```
# 1. The following questions use the data collected from the anonymous survey
collected at the beginning of the course and from previous semesters. The dataset
can be downloaded from the Datasets module on Canvas. Perform the following
steps.
# a. Load the survey data into a variable called "survey"
!pip install pandas
import pandas as pd
survey = pd.read_csv(
"https://drive.google.com/uc?export=download&id=1eqckt-gt5XIOW2OATNIVoJcVuYFOwdMz"
)
survey
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages
(2.1.4)
Requirement already satisfied: numpy<2,>=1.22.4 in
/usr/local/lib/python3.10/dist-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.10/dist-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.1 in
/usr/local/lib/python3.10/dist-packages (from pandas) (2024.1)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas)
(1.16.0)
```

	Semester	ComputerScience	Math	Statistics	MachineLearning	DomainExpertise	Communicatio
0	Spring2019	1.0	1.0	1.0	1.0	1.0	1.0
1	Spring2019	6.0	5.0	3.0	1.0	1.0	1.0
2	Spring2019	7.0	7.0	7.0	6.0	4.0	8.0
3	Spring2019	8.0	8.0	6.0	6.0	9.0	9.0
4	Spring2019	8.0	7.0	6.0	7.0	7.0	8.0
...	...	...	...	...	...	...	...
611	Fall2024	6.0	6.0	4.0	2.0	3.0	6.0
612	Fall2024	6.0	7.0	5.0	1.0	1.0	6.0
613	Fall2024	7.0	8.0	5.0	7.0	2.0	3.0
614	Fall2024	6.0	7.0	6.0	5.0	3.0	8.0
615	Fall2024	5.0	7.0	8.0	7.0	3.0	7.0

```
