In [3]:
```
# Importing numpy, pandas and matplot for visualization
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_theme(rc = {'figure.figsize':(15,8)}, style='white')
```

In [6]:
```
# read the dataset from a CSV file
data = pd.read_csv('Height of Male and Female by Country 2022.csv')
data.head()
```

Out[6]:

	Rank	Country Name	Male Height in Cm	Female Height in Cm	Male Height in Ft	Female Height in Ft
0	1	Netherlands	183.78	170.36	6.03	5.59
1	2	Montenegro	183.30	169.96	6.01	5.58
2	3	Estonia	182.79	168.66	6.00	5.53
3	4	Bosnia and Herzegovina	182.47	167.47	5.99	5.49
4	5	Iceland	182.10	168.91	5.97	5.54

In [3]:
```
# viewing data information
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 199 entries, 0 to 198
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Rank                 199 non-null    int64
 1   Country Name         199 non-null    object
 2   Male Height in Cm    199 non-null    float64
 3   Female Height in Cm  199 non-null    float64
 4   Male Height in Ft    199 non-null    float64
 5   Female Height in Ft  199 non-null    float64
dtypes: float64(4), int64(1), object(1)
memory usage: 9.5+ KB
```

Brief information about the dataset

- Categorical feature contains Country Name and Numerical features contains the other features
- No missing values
- Height in cm and ft convey the same information

In [7]:
```
# viewing the statistical summary of the dataset
data.describe()
```

Out[7]:

	Rank	Male Height in Cm	Female Height in Cm	Male Height in Ft	Female Height in Ft

	Rank	Male Height in Cm	Female Height in Cm	Male Height in Ft	Female Height in Ft
count	199.000000	199.000000	199.000000	199.000000	199.000000
mean	100.000000	173.089045	160.942915	5.678794	5.280402
std	57.590508	4.949832	4.076377	0.162510	0.133870
min	1.000000	160.130000	150.910000	5.250000	4.950000
25%	50.500000	169.490000	158.240000	5.560000	5.190000
50%	100.000000	173.530000	160.620000	5.690000	5.270000
75%	149.500000	176.510000	163.870000	5.790000	5.375000
max	199.000000	183.780000	170.360000	6.030000	5.590000

In [8]:
```
# viewing the unique values on the datatset fonor the columns
data.nunique()
```

Out[8]:
```
Rank                   199
Country Name           199
Male Height in Cm      187
Female Height in Cm    181
Male Height in Ft       60
Female Height in Ft     55
dtype: int64
```

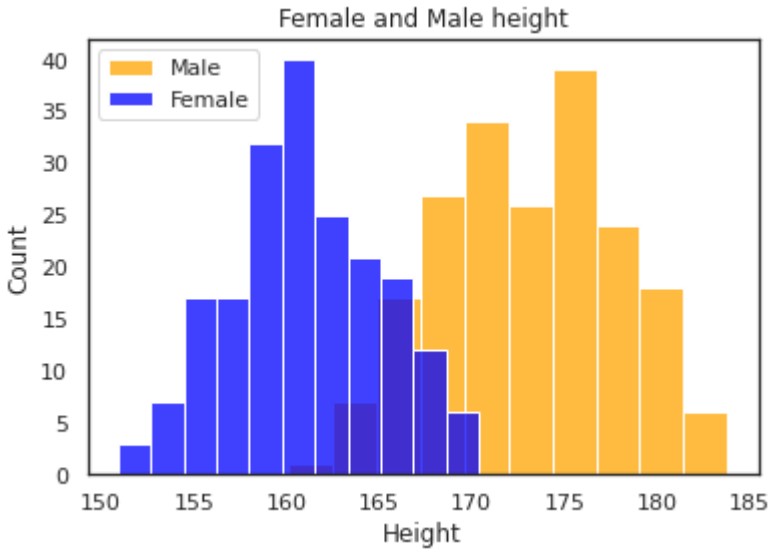
In [6]:
```
data.drop([data.columns[-1], data.columns[-2]], axis=1, inplace=True)
data.columns = ['Rank', 'Country', 'Male', 'Female']
```

