Exercícios - Estatística I

Questão 1

Os exercícios de 1 a 5 serão desenvolvidos utilizando o *dataset* `Titanic.csv`:

Calcule a frequência absoluta para os sobreviventes no *Titanic*.

Dica.: Utilize a função `.value_counts()`

Carregando as principais bibliotecas que iremos utilizar:

In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# sns.load_dataset('titanic')
```

In [ ]:
```
# Print carregando o dataset Titanic
# titanic = pd.read_csv("titanic.csv")
titanic = sns.load_dataset('titanic')

# Print das primeiras linhas
titanic.head()
```

Out[ ]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

In [ ]:
```
# Print do Título
print("Tabela de frequência ABSOLUTA da coluna 'Survived':")

# Print da Tabela de Frequencia absoluta
display(titanic["survived"].value_counts())
```

```
Tabela de frequência ABSOLUTA da coluna 'Survived':
```

```
0    549
1    342
Name: survived, dtype: int64
```

Questão 2

Os exercícios de 1 a 5 serão desenvolvidos utilizando o *dataset* `Titanic.csv`:

Calcule a frequência relativa, relativa percentual e acumulativa para os sobreviventes no *Titanic*.

Dica.: Utilize a função `.value_counts()`

In [ ]:
```
# Print do Titulo
print("Tabela de frequência RELATIVA da coluna 'Survived':")

# Print da Tabela de Frequencia relativa
display(titanic["survived"].value_counts(normalize=True))
```

```
Tabela de frequência RELATIVA da coluna 'Survived':
0    0.616162
1    0.383838
Name: survived, dtype: float64
```

In [ ]:
```
# Print do Titulo
print("Tabela de frequência PERCENTUAL da coluna 'Survived':")

# Print da Tabela de Frequencia Percentual
titanic["survived"].value_counts(normalize=True).apply(lambda x: str(round(x*100, 2)) + "%")
```

```
Tabela de frequência PERCENTUAL da coluna 'Survived':
```

Out[ ]:
```
0    61.62%
1    38.38%
Name: survived, dtype: object
```

In [ ]:
```
# Print do Titulo
print("Tabela de frequência Acumulada da coluna 'Survived':")

# Print da Tabela de Frequencia Percentual
titanic["survived"].value_counts(normalize=True).cumsum()
```

```
Tabela de frequência Acumulada da coluna 'Survived':
```

Out[ ]:
```
0    0.616162
1    1.000000
Name: survived, dtype: float64
```

Questão 3

Os exercícios de 1 a 5 serão desenvolvidos utilizando o *dataset* `Titanic.csv`:

Utilizando a coluna `Age` do *dataset Titanic*, defina as principais métricas estatísticas para essa variável tais como:

- Média;
- Desvio Padrão;
- Mínimo;
- Primeiro Quartil;
- Segundo Quartil (Mediana);
- Terceiro Quartil;
- Distância Interquartil (IQR);
- Máximo;
- Skewness;
- Moda.

In [ ]:
```
# Calculo das Metricas para Idade
print("Média de idades:", titanic["age"].mean())
print("Desvio padrão de idades:", titanic["age"].std())
print("\nIdade mínima:", titanic["age"].min())
print("\nIdade Q1:", titanic["age"].quantile(0.25))
print("Mediana de idades:", titanic["age"].median())
print("Idade Q3:", titanic["age"].quantile(0.75))
print("IQR das idades:", titanic["age"].quantile(0.75) - titanic["age"].quantile(0.25))
print("\nIdade máxima:", titanic["age"].max())
print("\nSkewness das idades:", titanic["age"].skew())
print("\nIdade(s) mais comum(s):")
display(titanic["age"].mode())
```

```
Média de idades: 29.69911764705882
Desvio padrão de idades: 14.526497332334044