# b. How many rows are there in the data? (code, answer)
print("Number of rows in the data:", len(survey))
```

```
Number of rows in the data: 616
```

```python
# c. Are there any NAs in the dataset?
print("Number of NAs in the dataset:")
print(survey.isna().sum())
```

```
Number of NAs in the dataset:
Semester           0
ComputerScience    1
Math               2
Statistics         1
MachineLearning    2
DomainExpertise    1
Communication      3
Visualization      4
TakenCPSC483       2
PlanCPSC483        6
MajorCS            1
FamiliarR          2
FamiliarPython     3
dtype: int64
```

```python
# d. How many rows have at least one NA?
print("Number of rows with at least one NA:")
print(survey.isna().any(axis=1).sum())
```

```
Number of rows with at least one NA:
18
```

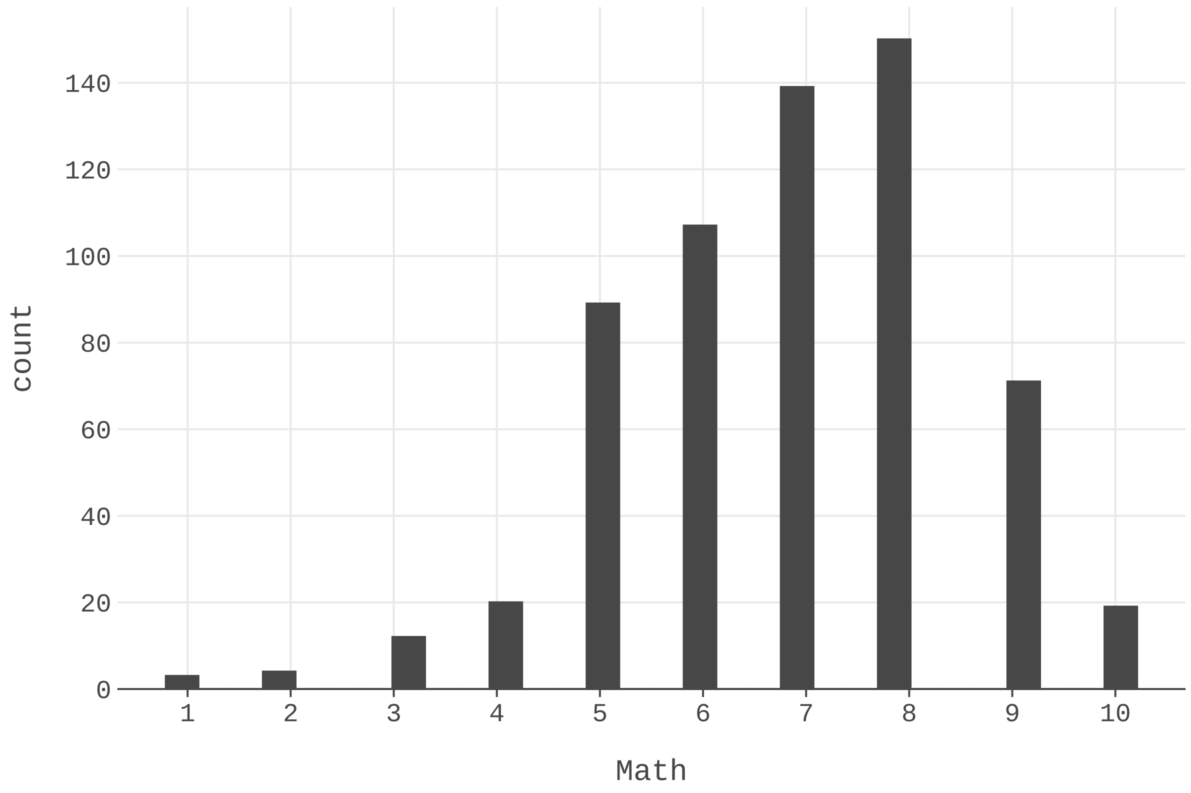
```python
# e. Write Python code to remove all rows which contain an NA value. Give your
Python code. How many rows does your data contain?
survey = survey.dropna()
print("Number of rows after removing rows with NA values:", len(survey))
```

```
Number of rows after removing rows with NA values: 598
```

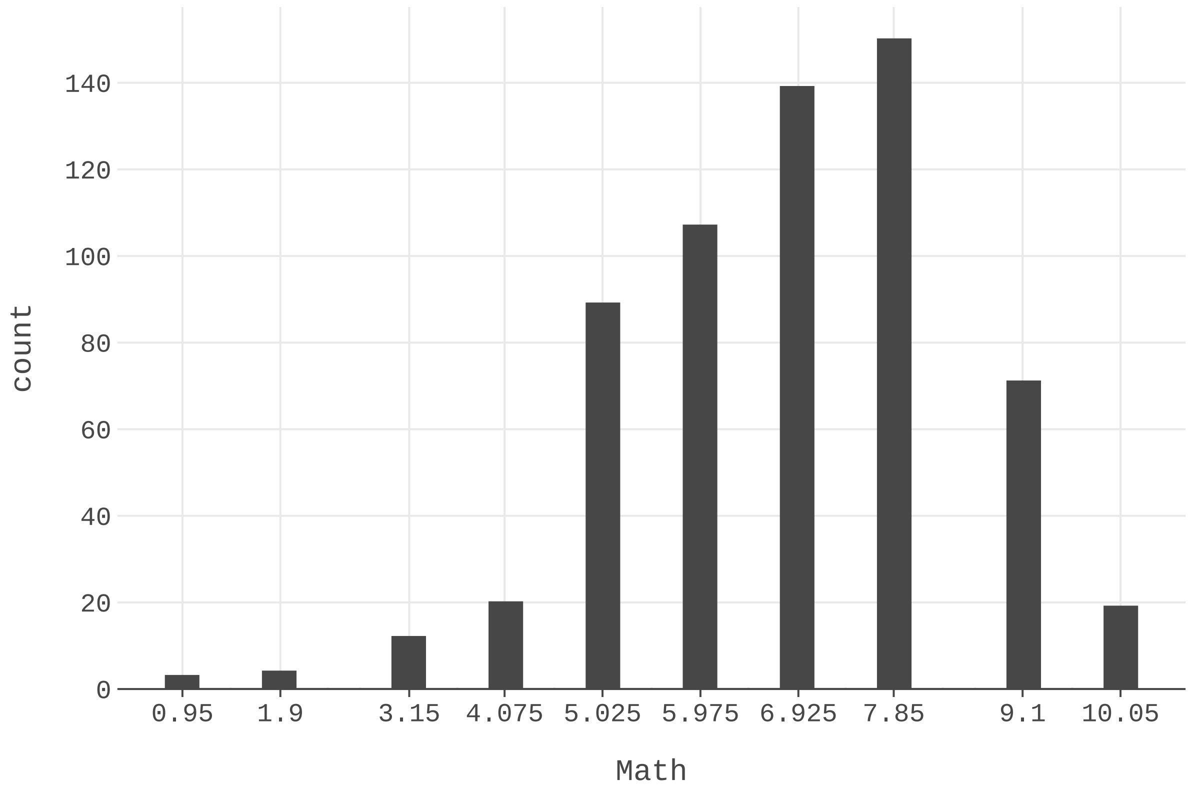
```python
# 2. Use the same data from the previous question (after removing NAs) to
generate the following graphs. All plots should use Lets-Plot. Include both the
Python code and the plot itself.
!pip install lets-plot
from lets_plot import *
LetsPlot.setup_html()

# This goes after the call to LetsPlot.setup_html(),
# but before calling ggplot().

def setup_colab2pdf():
  def _repr_svg_(self):
    from io import BytesIO
    from sys import stdout
    file_like = BytesIO()
    self.to_svg(file_like)
    return file_like.getvalue().decode(stdout.encoding)
  import lets_plot
  lets_plot.plot.core.PlotSpec._repr_svg_ = _repr_svg_

setup_colab2pdf()
```

```
Collecting lets-plot
  Downloading
  lets_plot-4.4.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
  (10.0 kB)
Collecting pypng (from lets-plot)
  Downloading pypng-0.20220715.0-py3-none-any.whl.metadata (13 kB)
Collecting palettable (from lets-plot)
  Downloading palettable-3.3.3-py2.py3-none-any.whl.metadata (3.3 kB)
Downloading
lets_plot-4.4.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.1
MB)
                         3.1/3.1 MB 4.7 MB/s eta 0:00:00
Downloading palettable-3.3.3-py2.py3-none-any.whl (332 kB)
                         332.3/332.3 kB 20.4 MB/s eta 0:00:00
Downloading pypng-0.20220715.0-py3-none-any.whl (58 kB)
                         58.1/58.1 kB 4.1 MB/s eta 0:00:00
Installing collected packages: pypng, palettable, lets-plot
Successfully installed lets-plot-4.4.1 palettable-3.3.3 pypng-0.20220715.0
```

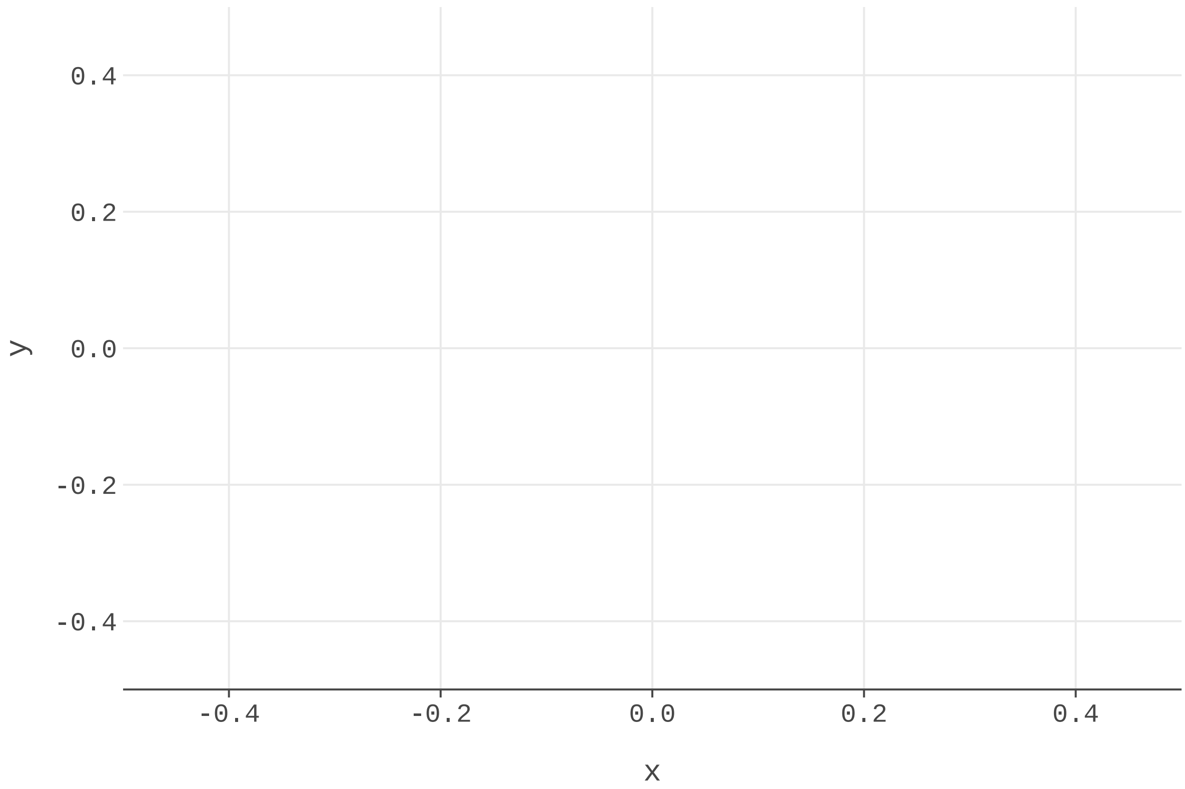
```
# a. Plot a histogram of variable Math
ggplot(survey, aes(x="Math")) + geom_histogram()
```



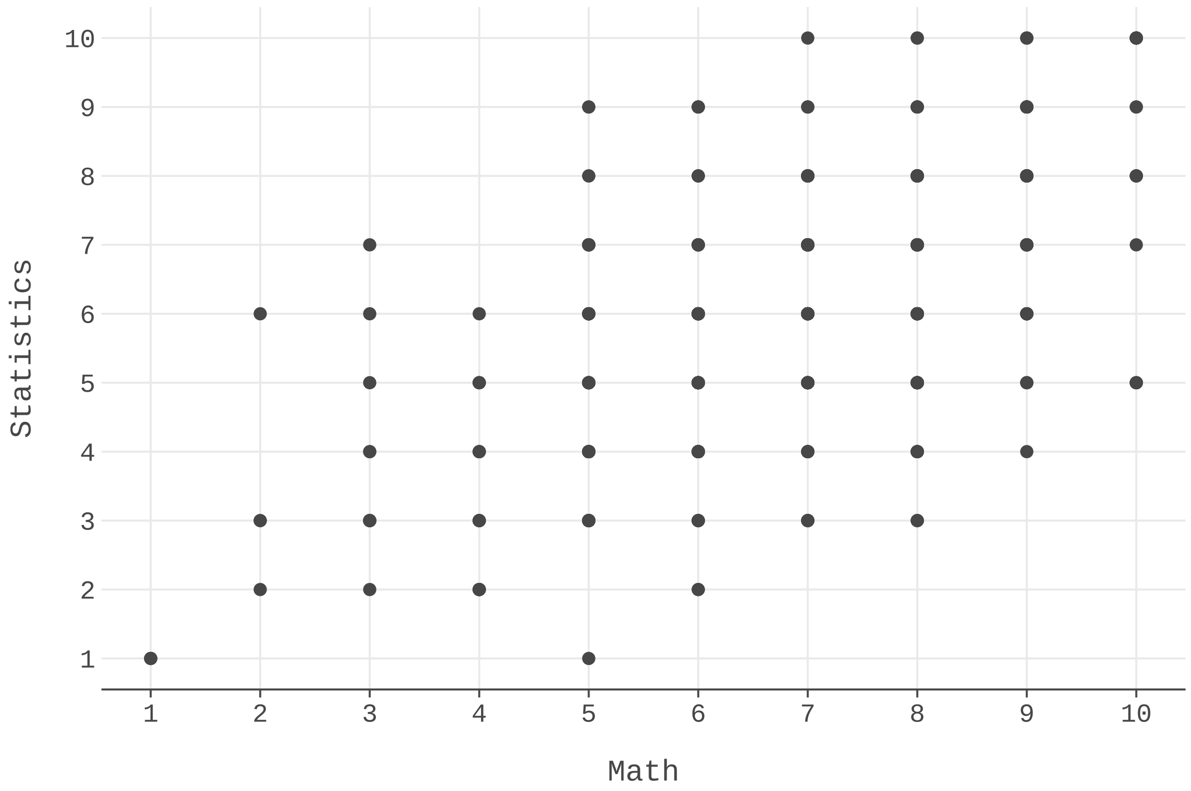
```
# b. The plot above likely has x-axis "breaks" not aligned with the bars. Provide
your own breaks to match the bars.
ggplot(survey, aes(x="Math")) + geom_histogram() +
scale_x_continuous(breaks=[.95, 1.9, 3.15, 4.075, 5.025, 5.975, 6.925, 7.85,
9.10, 10.05])
```



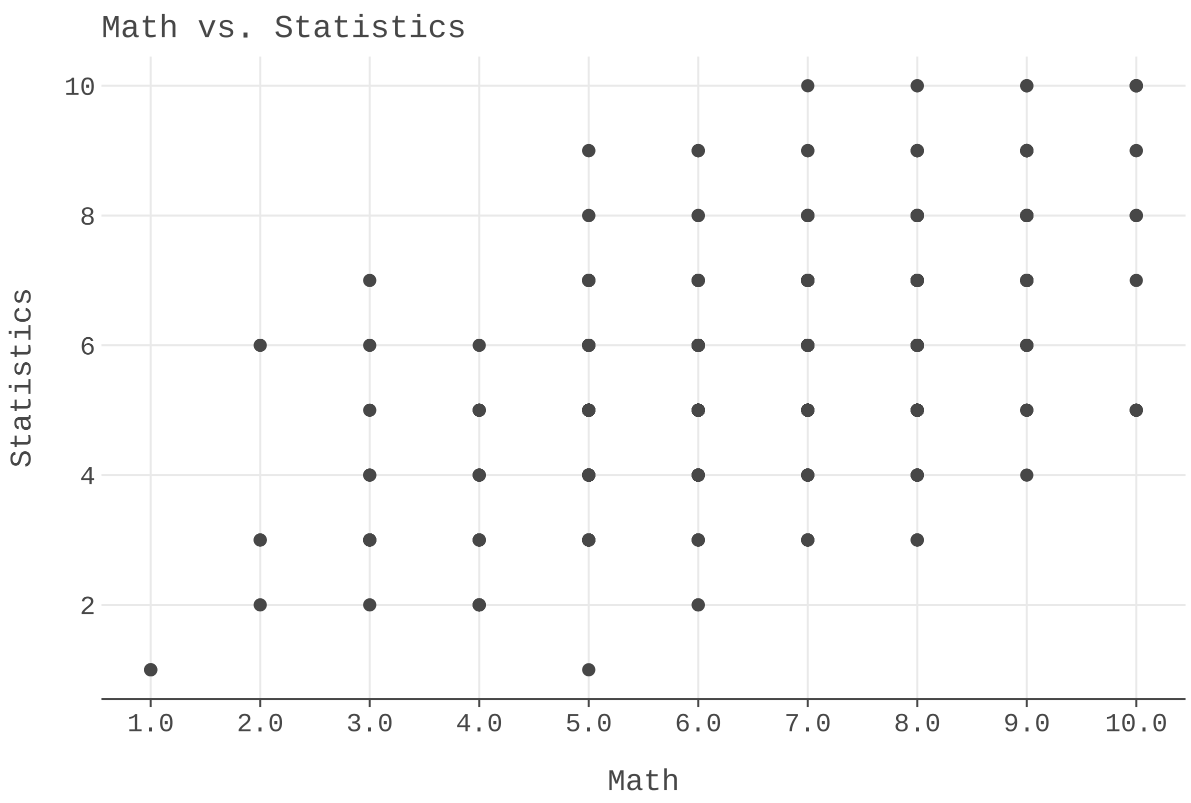
```
# c. Plot a boxplot of variable Math
ggplot(survey, aes(x="Math")) + geom_boxplot()
```



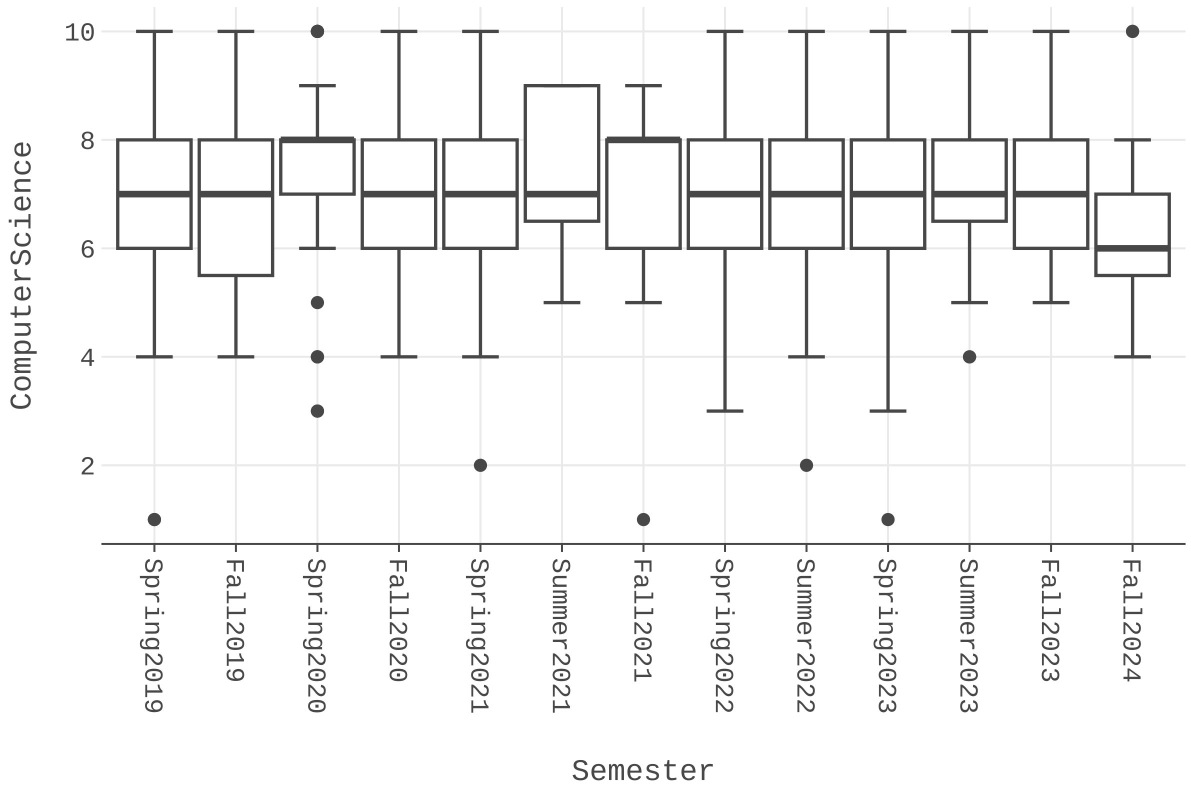
```
# d. Plot a scatterplot of variables Math and Statistics.
ggplot(survey, aes(x="Math", y="Statistics")) + geom_point()
```



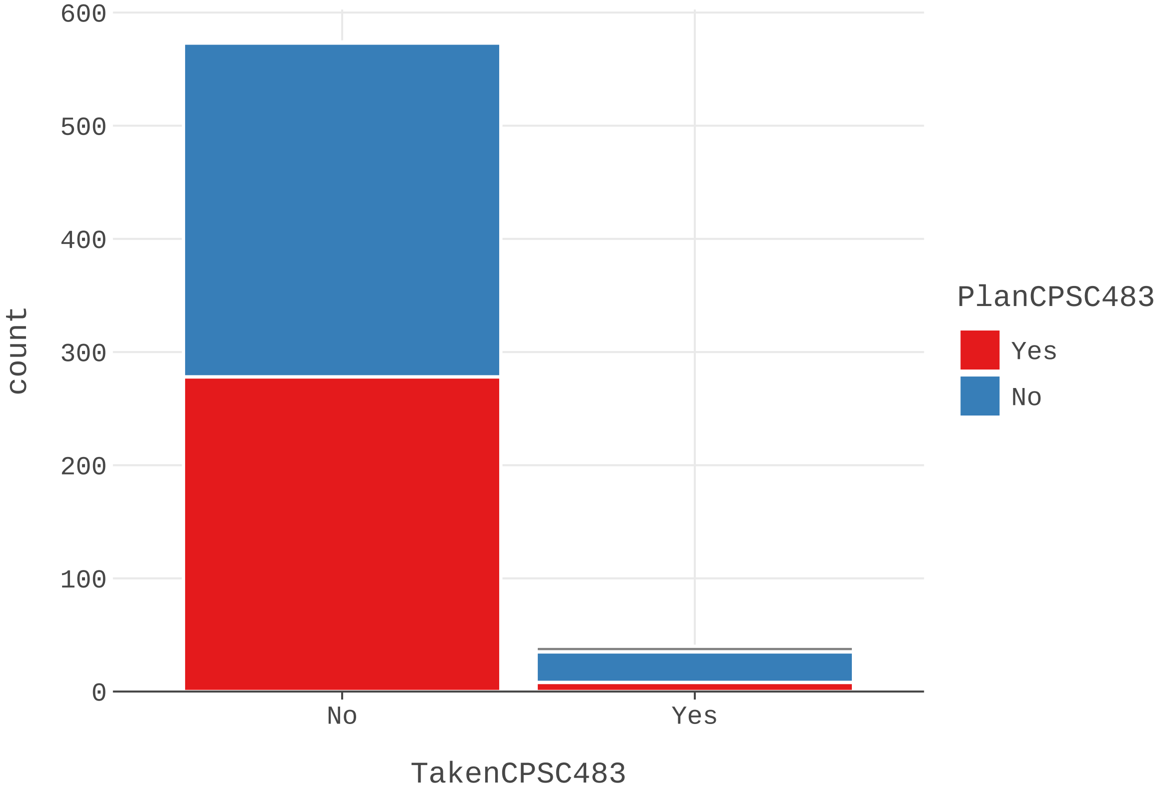
```
# e. Redraw the previous scatterplot but also:
# i. Add more descriptive x and y-axis labels, add a title that should be the
names of all group members, set both x-axis and y-axis limits to (1,10), and make
sure x-axis and y-axis breaks are aligned with the points.
ggplot(survey, aes(x="Math", y="Statistics")) + geom_point() + labs(x="Math",
y="Statistics", title="Math vs. Statistics") + scale_x_continuous(breaks=[1, 2,
3, 4, 5, 6, 7, 8, 9, 10])
```



```
# f. Plot a boxplot of variable ComputerScience vs. Semester
ggplot(survey, aes(x="Semester", y="ComputerScience")) + geom_boxplot()
```



```
# g. Visualize the two categorical variables TakenCPSC483 and PlanCPSC483
ggplot(survey, aes(x="TakenCPSC483", fill="PlanCPSC483")) + geom_bar()
```


count
600
500
400
300
200
100
0
No
Yes
TakenCPSC483
PlanCPSC483
🟥 Yes
🟦 No