In [7]:
```
data.head()
```

Out[7]:

	Rank	Country	Male	Female
0	1	Netherlands	183.78	170.36
1	2	Montenegro	183.30	169.96
2	3	Estonia	182.79	168.66
3	4	Bosnia and Herzegovina	182.47	167.47
4	5	Iceland	182.10	168.91

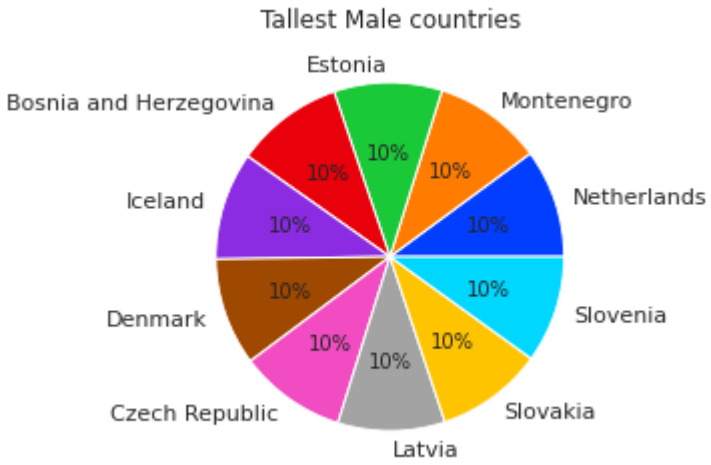
In [8]:
```
# Histogram chart plot for both Male and Female Height
sns.histplot(data.Male, label='Male', color='orange')
sns.histplot(data.Female, label='Female', color='blue')
plt.xlabel('Height')
plt.title('Female and Male height');
plt.legend();
```

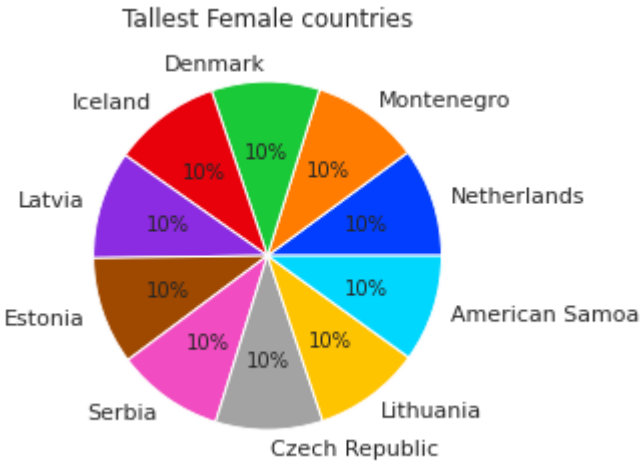


- So there is mild change in ratio, but in acceptable level (the ratio range from 30.45 to 30.51)

In [9]:
```
genders = ['Male', 'Female']
sorted_male_height = data.sort_values(by='Male', ascending=False)
sorted_female_height = data.sort_values(by='Female', ascending=False)
sorted_heights = [sorted_male_height, sorted_female_height]
n = 10
```

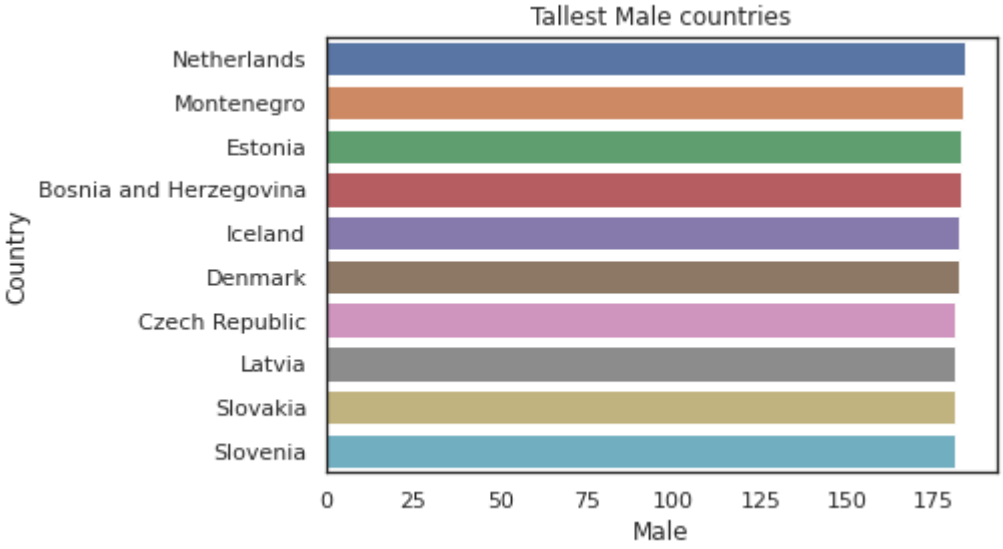
In [10]:
```
# writing a function to plot pie chart for the top 10 countries with tallest male and f
for i in range(2):
    colors = sns.color_palette('bright')
    plt.pie(x=sorted_heights[i][genders[i]][:n], labels=sorted_heights[i].Country[:n],c
    plt.title('Tallest {} countries'.format(genders[i]))
    plt.show()
```

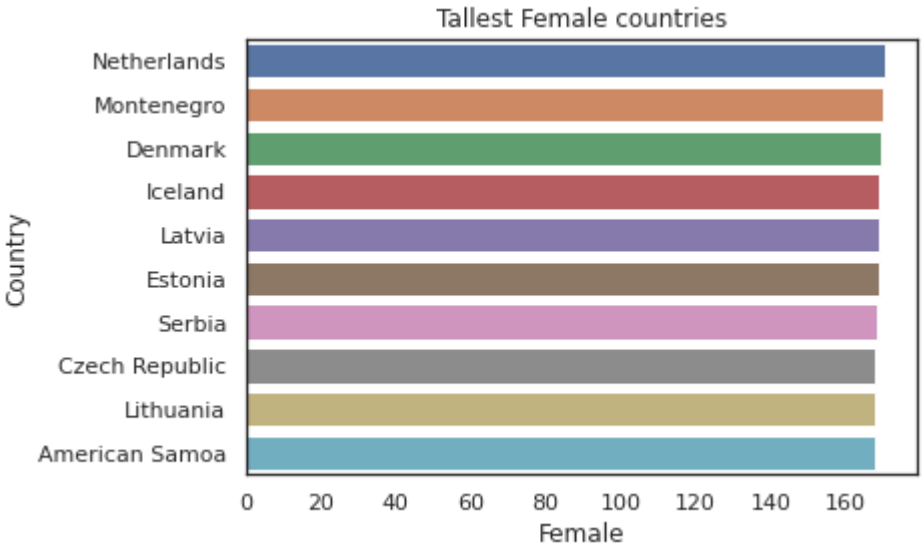




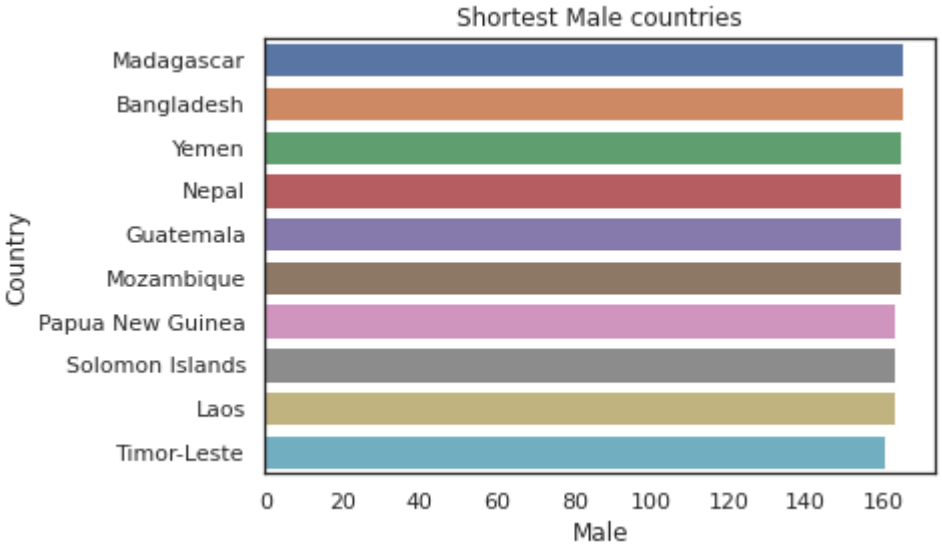
In [11]:

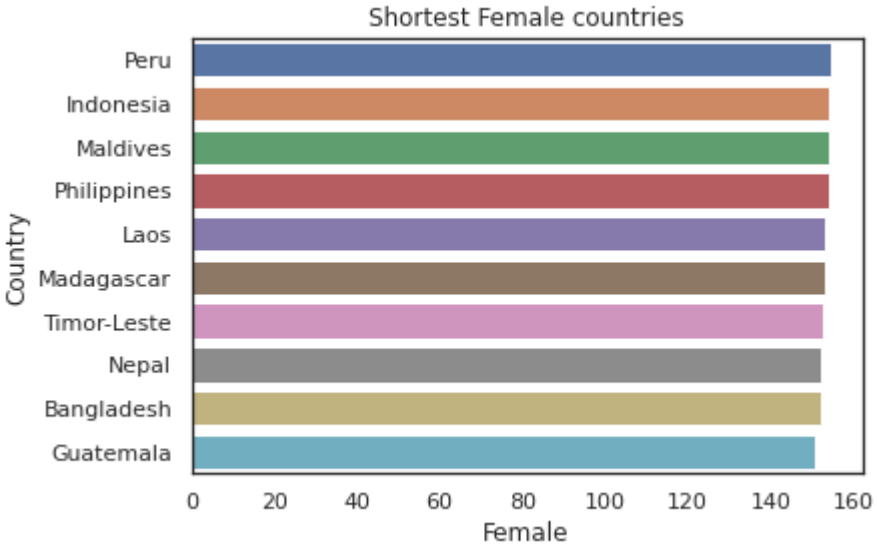
```
# writing a function to plot bar chart for the top 10 countries with tallest male and f
for i in range(2):
    sns.barplot(x=sorted_heights[i][genders[i]][:n], y=sorted_heights[i].Country[:n], c
    plt.title('Tallest {} countries'.format(genders[i]))
    plt.show()
```



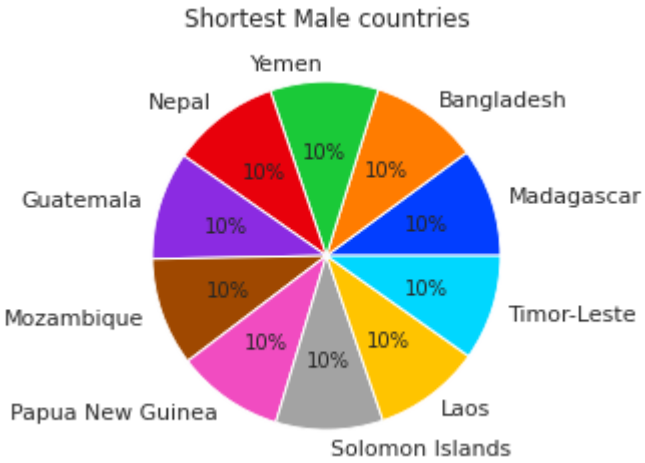


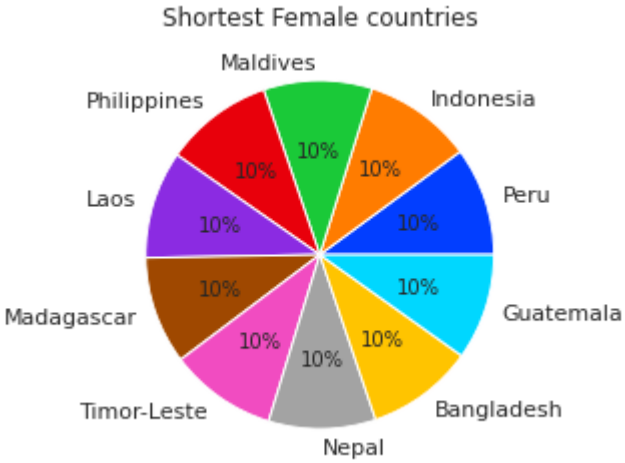
In [12]:
```
# writting a function to plot bar chart for the top 10 countries with shortest male and
for i in range(2):
    sns.barplot(x=sorted_heights[i][genders[i]][-n:], y=sorted_heights[i].Country[-n:],
    plt.title('Shortest {} countries'.format(genders[i]))
    plt.show()
```





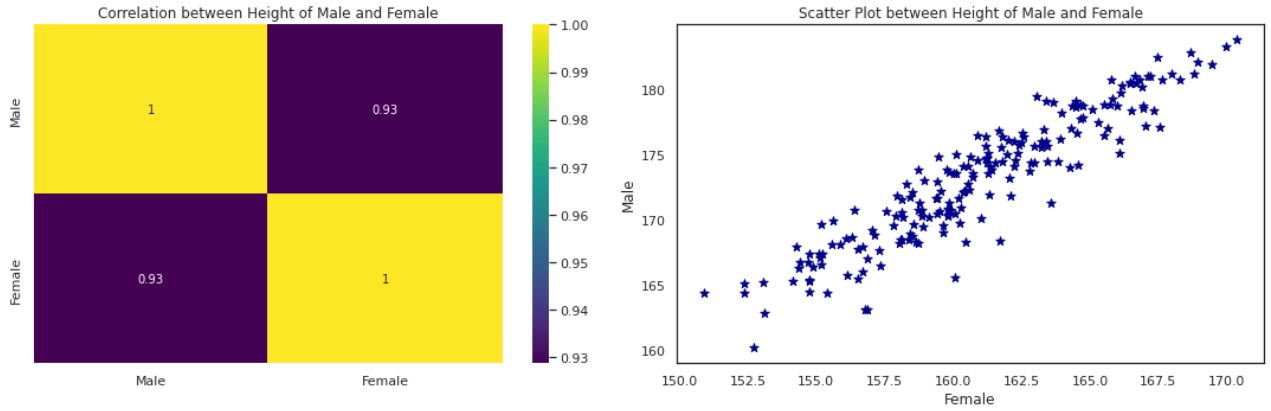
In [13]:
```
# writing a function to plot pie chart for the top 10 countries with shoetest male and
for i in range(2):
    colors = sns.color_palette('bright')
    plt.pie(x=sorted_heights[i][genders[i]][-n:], labels=sorted_heights[i].Country[-n:]
    plt.title('Shortest {} countries'.format(genders[i]))
    plt.show()
```





In [14]:

```
# Lets see the correlation between Male and Female's height 🐣
fig, ax= plt.subplots(ncols=2, figsize=(15,5))
sns.heatmap(data.iloc[:,2:].corr(),annot=True, ax=ax[0], cmap='viridis')
ax[0].set_title('Correlation between Height of Male and Female')
data.plot.scatter(x='Female', y='Male', color='darkblue',
                  marker='*', s=60, ax=ax[1])
ax[1].set_title('Scatter Plot between Height of Male and Female')
fig.tight_layout();
```



No surprise, there is exceedingly high correlation between height of Male and Female. Hence, the country with high average male height is likely to have high mean female's height.

Conclusion on my goals:

- Netherlands and Montenegro are the two countries with the tallest average height of both Male and Female
- Timor-Leste is the country with the shortest average height of Male and Guatemala is the country with the shortest average height of Female
- A country has one of the biggest average height of Male doesn't has to have the biggest of Female and vice versa, like the case of Bosnia and American Samoa
- Tallest countries are mostly from Europe
- Shortest countries are mostly from Asia and South Asia

In [14]: