rfc.fit(x_train, y_train)

y_pred = rfc.predict(x_test)
accuracy = accuracy_score(y_test, y_pred)
print("The accuracy of the random forest classifier is :",accuracy)
```

The accuracy of the random forest classifier is: 0.4

```
√ [12]
        from sklearn.ensemble import RandomForestClassifier
        from sklearn.model selection import GridSearchCV
        from sklearn.metrics import accuracy score
        rfc = RandomForestClassifier(random state=42)
        param grid = {
            'n estimators': [200, 500],
            'max_features': ['auto', 'sqrt', 'log2'],
            'max depth': [4, 5, 6, 7, 8],
            'criterion': ['gini', 'entropy']
        CV rfc = GridSearchCV(estimator=rfc, param grid=param grid, cv=5)
        CV rfc.fit(x train, y train)
        print(f"The best parameters for the random forest classifier are {CV rfc.best params }.")
        rfc1 = RandomForestClassifier(random state=42, max features='auto', n estimators=200, max depth=8, criterion='gini')
        rfc1.fit(x train, y train)
        pred = rfc1.predict(x test)
        accuracy = accuracy score(y test, pred)
        print("The accuracy of the random forest classifier is :",accuracy)
        /usr/local/lib/python3.10/dist-packages/sklearn/ensemble/ forest.py:424: FutureWarning: `max features='auto'` has been deprecated in 1.1 and
```

/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/ forest.py:424: FutureWarning: `max features='auto'` has been deprecated in 1.1 and

warn/

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_forest.py:424: FutureWarning: `max_features='auto'` has been deprecated in warn(
The best parameters for the random forest classifier are {'criterion': 'gini', 'max_depth': 4, 'max_features': 'auto', 'n_estimators' /usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_forest.py:424: FutureWarning: `max_features='auto'` has been deprecated in warn(
The accuracy of the random forest classifier is : 0.3 

#Note - We have trained 1000 images of cats and dogs. Increase in sample image will lead to increase in accuracy import pandas as pd

df = pd.DataFrame({"Actual_Value": y_test, "Predicted_Value": y_pred}) 

print(df)
```

	Actual_Value	Predicted_Value
0	1	1
1	1	0
2	1	1
3	0	1
4	1	0
5	0	0
6	0	1
7	1	0
8	0	0
9	0	1