Q1: A
Q2: C
Q3: B
Q4: A
Q5: A, C
Q6: B
Q7: A
Q8: B
Q9: B
Q10: B
Q11: B
Q12: A
Q13: C, D
Q14: A
Q15: B
Q16: C
Q17: B
Q18: C
Q19: C
Q20: D
Q21: C
Q22: C
Q23: B, C
Q24: B

Practical Introduction to Data Science - Practice exam

Date: May, 22 2023

First name:
Last name:
Email address:

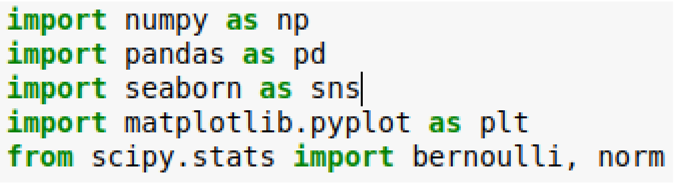
Instructions

The exam is a multiple choice questionnaire. All questions will be graded. Some questions have several correct answers, if you only get a fraction of the correct answers, you will get the same fraction of the points.
Every 4 wrong answers you will lose 0.5 point. Every good answer you will get 1 point. It is possible that one question has multiple good answers (unless specified in the question).
You have 60 minutes to answer this multiple choice questionnaire. Please record your answers on the additional sheet and **don't record** your answer directly below the questions. Make sure that you fill the circles correctly with a black or blue pen. Your answers will be autograded, so make sure to **fill the circles fully**.
If you realized you have made a mistake, please indicate the correct answer next to the associated line.

Return: Please return the questionnaire **and** the single sheet with your answers.

Materials: You can use a single A4 sheet with your own handwritten notes and a (not advanced scientific) calculator. All calculations will be easy and could be solved on paper.

Throughout the questionnaire, we will use several pieces of Python code. We will assume that the following libraries have been imported beforehand.



```python
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy.stats import bernoulli, norm
```

Part 1: Python, datasets and visualization

Question 1 to 3: Which library would we use for these use cases? (Connect Python library and use case)

I.	pandas	1.	Reading in a data set from a csv
II.	numpy	2.	Displaying a map with geographical data
III.	geopandas	3.	Generate a matrix of uniform (pseudo)random numbers

Question 1: I would connect Pandas with

A. 1.
B. 2.
C. 3.

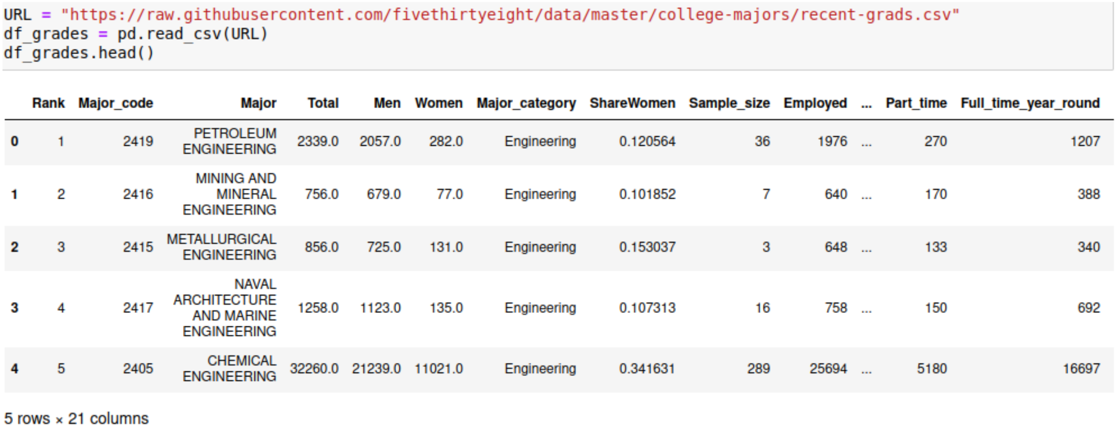
Question 2: I would connect Numpy with

A. 1.
B. 2.
C. 3.

Question 3: I would connect Geopandas with

A. 1.
B. 2.
C. 3.

Question 4: What is the type of the variable “df_grades” after executing this code?



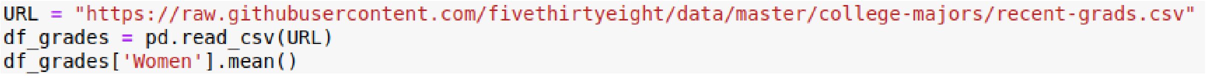
```
URL = "https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/recent-grads.csv"
df_grades = pd.read_csv(URL)
df_grades.head()
```

	Rank	Major_code	Major	Total	Men	Women	Major_category	ShareWomen	Sample_size	Employed	...	Part_time	Full_time_year_round
0	1	2419	PETROLEUM ENGINEERING	2339.0	2057.0	282.0	Engineering	0.120564	36	1976	...	270	1207
1	2	2416	MINING AND MINERAL ENGINEERING	756.0	679.0	77.0	Engineering	0.101852	7	640	...	170	388
2	3	2415	METALLURGICAL ENGINEERING	856.0	725.0	131.0	Engineering	0.153037	3	648	...	133	340
3	4	2417	NAVAL ARCHITECTURE AND MARINE ENGINEERING	1258.0	1123.0	135.0	Engineering	0.107313	16	758	...	150	692
4	5	2405	CHEMICAL ENGINEERING	32260.0	21239.0	11021.0	Engineering	0.341631	289	25694	...	5180	16697

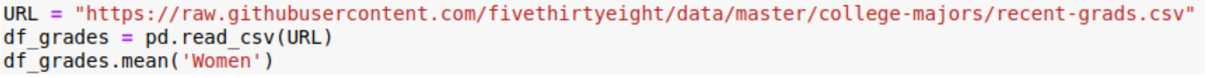
5 rows × 21 columns

A. Pandas DataFrame
B. Pandas Series
C. Pandas String
D. Numpy Matrix
E. Numpy Array

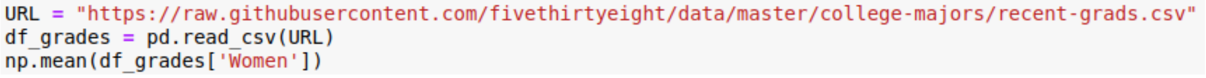
Question 5: Which code computes the mean value of the column “Women” from the variable ‘df_grades’ of the previous question?

A. 

```
URL = "https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/recent-grads.csv"
df_grades = pd.read_csv(URL)
df_grades['Women'].mean()
```

B. 

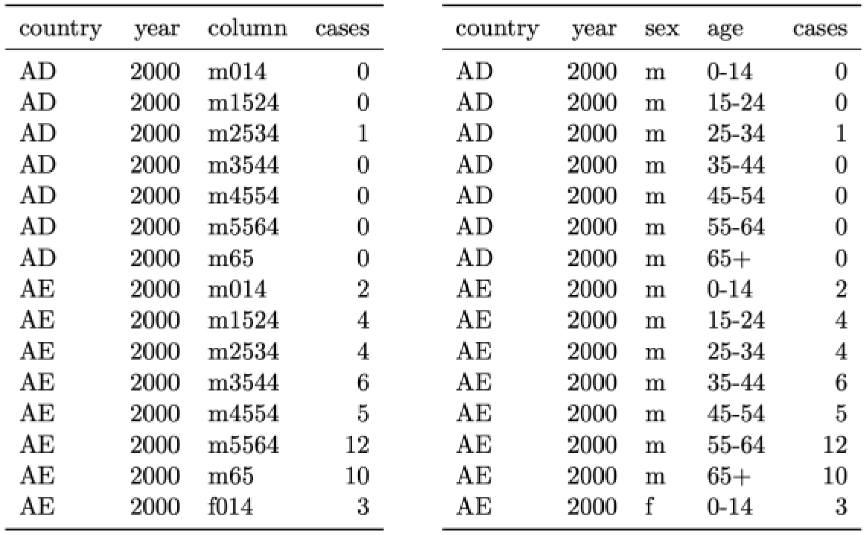
```
URL = "https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/recent-grads.csv"
df_grades = pd.read_csv(URL)
df_grades.mean('Women')
```

C. 

```
URL = "https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/recent-grads.csv"
df_grades = pd.read_csv(URL)
np.mean(df_grades['Women'])
```

Part 2: Dataset and statistics

Question 6: Here are two representations of the same dataset. Which one is tidier?



country	year	column	cases
AD	2000	m014	0
AD	2000	m1524	0
AD	2000	m2534	1
AD	2000	m3544	0
AD	2000	m4554	0
AD	2000	m5564	0
AD	2000	m65	0
AE	2000	m014	2
AE	2000	m1524	4
AE	2000	m2534	4
AE	2000	m3544	6
AE	2000	m4554	5
AE	2000	m5564	12
AE	2000	m65	10
AE	2000	f014	3

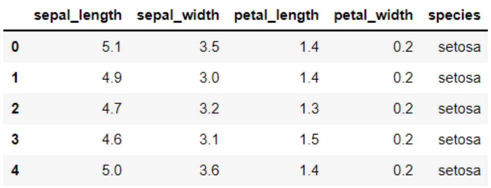
(a) Molten data

country	year	sex	age	cases
AD	2000	m	0-14	0
AD	2000	m	15-24	0
AD	2000	m	25-34	1
AD	2000	m	35-44	0
AD	2000	m	45-54	0
AD	2000	m	55-64	0
AD	2000	m	65+	0
AE	2000	m	0-14	2
AE	2000	m	15-24	4
AE	2000	m	25-34	4
AE	2000	m	35-44	6
AE	2000	m	45-54	5
AE	2000	m	55-64	12
AE	2000	m	65+	10
AE	2000	f	0-14	3

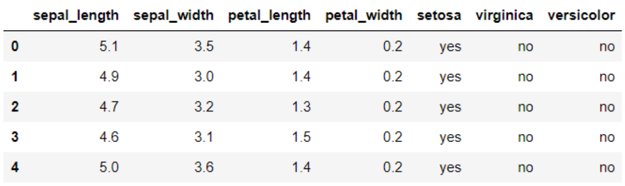
(b) Tidy data

A. The left one because there are less columns
B. The right one because multiple variables are not lumped in one column
C. The right one because the age ranges are properly formatted with a dash

Question 7: Which of these data frames is tidy (in the sense of the lectures)?



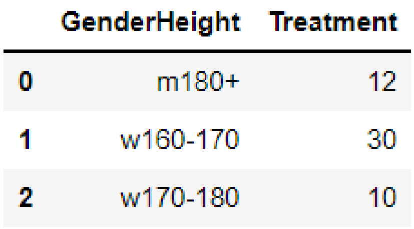
	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa



	sepal_length	sepal_width	petal_length	petal_width	setosa	virginica	versicolor
0	5.1	3.5	1.4	0.2	yes	no	no
1	4.9	3.0	1.4	0.2	yes	no	no
2	4.7	3.2	1.3	0.2	yes	no	no
3	4.6	3.1	1.5	0.2	yes	no	no
4	5.0	3.6	1.4	0.2	yes	no	no

A. Left
B. Right

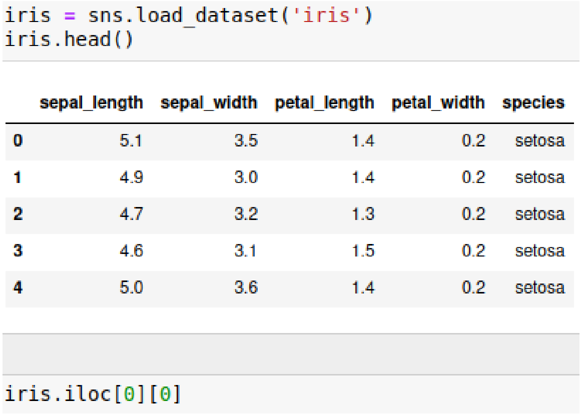
Question 8: Is this table tidy (in the sense of the lectures)?



	GenderHeight	Treatment
0	m180+	12
1	w160-170	30
2	w170-180	10

A. Yes
B. No

Question 9: What is the output of this code?



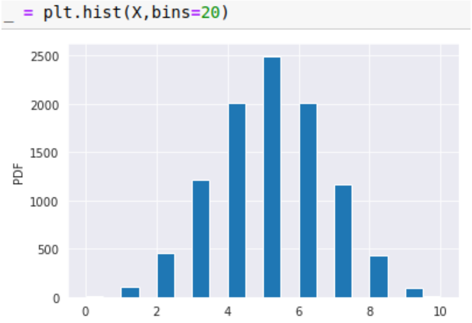
```
iris = sns.load_dataset('iris')
iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
iris.iloc[0][0]
```

A. setosa
B. 5.1
C. 5.0
D. The code will produce an error

Question 10: Based on some samples X, we obtain the following figure. Select the possible way X was generated.



```
_ = plt.hist(X,bins=20)
```

A. X = np.random.binomial(n=10,p=0.1,size=10000)
B. X = np.random.binomial(n=10,p=0.5,size=10000)
C. X = np.random.binomial(n=10,p=0.5,size=1000)
D. X = np.random.binomial(n=20,p=0.5,size=1000)

Question 11: To simulate a game of 100 coin tosses we generate 100 Bernoulli random variables. We are interested in the statistics of such games so we repeat this experiment *N* times using the following code:

```
toss_coin_all = bernoulli.rvs(p=0.5, size=(N,100))
```

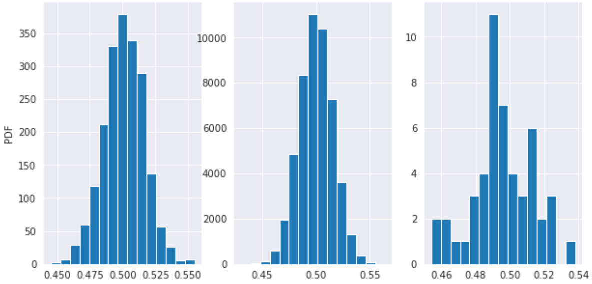
How can we get the mean of each of the *N* experiments?

A. toss_coin_all.mean(axis=0)
B. toss_coin_all.mean(axis=1)
C. np.mean(toss_coin_all)

Question 12: Following the previous question, we store the mean of each of the *N* experiments in a variable named `toss_coin_all_mean`. Then we display the histogram of this array, using the command:

```
plt.hist(toss_coin_all_mean,bins=20)
```

We set *N=2000*. Which figure is the most likely to be seen?



A. Left
B. Middle
C. Right

Question 13: In the previous question, we generated *N* batches of 100 Bernoulli random variables. Let us denote the *m*-th variable in the *n*-th batch by X_{mn}. We then computed the means of all batches to obtain $\overline{X}_n = \frac{1}{100}\sum_{m=1}^{100} X_{mn}$. Next, we plotted the histogram of $\{\overline{X}_1, \ \dots, \overline{X}_N\}$. The histogram can be approximated by a normal distribution with a certain mean and variance. Which of the following statements about the mean and the variance are true?

A. Mean is 0 and variance is 1
B. Mean and/or variance depend on the number of experiment N
C. Mean and/or variance depend on the number of element in the empirical mean (here 100)
D. Mean and/or variance depend on the parameter of the Bernoulli distribution (here p=0.5)
E. Mean and/or variance depend on some quantile of the Normal distribution

Question 14: This question is independent of the previous ones. Let us now simulate a game of 1000 coin tosses by generating 1000 Bernoulli random variables $\{\overline{X}_1, \ \dots, \overline{X}_N\}$ with *N=1000*. The theoretical mean of these random variables is *0.5* and the theoretical variance is *0.25*. We compute the empirical mean of the observations $\overline{X} = \frac{1}{1000}\sum_{n=1}^{1000} X_n$ and obtain $\overline{X} = 0.483$. We are wondering if the true mean is indeed *0.5*. Following the lectures, we compute the asymptotic 95% confidence interval around the true mean (using central limit theorem approximation) by using the formula:

$$I = [\bar{X} - 1.96 \times \frac{\sqrt{0.25}}{Y}, \bar{X} + 1.96 \times \frac{\sqrt{0.25}}{Y}].$$

What is the correct value *Y*?

A. $\sqrt{N}$
B. N
C. N^2

Question 15: Following the previous question, we observe $\overline{X} = 0.483$ and compute the following interval

$$I = [0.463, \ 0.503].$$

What is the probability that the true mean is within this interval?

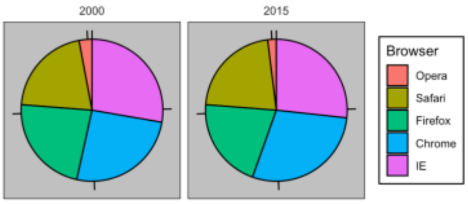
A. 5%
B. 95%
C. 0.05%
D. 0.95%

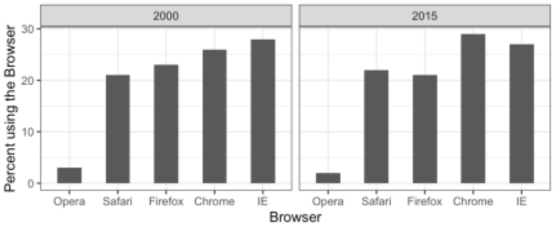
Part 3: Visualization

Question 16: Which graph type fits the following use case best: “Histogram of student heights”

A. Scatter graph
B. Line graph
C. Bar chart

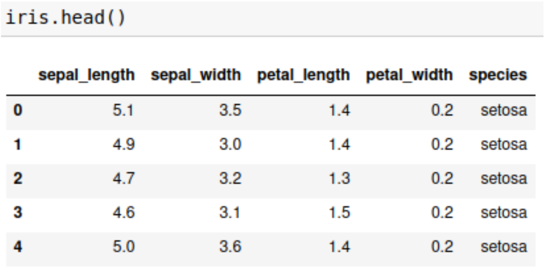
Question 17: Which of the two figures is preferable to visualize the the market shares of each browser?





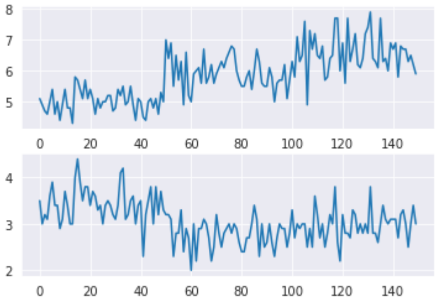
A. Pie chart
B. Bar plot
C. None of the above, a plot with a linear regression would have been better

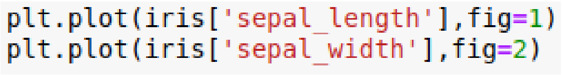
Question 18: Which code could have generated this graph from the dataset iris?



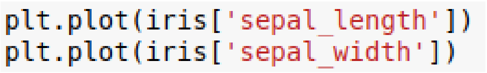
```
iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

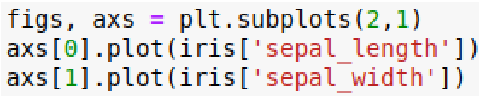


A.


```
plt.plot(iris['sepal_length'],fig=1)
plt.plot(iris['sepal_width'],fig=2)
```

B.


```
plt.plot(iris['sepal_length'])
plt.plot(iris['sepal_width'])
```

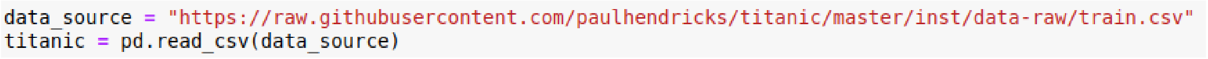
C.


```
figs, axs = plt.subplots(2,1)
axs[0].plot(iris['sepal_length'])
axs[1].plot(iris['sepal_width'])
```

D. None of them

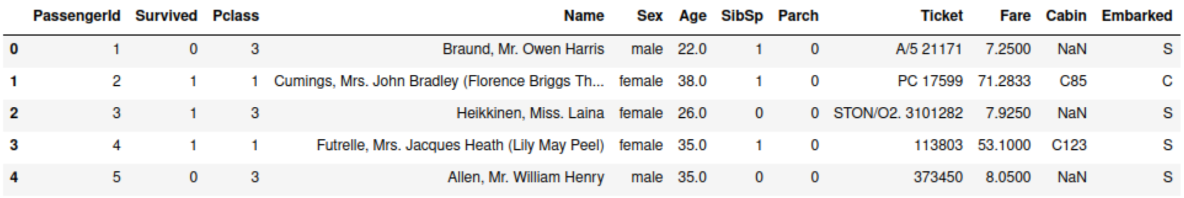
Part 4: Titanic dataset

In the following questions we will work with the Titanic dataset (known from exercise sheet 9)We load the dataset using the these two lines:



