In [1]:
```
import pandas as pd
covidData = pd.read_csv("Datasets/covid.csv")
```

In [2]:
```
covidData.head()
```

Out[2]:

	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovered	ActiveCases	Serious,Critical	Tot Cases/1M pop	Deaths/1M pop	TotalTests	Tests/1M pop	WHO Region
0	USA	North America	3.311981e+08	5032179	NaN	162804.0	NaN	2576668.0	NaN	2292707.0	18296.0	15194.0	492.0	63139605.0	190640.0	Americas
1	Brazil	South America	2.127107e+08	2917562	NaN	98644.0	NaN	2047660.0	NaN	771258.0	8318.0	13716.0	464.0	13206188.0	62085.0	Americas
2	India	Asia	1.381345e+09	2025409	NaN	41638.0	NaN	1377384.0	NaN	606387.0	8944.0	1466.0	30.0	22149351.0	16035.0	South-EastAsia
3	Russia	Europe	1.459409e+08	871894	NaN	14606.0	NaN	676357.0	NaN	180931.0	2300.0	5974.0	100.0	29716907.0	203623.0	Europe
4	South Africa	Africa	5.938157e+07	538184	NaN	9604.0	NaN	387316.0	NaN	141264.0	539.0	9063.0	162.0	3149807.0	53044.0	Africa

In [3]:
```
from sklearn.tree import DecisionTreeClassifier
```

In [4]:
```
features= ['TotalCases']
dt = DecisionTreeClassifier(max_depth = 3) # Increase max_depth to see effect in the plot
```

In [17]:
```
covidNan = covidData[covidData['Continent'].notna()]
```

In [18]:
```
dt.fit(covidNan[features], covidNan['Continent'])
```

Out[18]:
```
DecisionTreeClassifier(max_depth=3)
```

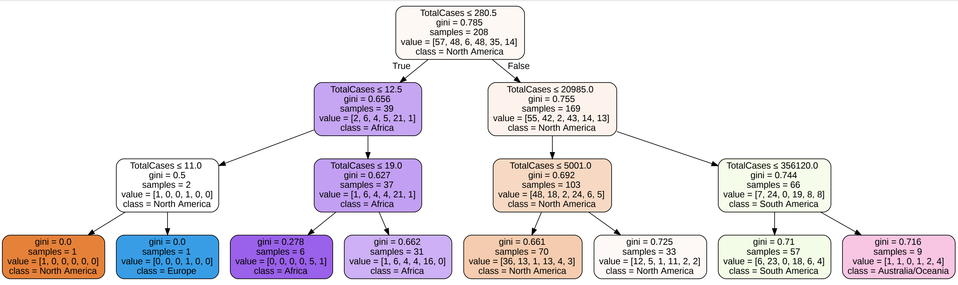
In [19]:
```
conda install -c anaconda python-graphviz
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

# All requested packages already installed.


Note: you may need to restart the kernel to use updated packages.
```

In [20]:
```
from sklearn import tree
import graphviz

def plot_tree_classification(model, features, class_names):
    # Generate plot data
    dot_data = tree.export_graphviz(model, out_file=None,
                                    feature_names=features,
                                    class_names=class_names,
                                    filled=True, rounded=True,
                                    special_characters=True)

    # Turn into graph using graphviz
    graph = graphviz.Source(dot_data)

    # Write out a pdf
    graph.render("decision_tree")

    # Display in the notebook
    return graph
```

In [21]:
```
plot_tree_classification(dt, features, covidNan.Continent.unique())
```

Out[21]:



In [22]:
```
predictions = dt.predict(covidNan[features])
```

In [23]:
```
def calculate_accuracy(predictions, actuals):
    if(len(predictions) != len(actuals)):
        raise Exception("The amount of predictions did not equal the amount of actuals")

    return (predictions == actuals).sum() / len(actuals)
```

In [24]:
```
calculate_accuracy(predictions, covidNan.Continent)
```

Out[24]:
```
0.47115384615384615
```

In [25]:
```
from sklearn.model_selection import train_test_split
```

In [26]:
```
covid_train, covid_test = train_test_split(covidNan, test_size=0.3, stratify=covidNan['Continent'], random_state=42)
print(covid_train.shape, covid_test.shape)
```

```
(145, 16) (63, 16)
```

In [27]:
```
features= ['TotalCases']
dt_classification = DecisionTreeClassifier(max_depth = 1) # Increase max_depth to see effect in the plot
dt_classification.fit(covid_train[features], covid_train['Continent'])
```

Out[27]:
```
DecisionTreeClassifier(max_depth=1)
```

In [30]:
```
predictionsOnTrainset = dt_classification.predict(covid_train[features])
predictionsOnTestset = dt_classification.predict(covid_test[features])

accuracyTrain = calculate_accuracy(predictionsOnTrainset, covid_train.Continent)
accuracyTest = calculate_accuracy(predictionsOnTestset, covid_test.Continent)

print("Accuracy on training set " + str(accuracyTrain))
print("Accuracy on test set " + str(accuracyTest))
```

```
Accuracy on training set 0.3586206896551724
Accuracy on test set 0.3809523809523809
```

In [31]:
```
covid_train
```

Out[31]:

	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovered	ActiveCases	Serious,Critical	Tot Cases/1M pop	Deaths/1M pop	TotalTests	Tests/1M pop	WHO Region
71	Ivory Coast	Africa	26437950.0	16447	NaN	103.0	NaN	12484.0	NaN	3860.0	NaN	622.0	4.0	104584.0	3956.0	Africa
63	Nepal	Asia	29186486.0	21750	NaN	65.0	NaN	15389.0	NaN	6296.0	NaN	745.0	2.0	731977.0	25079.0	South-EastAsia
195	Fiji	Australia/Oceania	897095.0	27	NaN	1.0	NaN	18.0	NaN	8.0	NaN	30.0	1.0	6693.0	7461.0	WesternPacific
130	Sierra Leone	Africa	7992169.0	1877	NaN	67.0	NaN	1427.0	NaN	383.0	NaN	235.0	8.0	NaN	NaN	Africa
72	S. Korea	Asia	51273732.0	14519	20.0	303.0	1.0	13543.0	42.0	673.0	18.0	283.0	6.0	1613652.0	31471.0	WesternPacific
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
196	Saint Lucia	North America	183712.0	25	NaN	NaN	NaN	24.0	NaN	1.0	NaN	136.0	NaN	3895.0	21202.0	Americas
203	Greenland	North America	56780.0	14	NaN	NaN	NaN	14.0	NaN	0.0	NaN	247.0	NaN	5977.0	105266.0	Europe
73	Denmark	Europe	5794279.0	14306	NaN	617.0	NaN	12787.0	NaN	902.0	2.0	2469.0	106.0	1654512.0	285542.0	Europe
55	Azerbaijan	Asia	10148243.0	33247	NaN	479.0	NaN	29275.0	NaN	3493.0	66.0	3276.0	47.0	766179.0	75499.0	Europe
23	Canada	North America	37775022.0	118561	NaN	8966.0	NaN	103106.0	NaN	6489.0	2263.0	3139.0	237.0	4319172.0	114339.0	Americas

145 rows × 16 columns

In [32]:
```
covid_test
```

Out[32]:

	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovered	ActiveCases	Serious,Critical	Tot Cases/1M pop	Deaths/1M pop	TotalTests	Tests/1M pop	WHO Region
119	Slovakia	Europe	5459915.0	2480	NaN	29.0	NaN	1824.0	NaN	627.0	2.0	454.0	5.0	272322.0	49877.0	Europe
189	Belize	North America	398312.0	86	NaN	2.0	NaN	31.0	NaN	53.0	2.0	216.0	5.0	3679.0	9236.0	Americas
202	Saint Kitts and Nevis	North America	53237.0	17	NaN	NaN	NaN	16.0	NaN	1.0	NaN	319.0	NaN	1146.0	21526.0	Americas
35	Belgium	Europe	11594739.0	71158	NaN	9859.0	NaN	17661.0	NaN	43638.0	61.0	6137.0	850.0	1767120.0	152407.0	Europe
150	Gambia	Africa	2422754.0	935	NaN	16.0	NaN	136.0	NaN	783.0	NaN	386.0	7.0	5183.0	2139.0	Africa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
178	Papua New Guinea	Australia/Oceania	8963009.0	163	NaN	3.0	NaN	53.0	NaN	107.0	NaN	18.0	0.3	10808.0	1206.0	WesternPacific
76	Bulgaria	Europe	6942854.0	13014	NaN	435.0	NaN	7374.0	NaN	5205.0	47.0	1874.0	63.0	294087.0	42358.0	Europe
163	Comoros	Africa	871326.0	396	NaN	7.0	NaN	340.0	NaN	49.0	NaN	454.0	8.0	NaN	NaN	Africa
140	Cyprus	Asia	1208238.0	1208	NaN	19.0	NaN	856.0	NaN	333.0	NaN	1000.0	16.0	216597.0	179267.0	Europe
45	Nigeria	Africa	206606300.0	45244	NaN	930.0	NaN	32430.0	NaN	11884.0	7.0	219.0	5.0	306894.0	1485.0	Africa

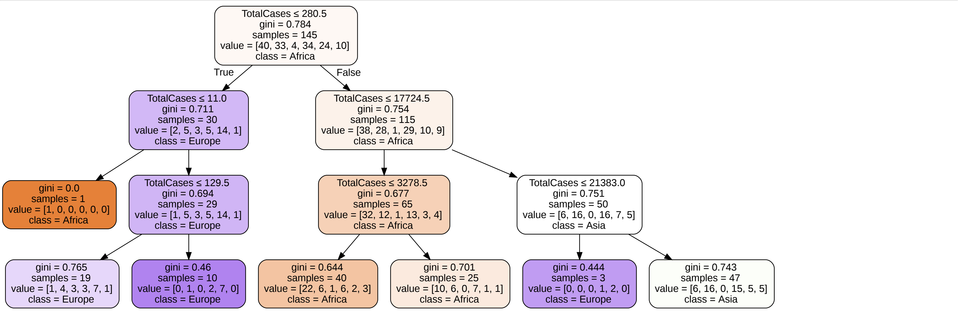
63 rows × 16 columns

In [33]:
```
dt.fit(covid_train[features], covid_train['Continent'])
```

Out[33]:
```
DecisionTreeClassifier(max_depth=3)
```

In [34]:
```
plot_tree_classification(dt, features, covid_train.Continent.unique())
```

Out[34]:



In [36]:
```
dt.fit(covid_test[features], covid_test['Continent'])
```

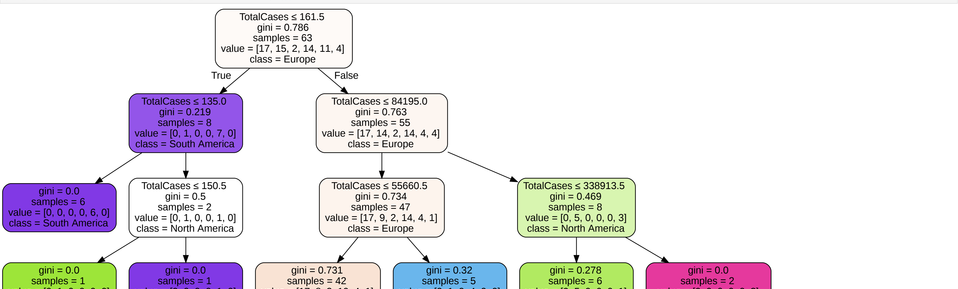
Out[36]:
```
DecisionTreeClassifier(max_depth=3)
```

In [37]:
```
features= ['TotalCases']
dt = DecisionTreeClassifier(max_depth = 3) # Increase max_depth to see effect in the plot
dt.fit(covid_test[features], covid_test['Continent'])
```

Out[37]:
```
DecisionTreeClassifier(max_depth=3)
```

In [38]:
```
plot_tree_classification(dt, features, covid_test.Continent.unique())
```

Out[38]:



The decisiontree checks the TotalCases per country. Every record is either lower or higher. According to this records are being seperated. When looking at the category you can see that certain continents have been affected more than others, but you cant say that the total cases depends on the continent.