Idade mínima: 0.42

Idade Q1: 20.125
Mediana de idades: 28.0
Idade Q3: 38.0
IQR das idades: 17.875

Idade máxima: 80.0

Skewness das idades: 0.38910778230082704

Idade(s) mais comum(s):
0    24.0
dtype: float64
```

In [ ]:
```
titanic['age'].describe()
```

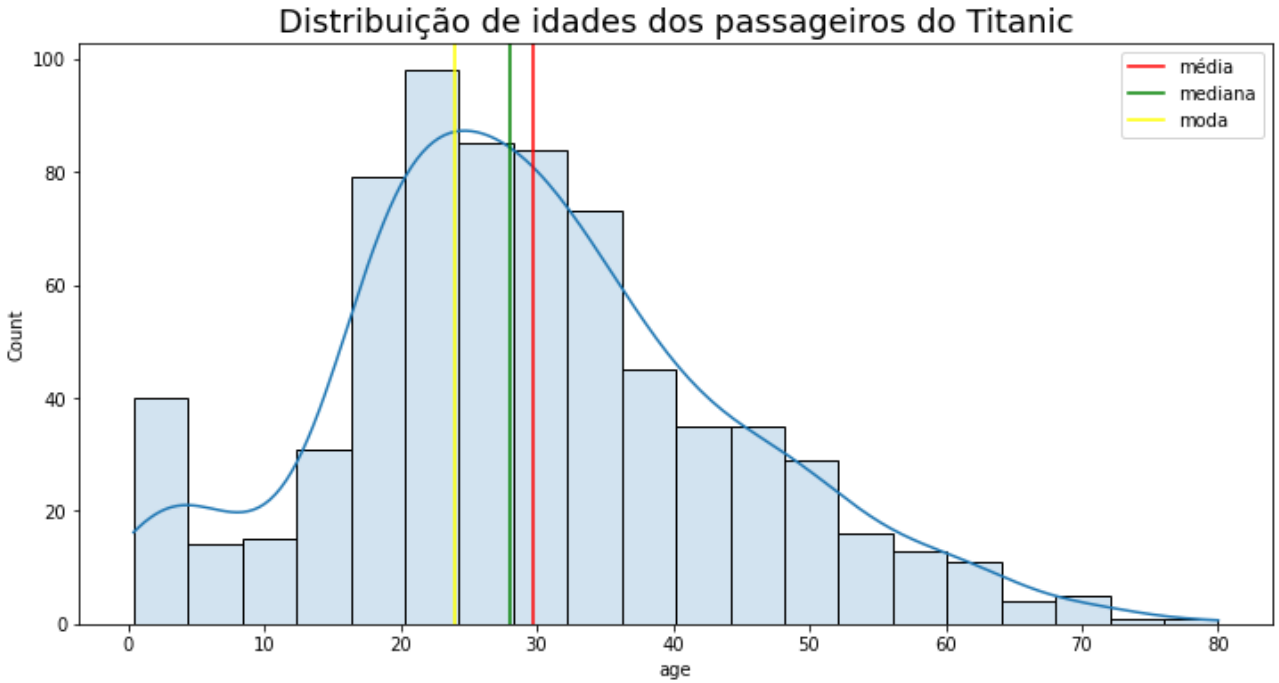
Out[ ]:
```
count    714.000000
mean      29.699118
std       14.526497
min        0.420000
25%       20.125000
50%       28.000000
75%       38.000000
max       80.000000
Name: age, dtype: float64
```

Questão 4

Os exercícios de 1 a 5 serão desenvolvidos utilizando o *dataset* `Titanic.csv`:

Crie um gráfico de distribuição das idades dos passageiros do *Titanic* e identifique os pontos onde se encontram a média, mediana e moda das idades.

In [ ]:
```
# Define o tamanho da figura
plt.figure(figsize=(12, 6))

# Define o Titulo
plt.title("Distribuição de idades dos passageiros do Titanic", size=18)

# Plot do Histograma
sns.histplot(titanic["age"], kde=True, alpha=0.2)

# plotando média
plt.axvline(x=titanic["age"].mean(), color="red", label="média")

# plotando a mediana
plt.axvline(titanic["age"].median(), color="green", label="mediana")

# Loop para plotar as modas
for i in range(titanic["age"].mode().shape[0]):
    plt.axvline(titanic["age"].mode()[i], color="yellow", label="moda")

# Cria uma legenda
plt.legend()

# Mostra o Gráfico
plt.show()
```



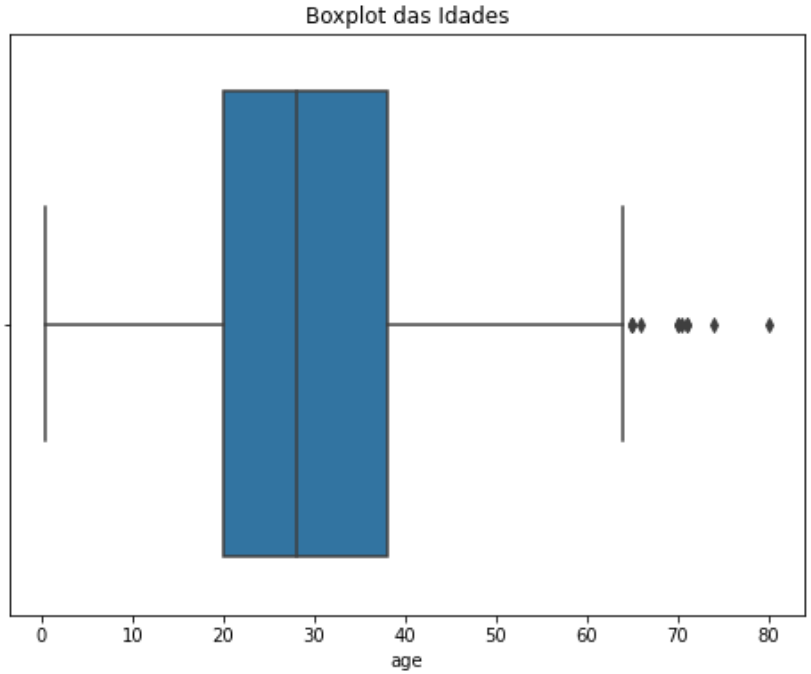
Questão 5

Os exercícios de 1 a 5 serão desenvolvidos utilizando o *dataset* `Titanic.csv`:

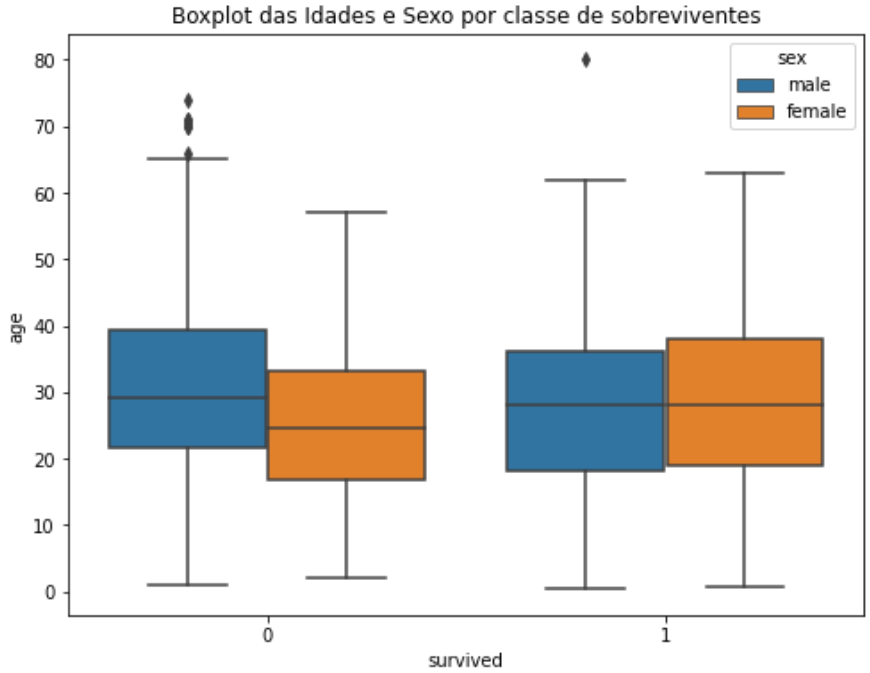
Ainda trabalhando com os valores das idades crie três *Boxplots*:

- *Boxplot* das idades para todos os passageiros;
- *Boxplot* das idades dos passageiros separados pelo sexo.
- *Boxplot* das idades dos passageiros separados pelo sexo e por sobreviventes.

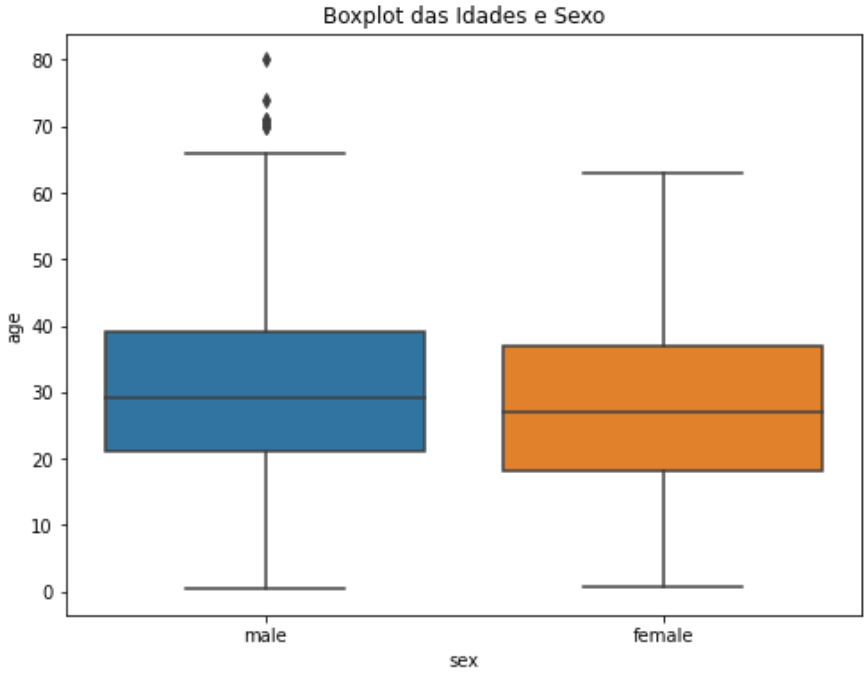
In [ ]:
```
# Define o tamanho da figura
plt.figure(figsize=(8, 6))

# Cria um BoxPlot
sns.boxplot(data=titanic, x="age")

# Cria um titulo
plt.title("Boxplot das Idades")

#Mostra o gráfico
plt.show()
```



In [ ]:
```
# Define o tamanho da figura
plt.figure(figsize=(8, 6))

# Plot do Boxplot
sns.boxplot(data=titanic, y="age", x="survived", hue="sex")

# Cria um titulo
plt.title("Boxplot das Idades e Sexo por classe de sobreviventes")

# Mostra o gráfico
plt.show()
```



In [ ]:
```
# Define o tamanho da figura
plt.figure(figsize=(8, 6))

# Plot do Boxplot
sns.boxplot(data=titanic, y="age", x="sex")

# Cria um titulo
plt.title("Boxplot das Idades e Sexo")

# Mostra o gráfico
plt.show()
```



Com o dataset `penguins`, responda as questões abaixo:

sns.load_dataset('penguins')

Questão 6

Classifique o tipo de dado de cada coluna

In [ ]:
```
penguins = sns.load_dataset('penguins')
penguins
```

Out[ ]:

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	Male
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	Female
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	Female
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	Female
...	...	...	...	...	...	...	...
339	Gentoo	Biscoe	NaN	NaN	NaN	NaN	NaN
340	Gentoo	Biscoe	46.8	14.3	215.0	4850.0	Female
341	Gentoo	Biscoe	50.4	15.7	222.0	5750.0	Male
342	Gentoo	Biscoe	45.2	14.8	212.0	5200.0	Female
343	Gentoo	Biscoe	49.9	16.1	213.0	5400.0	Male

344 rows × 7 columns

In [ ]:
```
penguins.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   species            344 non-null    object 
 1   island             344 non-null    object 
 2   bill_length_mm     342 non-null    float64
 3   bill_depth_mm      342 non-null    float64
 4   flipper_length_mm  342 non-null    float64
 5   body_mass_g        342 non-null    float64
 6   sex                333 non-null    object 
dtypes: float64(4), object(3)
memory usage: 18.9+ KB
```

Quantitativo contínuos: bill_length_mm, bill_depth_mm, flipper_length_mm, body_mass_g,

Qualitativos nominais: species, island, sex

Questão 7

Calcule a frequência absoluta para cada espécie de pinguim

In [ ]:
```
print("Tabela de frequência ABSOLUTA da coluna 'species':")

# Print da Tabela de Frequencia absoluta
display(penguins["species"].value_counts())
```

```
Tabela de frequência ABSOLUTA da coluna 'species':
Adelie       152
Gentoo       124
Chinstrap     68
Name: species, dtype: int64
```

Questão 8

Calcule a frequência relativa, relativa percentual e acumulativa de cada espécie de pinguim

In [ ]:
```
print("Tabela de frequência RELATIVA da coluna 'species':")
display(penguins["species"].value_counts(normalize=True))
```

```
Tabela de frequência RELATIVA da coluna 'species':
Adelie       0.441860
Gentoo       0.360465
Chinstrap    0.197674
Name: species, dtype: float64
```

In [ ]:
```
print("Tabela de frequência PERCENTUAL da coluna 'species':")

# Print da Tabela de Frequencia Percentual
penguins["species"].value_counts(normalize=True).apply(lambda x: str(round(x*100, 2)) + "%")
```

Out[ ]:
```
Tabela de frequência PERCENTUAL da coluna 'species':
Adelie       44.19%
Gentoo       36.05%
Chinstrap    19.77%
Name: species, dtype: object
```

In [ ]:
```
print("Tabela de frequência Acumulada da coluna 'species':")

# Print da Tabela de Frequencia Percentual
penguins["species"].value_counts(normalize=True).cumsum().apply(lambda x: str(round(x*100, 2)) + "%")
```

Out[ ]:
```
Tabela de frequência Acumulada da coluna 'species':
Adelie       44.19%
Gentoo       80.23%
Chinstrap    100.0%
Name: species, dtype: object
```

Questão 9

Utilizando a coluna `body_mass_g` do dataset Penguins, defina as principais métricas estatísticas para essa variável tais como:

- Média;
- Desvio Padrão;
- Mínimo;
- Primeiro Quartil;
- Segundo Quartil (Mediana);
- Terceiro Quartil;
- Distância Interquartil (IQR);
- Máximo;
- Skewness;
- Moda.

In [ ]:
```
penguins['body_mass_g'].describe()
```

Out[ ]:
```
count     342.000000
mean     4201.754386
std       801.954536
min      2700.000000
25%      3550.000000
50%      4050.000000
75%      4750.000000
max      6300.000000
Name: body_mass_g, dtype: float64
```

In [ ]:
```
print("IQR :", penguins["body_mass_g"].quantile(0.75) - penguins["body_mass_g"].quantile(0.25))
print("\nSkewness das massas:", penguins["body_mass_g"].skew())
print("\nIdade(s) mais comum(s):")
display(penguins["body_mass_g"].mode())
```

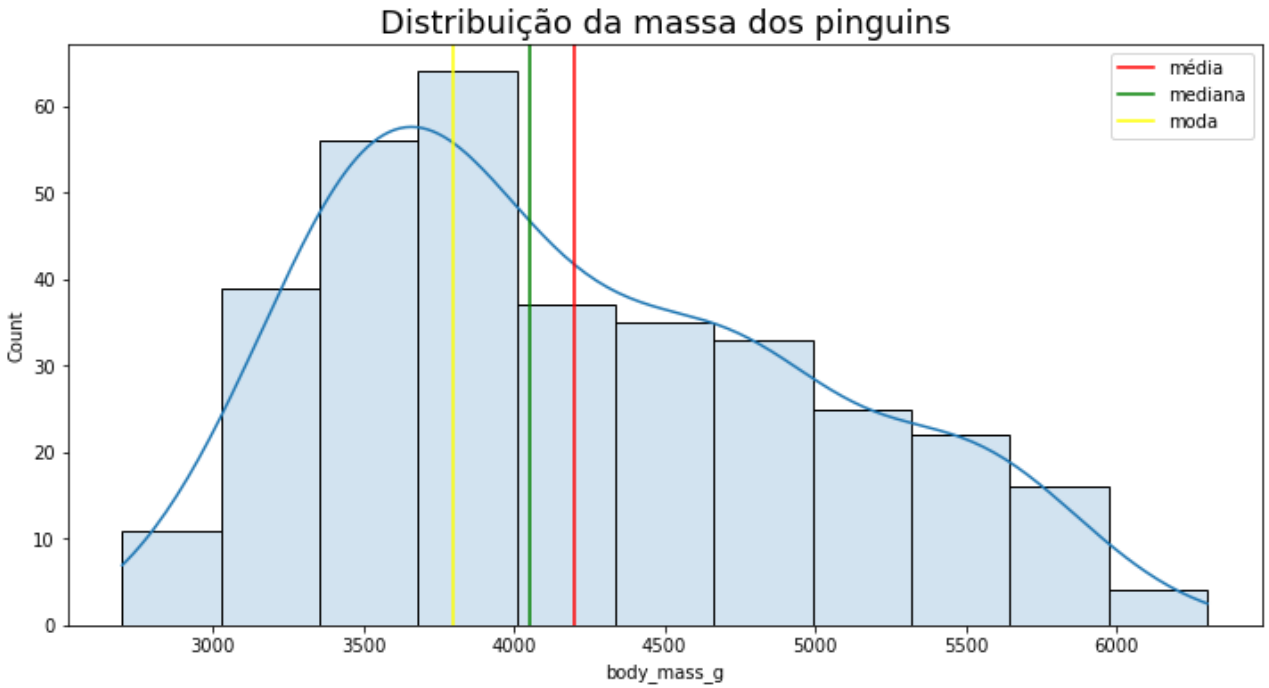
```
IQR : 1200.0

Skewness das massas: 0.470329330480123

Idade(s) mais comum(s):
0    3800.0
dtype: float64
```

Questão 10

Crie um gráfico da distribuição da massa dos pinguins e identifique os pontos onde se encontram a média, mediana e moda das idades.

In [ ]:
```
# Define o tamanho da figura
plt.figure(figsize=(12, 6))

# Define o Titulo
plt.title("Distribuição da massa dos pinguins", size=18)

# Plot do Histograma
```

```python
sns.histplot(penguins["body_mass_g"], kde=True, alpha=0.2)

# plotando média
plt.axvline(x=penguins["body_mass_g"].mean(), color="red", label="média")

# plotando a mediana
plt.axvline(penguins["body_mass_g"].median(), color="green", label="mediana")

# Loop para plotar as modas
for i in range(penguins["body_mass_g"].mode().shape[0]):
    plt.axvline(penguins["body_mass_g"].mode()[i], color="yellow", label="moda")