```
data_source = "https://raw.githubusercontent.com/paulhendricks/titanic/master/inst/data-raw/train.csv"
titanic = pd.read_csv(data_source)
```

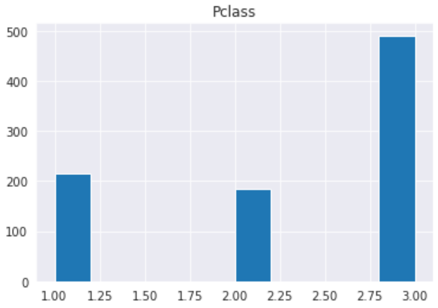
Question 19: How can we display the first 5 lines of the dataset?



	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

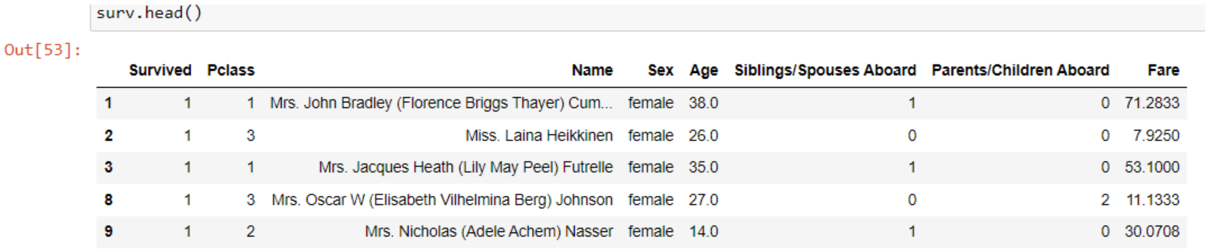
A. print(titanic)
B. plt.plot(titanic)
C. titanic.head()
D. titanic.row(5)

Question 20: Which code could have produced this histogram of the passenger classes?



A. plt.plot(titanic.hist)["Pclass"]
B. titanic.lineplot("Pclass")
C. np.hist(np.titanic["Pclass"])
D. titanic.hist("Pclass")

Question 21: Which code could have produced this subsection of the Titanic dataset containing only people who survived?



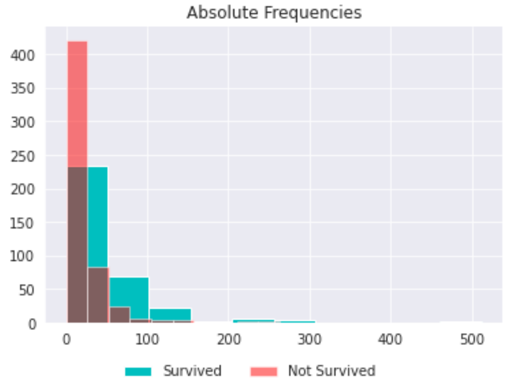
```
surv.head()
```

Out[53]:

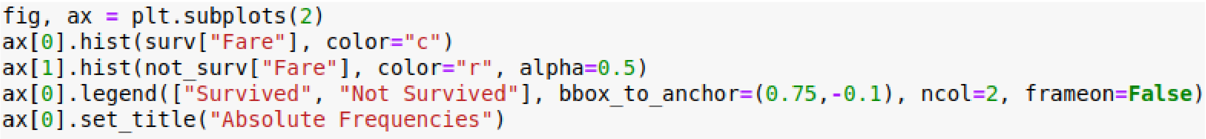
	Survived	Pclass	Name	Sex	Age	Siblings/Spouses Aboard	Parents/Children Aboard	Fare
1	1	1	Mrs. John Bradley (Florence Briggs Thayer) Cum...	female	38.0	1	0	71.2833
2	1	3	Miss. Laina Heikkinen	female	26.0	0	0	7.9250
3	1	1	Mrs. Jacques Heath (Lily May Peel) Futrelle	female	35.0	1	0	53.1000
8	1	3	Mrs. Oscar W (Elisabeth Vilhelmina Berg) Johnson	female	27.0	0	2	11.1333
9	1	2	Mrs. Nicholas (Adele Achem) Nasser	female	14.0	1	0	30.0708

A. surv = titanic["Survived"] == 1
B. surv = titanic["Survived"]
C. surv = titanic[titanic["Survived"] == 1]
D. surv = titanic[titanic["Survived"]]
E. surv = titanic[“Survived” == 1]

Question 22: Following the previous question, we also extract a data frame of passangers that did not survive in the variable ‘not_surv’.
Which code generates the histograms of fares paid for both passengers that survived and passengers that did not survive on the same plot?

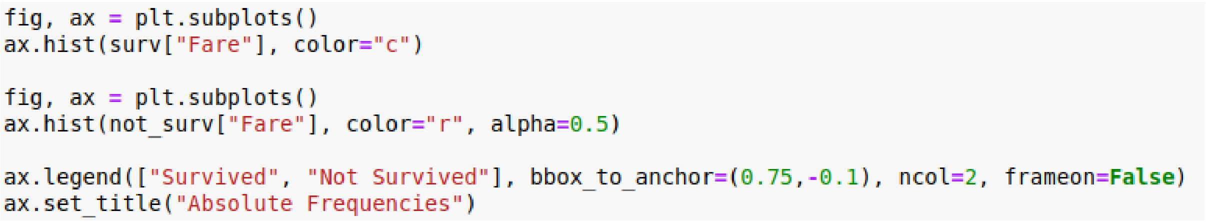


A.



```
fig, ax = plt.subplots(2)
ax[0].hist(surv["Fare"], color="c")
ax[1].hist(not_surv["Fare"], color="r", alpha=0.5)
ax[0].legend(["Survived", "Not Survived"], bbox_to_anchor=(0.75,-0.1), ncol=2, frameon=False)
ax[0].set_title("Absolute Frequencies")
```

B.

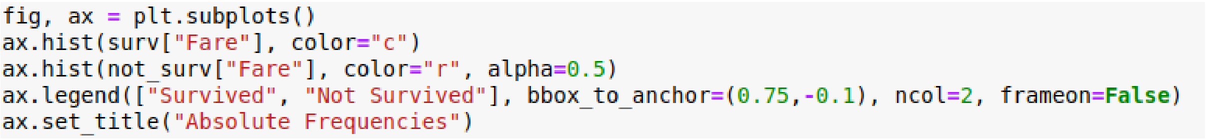


```
fig, ax = plt.subplots()
ax.hist(surv["Fare"], color="c")

fig, ax = plt.subplots()
ax.hist(not_surv["Fare"], color="r", alpha=0.5)

ax.legend(["Survived", "Not Survived"], bbox_to_anchor=(0.75,-0.1), ncol=2, frameon=False)
ax.set_title("Absolute Frequencies")
```

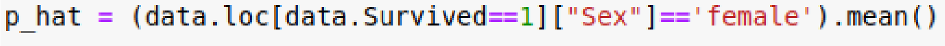
C.



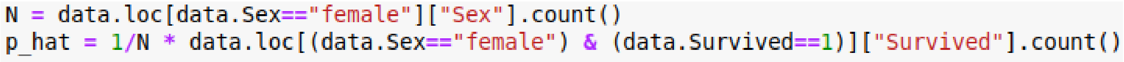
```
fig, ax = plt.subplots()
ax.hist(surv["Fare"], color="c")
ax.hist(not_surv["Fare"], color="r", alpha=0.5)
ax.legend(["Survived", "Not Survived"], bbox_to_anchor=(0.75,-0.1), ncol=2, frameon=False)
ax.set_title("Absolute Frequencies")
```

D. None of them

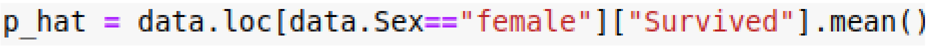
Question 23: We want to estimate the probability that a randomly selected female passenger survived the Titanic disaster. We model these survival outcomes as independent, identically distributed Bernoulli random variables X_1, ..., X_N (1 for survival, 0 otherwise), with probability of survival equal to *p*. Which code computes a plausible estimate $\hat{p}$ of p ?

A. 

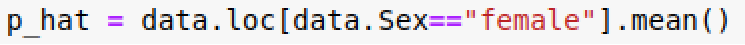
```
p_hat = (data.loc[data.Survived==1]["Sex"]=='female').mean()
```

B. 

```
N = data.loc[data.Sex=="female"]["Sex"].count()
p_hat = 1/N * data.loc[(data.Sex=="female") & (data.Survived==1)]["Survived"].count()
```

C. 

```
p_hat = data.loc[data.Sex=="female"]["Survived"].mean()
```

D. 

```
p_hat = data.loc[data.Sex=="female"].mean()
```

E. None of the above

Question 24: Which mathematical fact about averages is leveraged in the above question?

A. Law of moderate numbers
B. Law of large numbers
C. The Riemann hypothesis (assuming it is true)
D. The normality corollary