# Cria uma legenda
plt.legend()

# Mostra o Gráfico
plt.show()
```



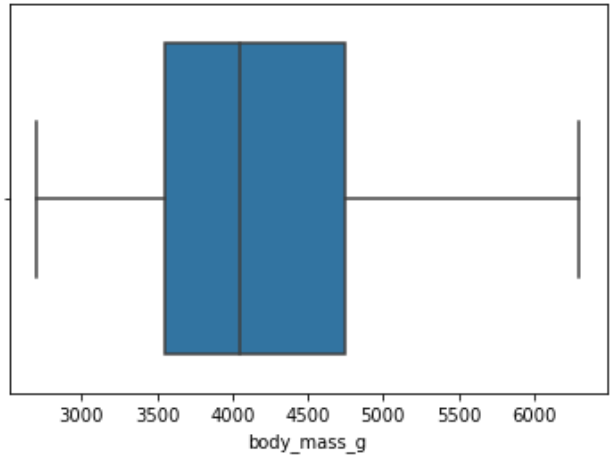
Questão 11

Ainda trabalhando com os valores das massas dos pinguins crie três Boxplots:

- Boxplot da massa para todos os pinguins
- Boxplot da massas dos pinguins separados pelo sexo.
- Boxplot da massas dos pinguins separados pelo sexo e pela espécies.

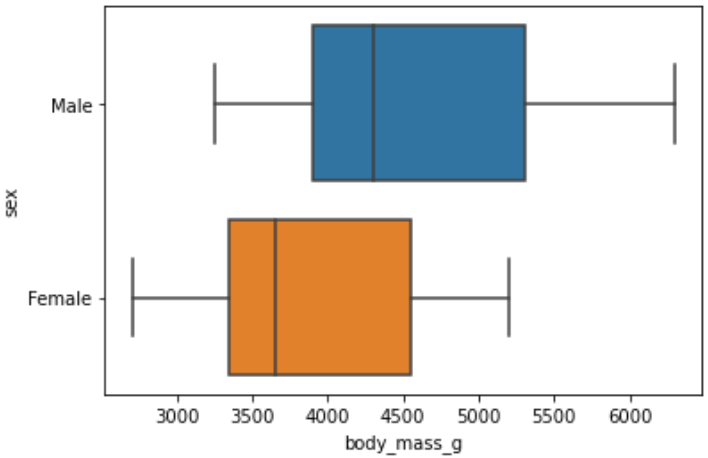
In [ ]:
```
sns.boxplot(data=penguins, x="body_mass_g")#, x="Survived", hue="Sex")
```

Out[ ]:
```
<AxesSubplot:xlabel='body_mass_g'>
```



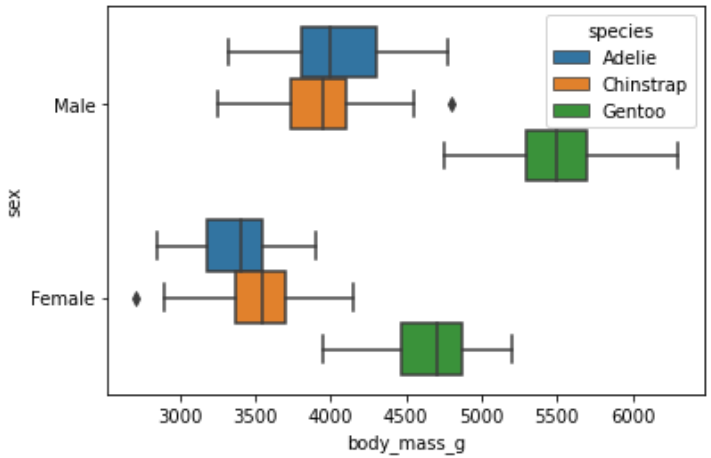
In [ ]:
```
sns.boxplot(data=penguins, x="body_mass_g", y="sex")
```

Out[ ]:
```
<AxesSubplot:xlabel='body_mass_g', ylabel='sex'>
```



In [ ]:
```
sns.boxplot(data=penguins, x="body_mass_g", y="sex", hue='species')
```

Out[ ]:
```
<AxesSubplot:xlabel='body_mass_g', ylabel='sex'>
```


species
Adelie
Chinstrap
Gentoo
Male
Female
sex
3000
3500
4000
4500
5000
5500
6000
body_mass_g

In [ ]:

In [ ]: