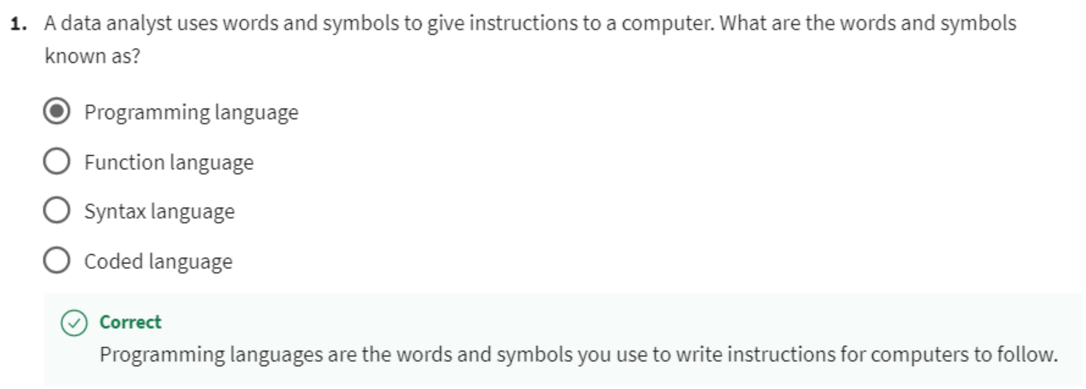
Week 1



1. A data analyst uses words and symbols to give instructions to a computer. What are the words and symbols known as?

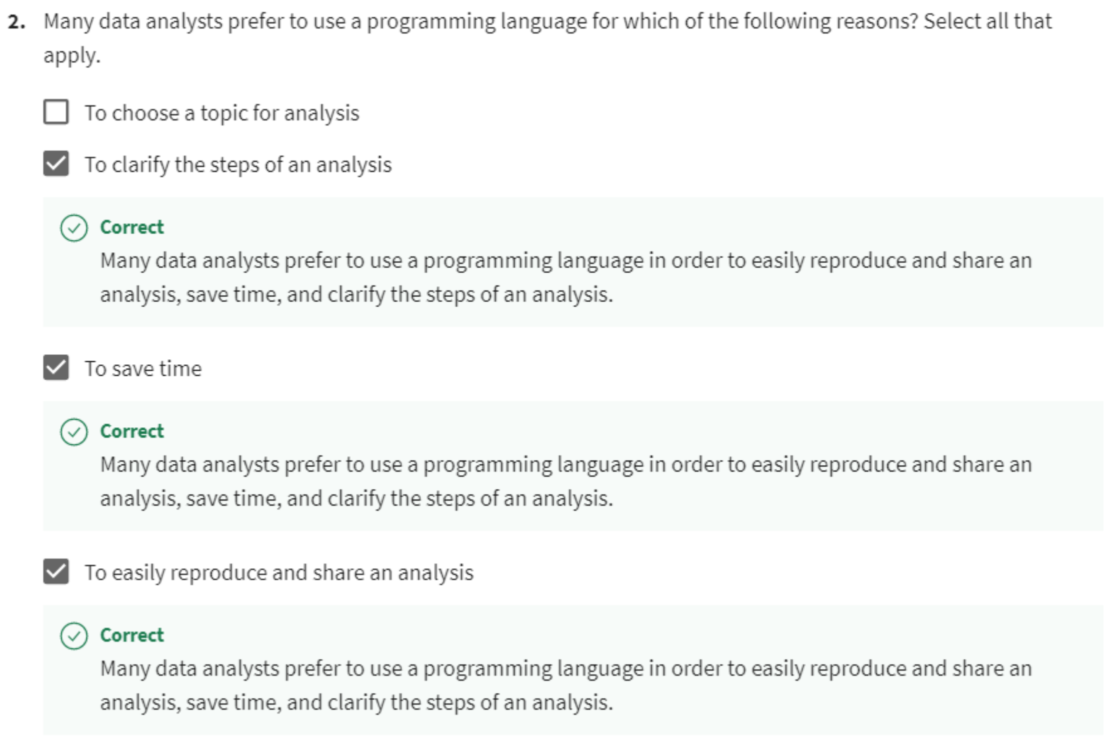
● Programming language

○ Function language

○ Syntax language

○ Coded language

✓ **Correct**
Programming languages are the words and symbols you use to write instructions for computers to follow.



2. Many data analysts prefer to use a programming language for which of the following reasons? Select all that apply.

☐ To choose a topic for analysis

☑ To clarify the steps of an analysis

✓ **Correct**
Many data analysts prefer to use a programming language in order to easily reproduce and share an analysis, save time, and clarify the steps of an analysis.

☑ To save time

✓ **Correct**
Many data analysts prefer to use a programming language in order to easily reproduce and share an analysis, save time, and clarify the steps of an analysis.

☑ To easily reproduce and share an analysis

✓ **Correct**
Many data analysts prefer to use a programming language in order to easily reproduce and share an analysis, save time, and clarify the steps of an analysis.

3. Which of the following are benefits of open-source code? Select all that apply.

☑ Anyone can fix bugs in the code

✓ **Correct**
The benefits of open-source code include the following: anyone can use the code for free, fix bugs in the code, and create add-on packages for the code.

☑ Anyone can use the code for free

✓ **Correct**
The benefits of open-source code include the following: anyone can use the code for free, fix bugs in the code, and create add-on packages for the code.

□ Anyone can pay a fee for access to the code

☑ Anyone can create an add-on package for the code

✓ **Correct**
The benefits of open-source code include the following: anyone can use the code for free, fix bugs in the code, and create add-on packages for the code.

4. For what reasons do many data analysts choose to use R? Select all that apply.

☑ R can quickly process lots of data

✓ **Correct**
Many data analysts choose to use R because it can quickly process lots of data and create high quality visualization. R is also a data-centric programming language, designed to work with data.

□ R is a data-centric programming language

☑ R can create high quality visualizations

✓ **Correct**
Many data analysts choose to use R because it can quickly process lots of data and create high quality visualization. R is also a data-centric programming language, designed to work with data.

□ R is a closed source programming language

You didn't select all the correct answers

5. A team of data analysts is working on a complex analysis. The team needs to quickly process lots of data. They also need to easily reproduce and share every step of their analysis. What should they use for the analysis?

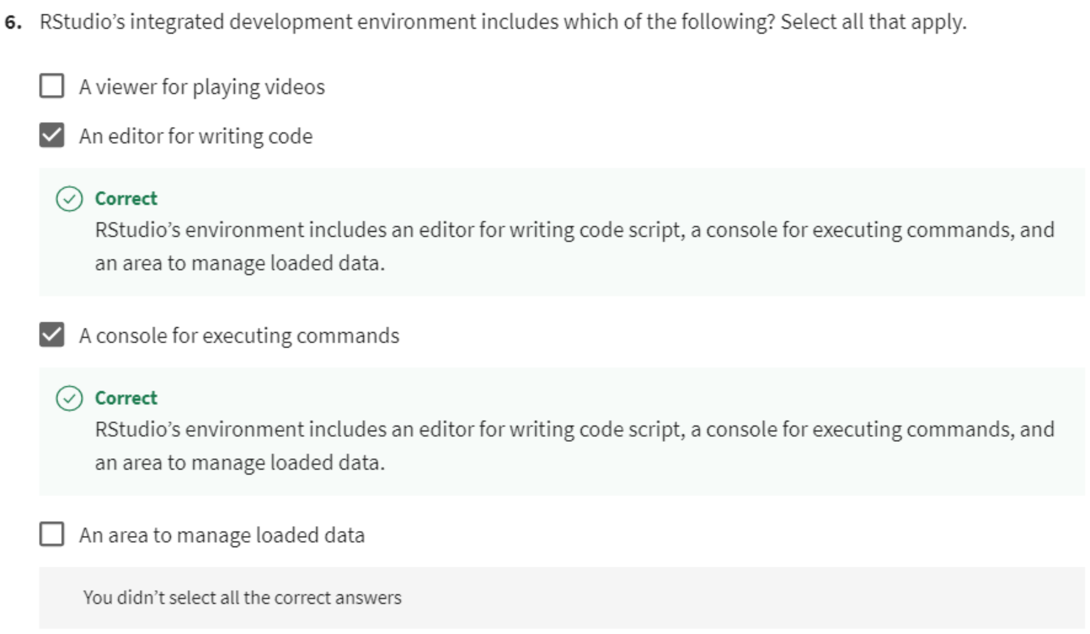
● R programming language

○ A database

○ Structured query language

○ A dashboard

✓ **Correct**
They should use the R programming language. R can quickly process lots of data and reproduce and share every step of an analysis.



6. RStudio's integrated development environment includes which of the following? Select all that apply.

☐ A viewer for playing videos

☑ An editor for writing code

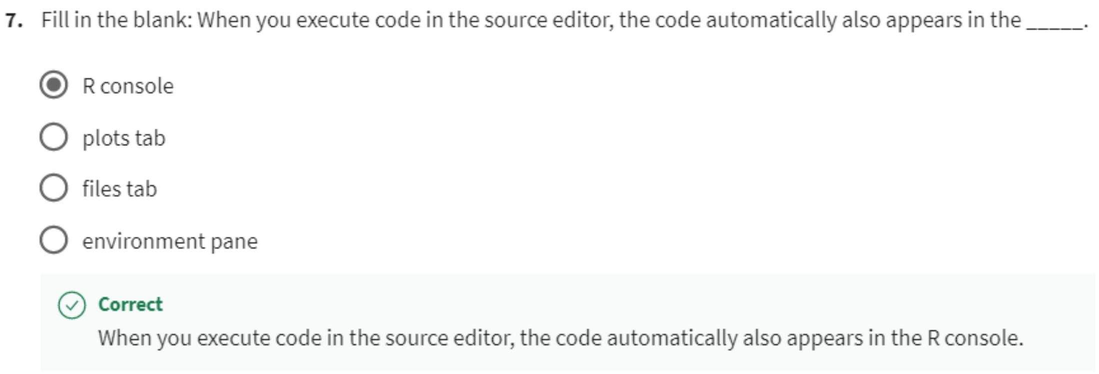
✓ **Correct**
RStudio's environment includes an editor for writing code script, a console for executing commands, and an area to manage loaded data.

☑ A console for executing commands

✓ **Correct**
RStudio's environment includes an editor for writing code script, a console for executing commands, and an area to manage loaded data.

☐ An area to manage loaded data

You didn't select all the correct answers



7. Fill in the blank: When you execute code in the source editor, the code automatically also appears in the _____.

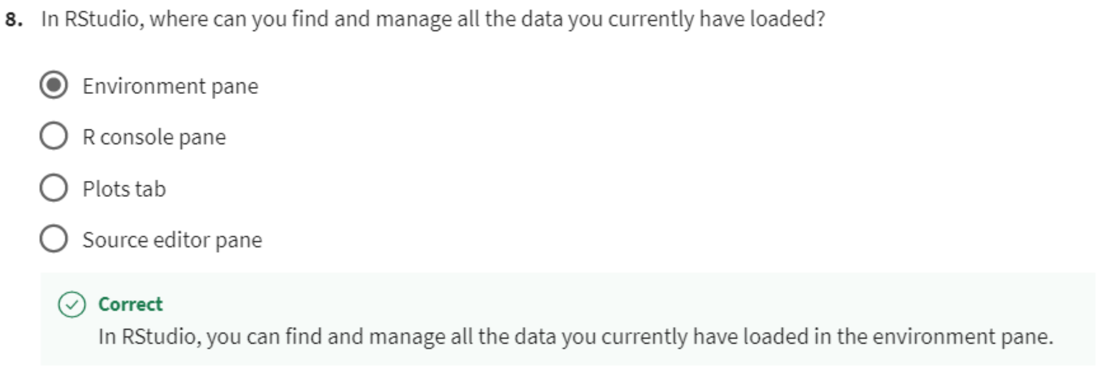
● R console

○ plots tab

○ files tab

○ environment pane

✓ **Correct**
When you execute code in the source editor, the code automatically also appears in the R console.



8. In RStudio, where can you find and manage all the data you currently have loaded?

● Environment pane

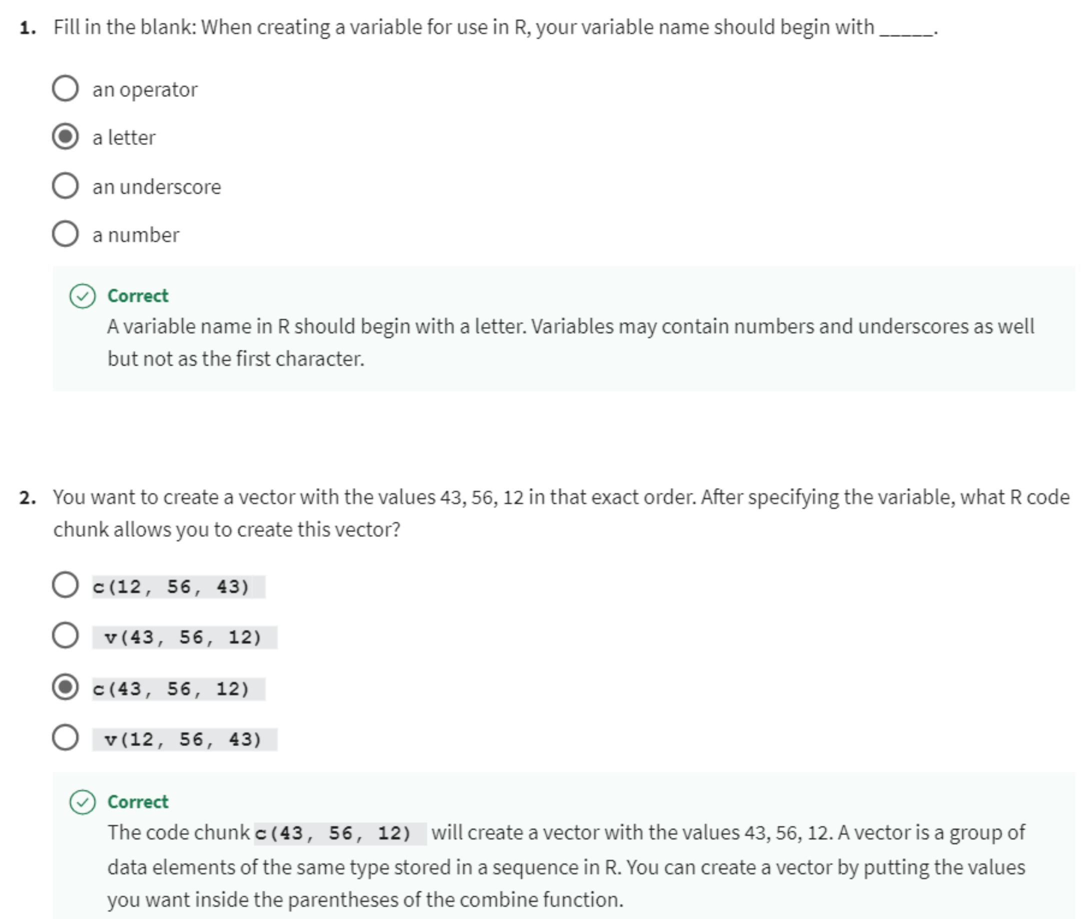
○ R console pane

○ Plots tab

○ Source editor pane

✓ **Correct**
In RStudio, you can find and manage all the data you currently have loaded in the environment pane.

Week 2



1. Fill in the blank: When creating a variable for use in R, your variable name should begin with _____.

○ an operator

● a letter

○ an underscore

○ a number

Correct
A variable name in R should begin with a letter. Variables may contain numbers and underscores as well but not as the first character.

2. You want to create a vector with the values 43, 56, 12 in that exact order. After specifying the variable, what R code chunk allows you to create this vector?

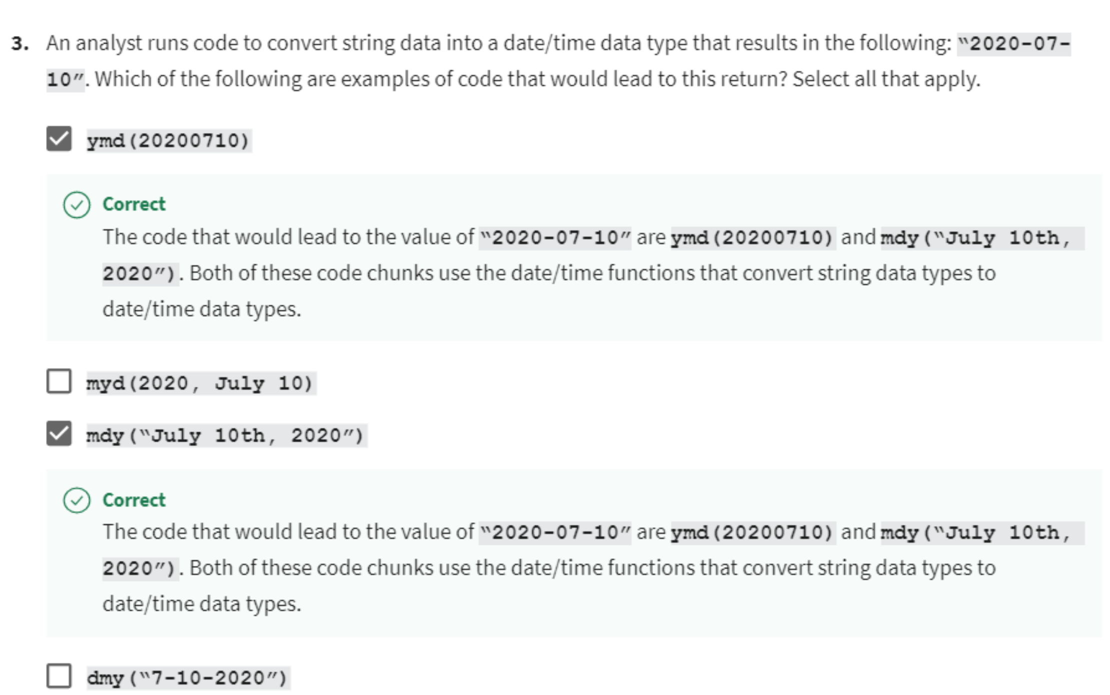
○ `c(12, 56, 43)`

○ `v(43, 56, 12)`

● `c(43, 56, 12)`

○ `v(12, 56, 43)`

Correct
The code chunk `c(43, 56, 12)` will create a vector with the values 43, 56, 12. A vector is a group of data elements of the same type stored in a sequence in R. You can create a vector by putting the values you want inside the parentheses of the combine function.



3. An analyst runs code to convert string data into a date/time data type that results in the following: `“2020-07-10”`. Which of the following are examples of code that would lead to this return? Select all that apply.

☑ `ymd(20200710)`

Correct
The code that would lead to the value of `“2020-07-10”` are `ymd(20200710)` and `mdy(“July 10th, 2020”)`. Both of these code chunks use the date/time functions that convert string data types to date/time data types.

□ `myd(2020, July 10)`

☑ `mdy(“July 10th, 2020”)`

Correct
The code that would lead to the value of `“2020-07-10”` are `ymd(20200710)` and `mdy(“July 10th, 2020”)`. Both of these code chunks use the date/time functions that convert string data types to date/time data types.

□ `dmy(“7-10-2020”)`

4. A data analyst inputs the following code in RStudio:

```
change_1 <- 70
```

Which of the following types of operators does the analyst use in the code?

○ Relational

○ Logical

● Assignment

○ Arithmetic

Correct

In this code, the analyst uses an assignment operator: <-. The assignment operator assigns the value 70 to the variable `change_1`.

5. Which of the following variables have names that follow widely accepted naming convention rules? Select all that apply.

☑ `plum_total_1`

Correct

The variables with a name that follows widely accepted naming convention rules are `total_plums` and `plum_total_1`. These variable names use only lowercase letters and underscores and are clear, concise, and meaningful.

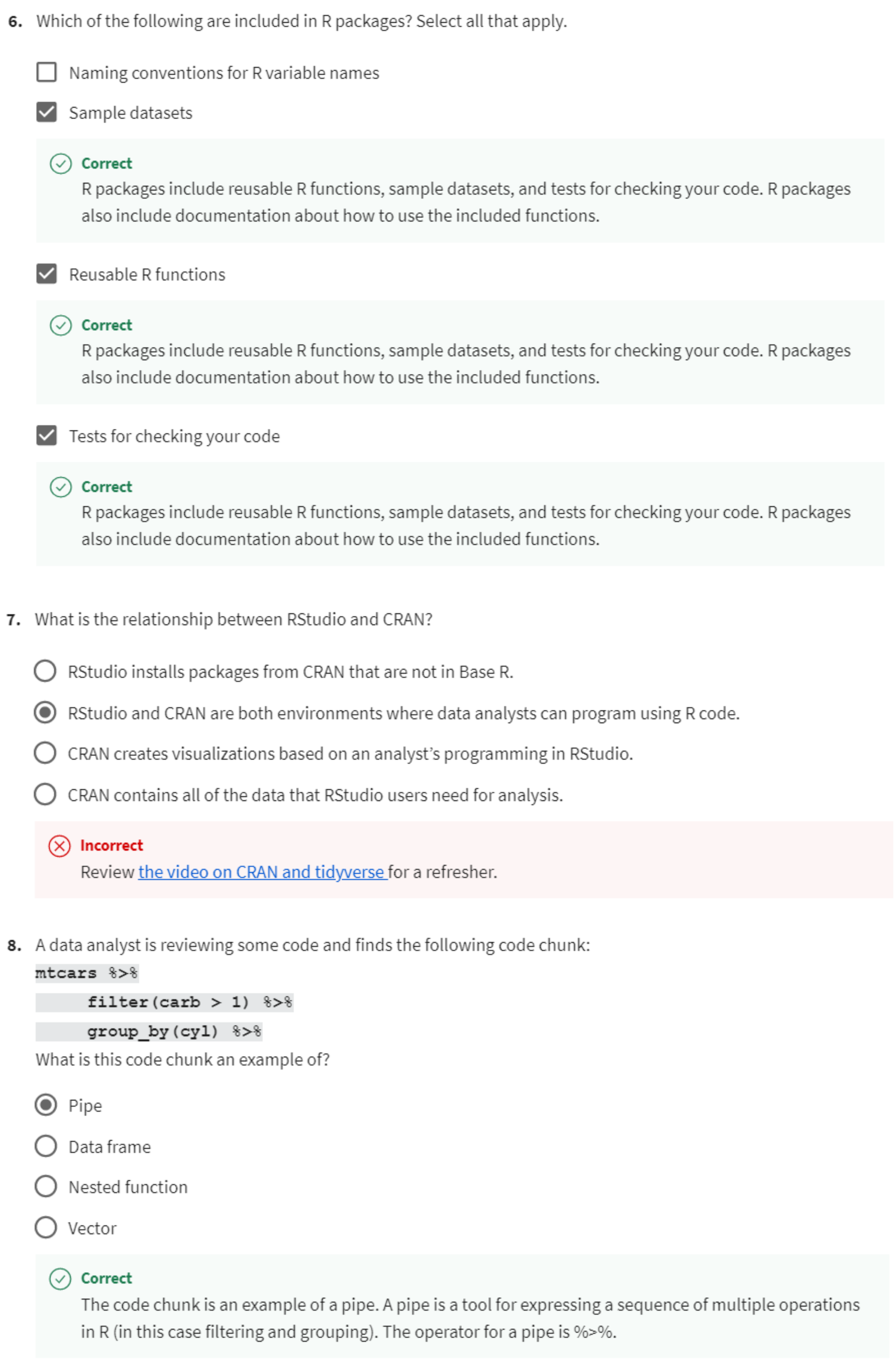
☑ `total_plums`

Correct

The variables with a name that follows widely accepted naming convention rules are `total_plums` and `plum_total_1`. These variable names use only lowercase letters and underscores and are clear, concise, and meaningful.

□ `*totalplums*`

□ `1_plum_total`



6. Which of the following are included in R packages? Select all that apply.

□ Naming conventions for R variable names

☑ Sample datasets

✓ **Correct**
R packages include reusable R functions, sample datasets, and tests for checking your code. R packages also include documentation about how to use the included functions.

☑ Reusable R functions

✓ **Correct**
R packages include reusable R functions, sample datasets, and tests for checking your code. R packages also include documentation about how to use the included functions.

☑ Tests for checking your code

✓ **Correct**
R packages include reusable R functions, sample datasets, and tests for checking your code. R packages also include documentation about how to use the included functions.

7. What is the relationship between RStudio and CRAN?

○ RStudio installs packages from CRAN that are not in Base R.

● RStudio and CRAN are both environments where data analysts can program using R code.

○ CRAN creates visualizations based on an analyst's programming in RStudio.

○ CRAN contains all of the data that RStudio users need for analysis.

☒ **Incorrect**
Review the video on CRAN and tidyverse for a refresher.

8. A data analyst is reviewing some code and finds the following code chunk:

```
mtcars %>%
    filter(carb > 1) %>%
    group_by(cyl) %>%
```

What is this code chunk an example of?

● Pipe

○ Data frame

○ Nested function

○ Vector

✓ **Correct**
The code chunk is an example of a pipe. A pipe is a tool for expressing a sequence of multiple operations in R (in this case filtering and grouping). The operator for a pipe is %>%.

Week 3

1.

A data analyst is working with a dataset in R that has more than 50,000 observations. Why might they choose to use a tibble instead of the standard data frame? Select all that apply.

☑ Tibbles automatically only preview the first 10 rows of data

✓ **Correct**
Tibbles make printing in R easier. They won’t accidentally overload the data analyst’s console because they’re automatically set to pull up only the first 10 rows and as many columns as fit on screen.

☑ Tibbles automatically only preview as many columns as fit on screen

✓ **Correct**
Tibbles make printing in R easier. They won’t accidentally overload the data analyst’s console because they’re automatically set to pull up only the first 10 rows and as many columns as fit on screen.

□ Tibbles can create row names

□ Tibbles can automatically change the names of variables

2. A data analyst is exploring their data to get more familiar with it. They want a preview of just the first six rows to get a better idea of how the data frame is laid out. What function should they use?

○ colnames()

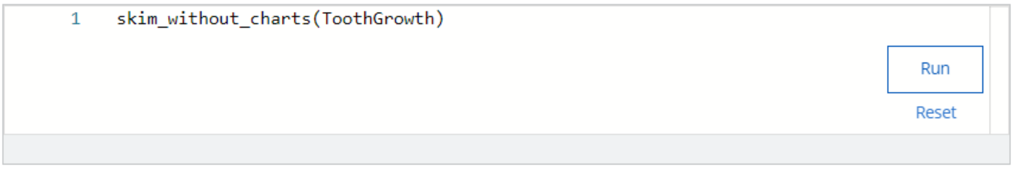
○ print()

● head()

○ preview()

✓ **Correct**
The head() function can be used to return a preview of the first six rows of a data frame. This is a useful way to explore a data frame and get more familiar with how it is structured.

3. You are working with the ToothGrowth dataset. You want to use the skim_without_charts() function to get a comprehensive view of the dataset. Write the code chunk that will give you this view.



```
skim_without_charts(ToothGrowth)
```

How many rows does the ToothGrowth dataset contain?

○ 50

● 60

○ 40

○ 25

✓ **Correct**

The code chunk `skim_without_charts(ToothGrowth)` gives you a comprehensive view of the dataset. Inside the parentheses of the skim_without_charts() function is the name of the dataset you want to view. The code returns a summary with the name of the dataset and the number of rows and columns. It also shows the column types and data types contained in the dataset. The ToothGrowth dataset contains 60 rows.

4. A data analyst is working with the penguins dataset. What code chunk does the analyst write to make sure all the column names are unique and consistent and contain only letters, numbers, and underscores?

○ `drop_na(penguins)`

○ `select(penguins)`

○ `rename(penguins)`

● `clean_names(penguins)`

✓ **Correct**

The code chunk is `clean_names(penguins)`. The clean_names() function ensures that there are only characters, numbers, and underscores in the names used in the data frame.

5. A data analyst is working with the penguins dataset in R. What code chunk will allow them to sort the penguins data by the variable *bill_length_mm*?

○ `arrange(bill_length_mm, penguins)`

● `arrange(penguins, bill_length_mm)`

○ `arrange(=bill_length_mm)`

○ `arrange(penguins)`

✓ **Correct**

The code chunk is `arrange(penguins, bill_length_mm)`. The arrange function allows the analyst to sort data in their dataset. The arguments for the function identify the dataset as the penguins data, and that the sort should be based on the *bill_length_mm* variable. The data is automatically sorted in ascending order.

6. You are working with the penguins dataset. You want to use the summarize() and max() functions to find the maximum value for the variable *flipper_length_mm*. You write the following code:

```
penguins %>%

  drop_na() %>%

  group_by(species) %>%
```

Add the code chunk that lets you find the maximum value for the variable *flipper_length_mm.*

```
summarize(max(flipper_length_mm))
```

Run

Reset

What is the maximum flipper length in mm for the Gentoo species?

○ 200

○ 212

● 231

○ 210

✓ **Correct**

The code chunk `summarize(max(flipper_length_mm))` lets you find the maximum value for the variable flipper_length_mm. The correct code is `penguins %>% drop_na() %>% group_by(species) %>% summarize(max(flipper_length_mm))`. The summarize() function displays summary statistics. You can use the summarize() function in combination with other functions -- such as mean(), max(), and min() -- to calculate specific statistics. In this case, you use max() to calculate the maximum value for flipper length. The maximum flipper length for the Gentoo species is 231mm.

7. A data analyst is working with a data frame called *salary_data*. They want to create a new column named *hourly_salary* that includes data from the *wages* column divided by 40. What code chunk lets the analyst create the *hourly_salary* column?

○ `mutate(hourly_salary = wages / 40)`

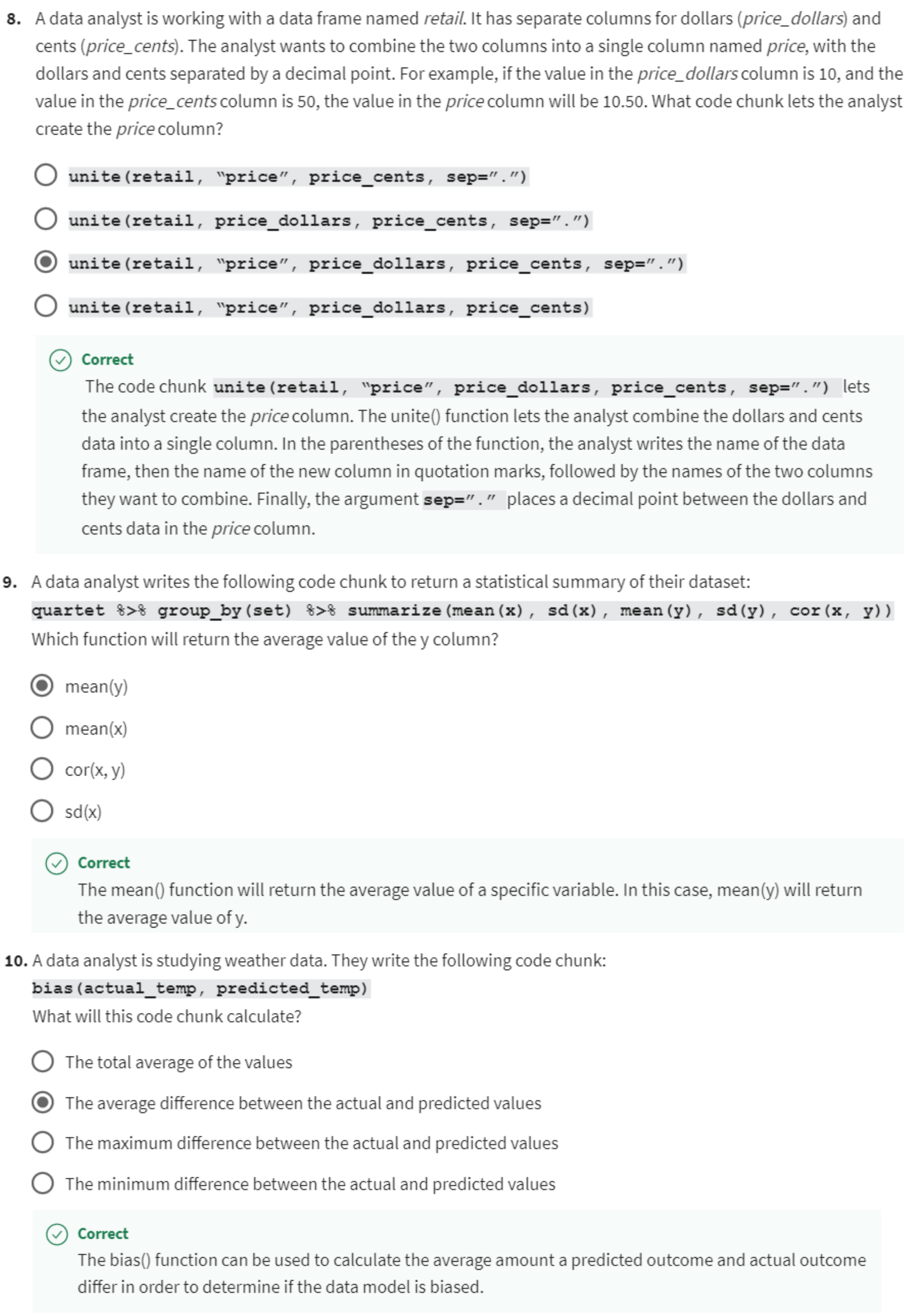
○ `mutate(hourly_salary, salary_data = wages / 40)`

● `mutate(salary_data, hourly_salary = wages / 40)`

○ `mutate(salary_data, hourly_salary = wages * 40)`

✓ **Correct**

The code chunk is `mutate(salary_data, hourly_salary = wages / 40)`. The analyst can use the mutate() function to create a new column for wages divided by 40 called hourly_salary. The mutate() function can create a new column without affecting any existing columns.



8. A data analyst is working with a data frame named *retail*. It has separate columns for dollars (*price_dollars*) and cents (*price_cents*). The analyst wants to combine the two columns into a single column named *price*, with the dollars and cents separated by a decimal point. For example, if the value in the *price_dollars* column is 10, and the value in the *price_cents* column is 50, the value in the *price* column will be 10.50. What code chunk lets the analyst create the *price* column?

○ `unite(retail, “price”, price_cents, sep=”.”)`

○ `unite(retail, price_dollars, price_cents, sep=”.”)`

● `unite(retail, “price”, price_dollars, price_cents, sep=”.”)`

○ `unite(retail, “price”, price_dollars, price_cents)`

✓ **Correct**

The code chunk `unite(retail, “price”, price_dollars, price_cents, sep=”.”)` lets the analyst create the *price* column. The unite() function lets the analyst combine the dollars and cents data into a single column. In the parentheses of the function, the analyst writes the name of the data frame, then the name of the new column in quotation marks, followed by the names of the two columns they want to combine. Finally, the argument `sep=”.”` places a decimal point between the dollars and cents data in the *price* column.

9. A data analyst writes the following code chunk to return a statistical summary of their dataset:

`quartet %>% group_by(set) %>% summarize(mean(x), sd(x), mean(y), sd(y), cor(x, y))`

Which function will return the average value of the y column?

● mean(y)

○ mean(x)

○ cor(x, y)

○ sd(x)

✓ **Correct**

The mean() function will return the average value of a specific variable. In this case, mean(y) will return the average value of y.

10. A data analyst is studying weather data. They write the following code chunk:

`bias(actual_temp, predicted_temp)`

What will this code chunk calculate?

○ The total average of the values

● The average difference between the actual and predicted values

○ The maximum difference between the actual and predicted values

○ The minimum difference between the actual and predicted values

✓ **Correct**

The bias() function can be used to calculate the average amount a predicted outcome and actual outcome differ in order to determine if the data model is biased.

Week 4

1. Which of the following are benefits of using ggplot2? Select all that apply.

☑ Easily add layers to your plot

> **Correct**
> The benefits of using ggplot2 include easily adding layers to your plot, customizing the look and feel of your plot, combining data manipulation and visualization.

☑ Combine data manipulation and visualization

> **Correct**
> The benefits of using ggplot2 include easily adding layers to your plot, customizing the look and feel of your plot, combining data manipulation and visualization.

☑ Customize the look and feel of your plot

> **Correct**
> The benefits of using ggplot2 include easily adding layers to your plot, customizing the look and feel of your plot, combining data manipulation and visualization.

□ Automatically clean data before creating a plot

2. In ggplot2, what symbol do you use to add layers to your plot?

● The plus sign (+)

○ The equal sign (=)

○ The pipe operator (%>%)

○ The ampersand symbol (&)

> **Correct**
> In ggplot2, you use the plus sign (+) to add layers to your plot.

3. A data analyst creates a plot using the following code chunk:

```
ggplot(data = penguins) +
    geom_point(mapping = aes(x = flipper_length_mm, y = body_mass_g))
```

Which of the following represents an aesthetic attribute in the code chunk? Select all that apply.

□ `x`

☑ `body_mass_g`

> **This should not be selected**
> Review the video on creating a plot for a refresher.

☑ `flipper_length_mm`

> **This should not be selected**
> Review the video on creating a plot for a refresher.

□ `y`

4. In ggplot2, which of the following aesthetic attributes can you use to map variables to points? Select all that apply.

☑ Color

✓ **Correct**
In ggplot2, color, shape, and size are aesthetic attributes you can use to map variables to points. Color refers to the color of the points on your plot, shape to the shape of the points, and size to the size of the points.

☑ Shape

✓ **Correct**
In ggplot2, color, shape, and size are aesthetic attributes you can use to map variables to points. Color refers to the color of the points on your plot, shape to the shape of the points, and size to the size of the points.

□ Facet

☑ Size

✓ **Correct**
In ggplot2, color, shape, and size are aesthetic attributes you can use to map variables to points. Color refers to the color of the points on your plot, shape to the shape of the points, and size to the size of the points.

5. The _____ aesthetic makes some points on a plot more transparent, or see-through, than others.

● alpha

○ fill

○ linetype

○ color

✓ **Correct**
The alpha aesthetic makes some points on a plot more transparent, or see-through, than others.

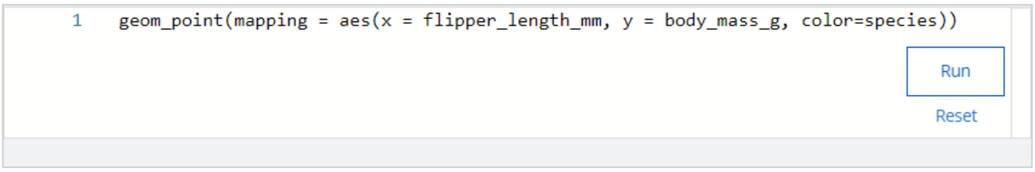
6. You are working with the penguins dataset. You create a scatterplot with the following code:

```
ggplot(data = penguins) +

  geom_point(mapping = aes(x = flipper_length_mm, y = body_mass_g))
```

You want to highlight the different penguin species on your plot. Add a code chunk to the second line of code to map the aesthetic *shape* to the variable *species.*

NOTE: the three dots (...) indicate where to add the code chunk.



```
geom_point(mapping = aes(x = flipper_length_mm, y = body_mass_g, color=species))
```

Which penguin species does your visualization display?

● Adelie, Chinstrap, Gentoo

○ Emperor, Chinstrap, Gentoo

○ Adelie, Chinstrap, Emperor

○ Adelie, Gentoo, Macaroni

✓ **Correct**

You add the code chunk `shape = species` to the second line of code to map the aesthetic shape to the variable species. The correct code is `ggplot(data = penguins) + geom_point(mapping = aes(x = flipper_length_mm, y = body_mass_g, shape = species))`. Inside the parentheses of the aes() function, after the comma that follows y = body_mass_g, write the aesthetic (shape), then an equals sign, then the variable (species). The data points for each penguin species now appear in different shapes.

Your visualization displays the Adelie, Chinstrap, and Gentoo penguin species.

7. Fill in the blank: The _____ creates a scatterplot and then adds a small amount of random noise to each point in the plot to make the points easier to find.

○ geom_smooth() function

○ geom_bar() function

● geom_jitter() function

○ geom_point() function

✓ **Correct**

The geom_jitter() function creates a scatterplot and then adds a small amount of random noise to each point in the plot to make the points easier to find.

8. You are working with the diamonds dataset. You create a bar chart with the following code:

```
ggplot(data = diamonds) +

  geom_bar(mapping = aes(x = color, fill = cut)) +
```

You want to use the facet_wrap() function to display subsets of your data. Add the code chunk that lets you facet your plot based on the variable *clarity.*

```
facet_wrap(~clarity)
```

Run

Reset

How many subplots does your visualization show?

○ 9

○ 7

○ 6

● 8

Correct

You add the code chunk `facet_wrap(~clarity)` to facet your plot based on the variable clarity. The correct code is `ggplot(data = diamonds) + geom_bar(mapping = aes(x = color, fill = cut)) + facet_wrap(~clarity)`. Inside the parentheses of the facet_wrap() function, write a tilde symbol (~) followed by the name of the variable you want to facet. The facet_wrap() function lets you display subsets of your data.

Your visualization shows 8 subplots.

9. A data analyst creates a scatterplot. The analyst wants to put a text label on the plot to call out specific data points. What function does the analyst use?

○ The ggplot() function

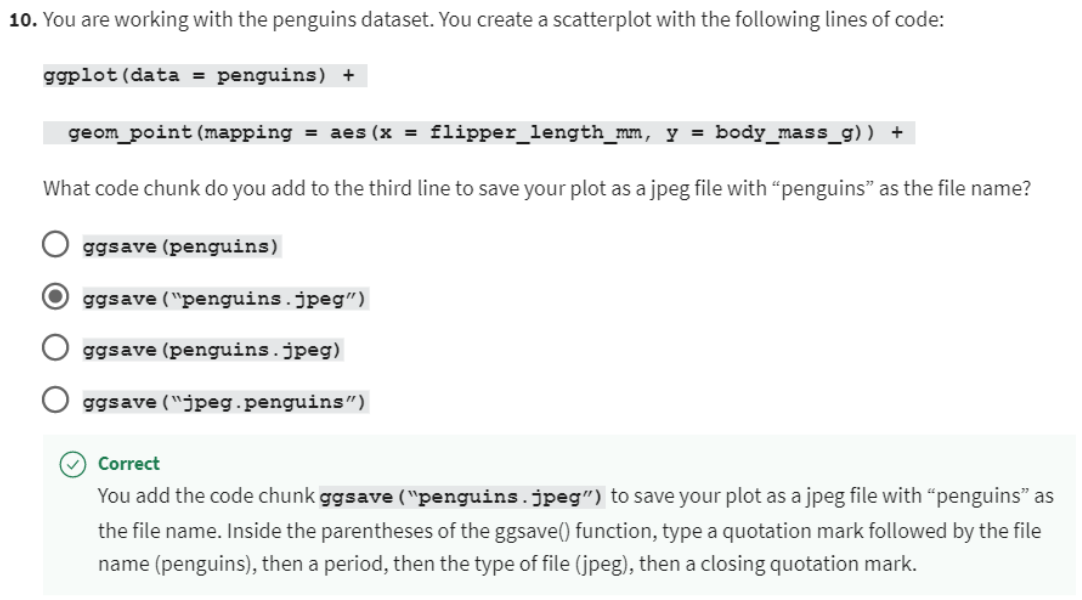
● The annotate() function

○ The facet_grid() function

○ The geom_smooth() function

Correct

The analyst uses the annotate() function. The annotate() function can put a text label on a plot to call out specific data points.



10. You are working with the penguins dataset. You create a scatterplot with the following lines of code:

```
ggplot(data = penguins) +

  geom_point(mapping = aes(x = flipper_length_mm, y = body_mass_g)) +
```

What code chunk do you add to the third line to save your plot as a jpeg file with “penguins” as the file name?

○ `ggsave(penguins)`

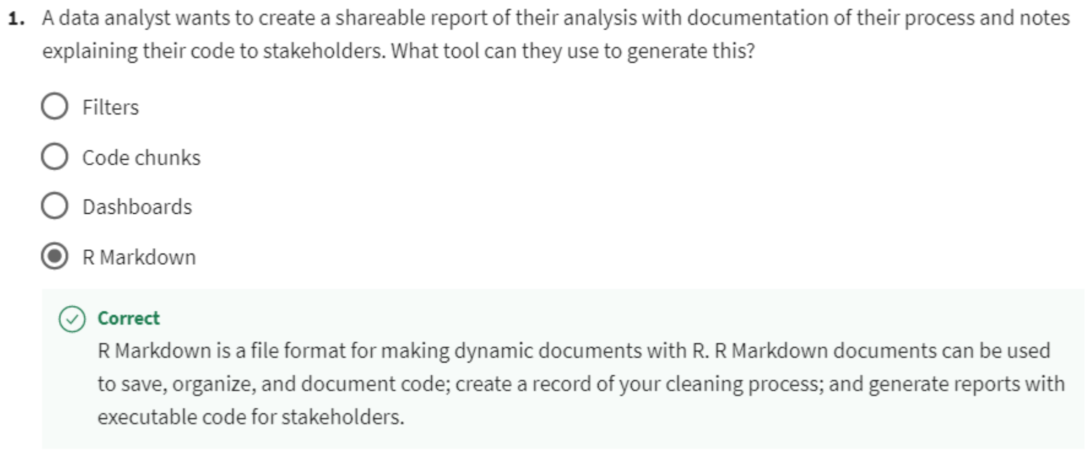
● `ggsave(“penguins.jpeg”)`

○ `ggsave(penguins.jpeg)`

○ `ggsave(“jpeg.penguins”)`

✓ **Correct**
You add the code chunk **`ggsave(“penguins.jpeg”)`** to save your plot as a jpeg file with “penguins” as the file name. Inside the parentheses of the ggsave() function, type a quotation mark followed by the file name (penguins), then a period, then the type of file (jpeg), then a closing quotation mark.

Week 5



1. A data analyst wants to create a shareable report of their analysis with documentation of their process and notes explaining their code to stakeholders. What tool can they use to generate this?

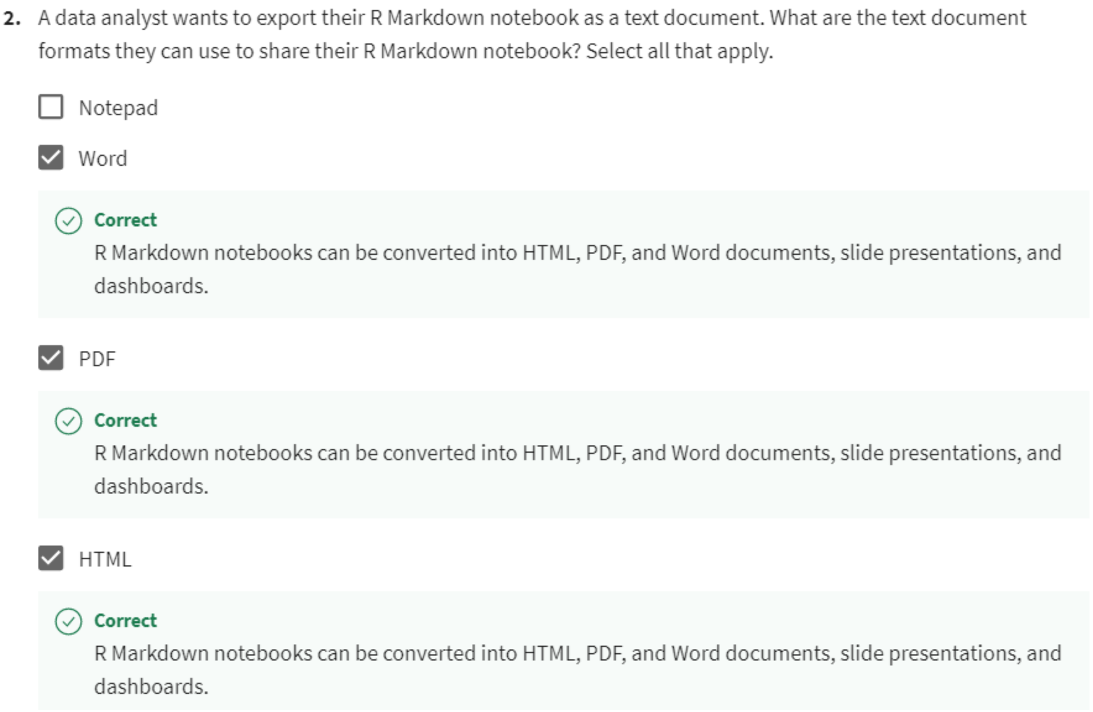
○ Filters

○ Code chunks

○ Dashboards

● R Markdown

✓ **Correct**
R Markdown is a file format for making dynamic documents with R. R Markdown documents can be used to save, organize, and document code; create a record of your cleaning process; and generate reports with executable code for stakeholders.



2. A data analyst wants to export their R Markdown notebook as a text document. What are the text document formats they can use to share their R Markdown notebook? Select all that apply.

☐ Notepad

☑ Word

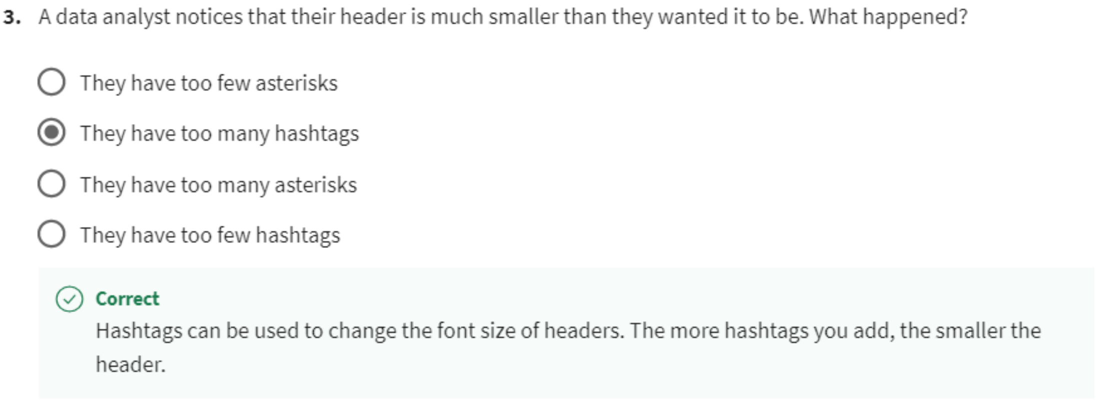
✓ **Correct**
R Markdown notebooks can be converted into HTML, PDF, and Word documents, slide presentations, and dashboards.

☑ PDF

✓ **Correct**
R Markdown notebooks can be converted into HTML, PDF, and Word documents, slide presentations, and dashboards.

☑ HTML

✓ **Correct**
R Markdown notebooks can be converted into HTML, PDF, and Word documents, slide presentations, and dashboards.



3. A data analyst notices that their header is much smaller than they wanted it to be. What happened?

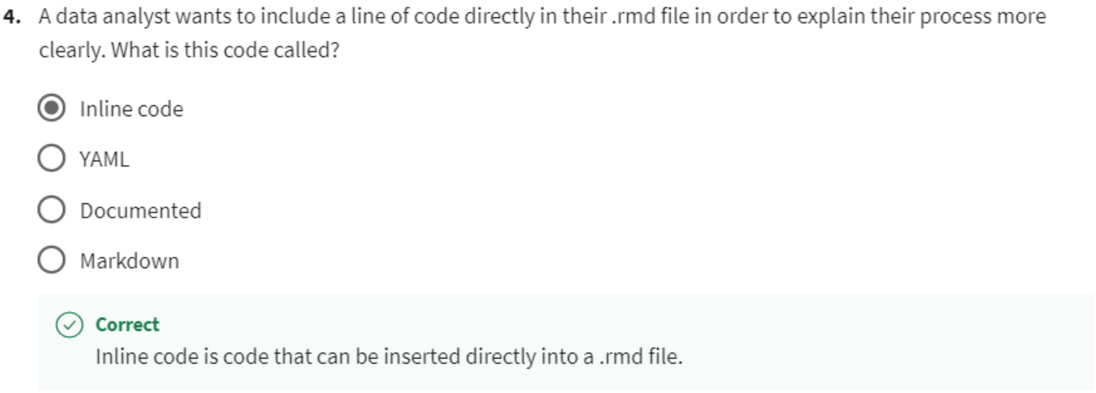
○ They have too few asterisks

● They have too many hashtags

○ They have too many asterisks

○ They have too few hashtags

✓ **Correct**
Hashtags can be used to change the font size of headers. The more hashtags you add, the smaller the header.



4. A data analyst wants to include a line of code directly in their .rmd file in order to explain their process more clearly. What is this code called?

● Inline code

○ YAML

○ Documented

○ Markdown

✓ **Correct**
Inline code is code that can be inserted directly into a .rmd file.

5. A data analyst inputs asterisks before a word or phrase in R Markdown. How will this appear in the document?

● As bullet points

○ As a numerical list

○ As inline code

○ As a code chunk

✓ **Correct**
Asterisks can be used to add bullet points to an .rmd file.

6. A data analyst includes a code chunk in their .rmd file. What does this allow other users to do? Select all that apply.

□ Undo original project code directly from the .rdm file

☑ Copy code directly from the .rmd file

✓ **Correct**
Code added to an .rmd file is usually referred to as a code chunk. Code chunks allow users to execute, modify, and copy R code from within the .rmd file.

☑ Execute code directly from the .rmd file

✓ **Correct**
Code added to an .rmd file is usually referred to as a code chunk. Code chunks allow users to execute, modify, and copy R code from within the .rmd file.

☑ Modify code directly from the .rmd file

✓ **Correct**
Code added to an .rmd file is usually referred to as a code chunk. Code chunks allow users to execute, modify, and copy R code from within the .rmd file.

7. Fill in the blank: A delimiter is a character that indicates the beginning or end of _____.

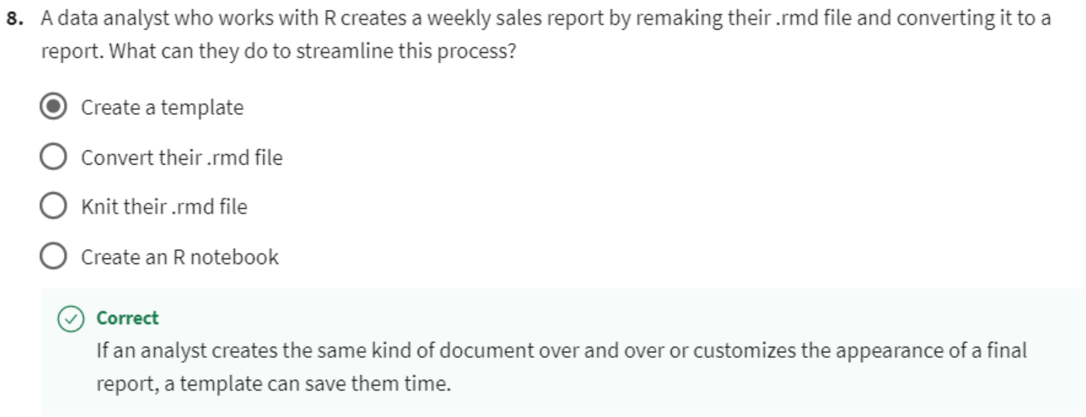
○ a header

● a data item

○ a section

○ an analysis

✓ **Correct**
A delimiter is a character that indicates the beginning or end of a data item in a code chunk.



8. A data analyst who works with R creates a weekly sales report by remaking their .rmd file and converting it to a report. What can they do to streamline this process?

● Create a template

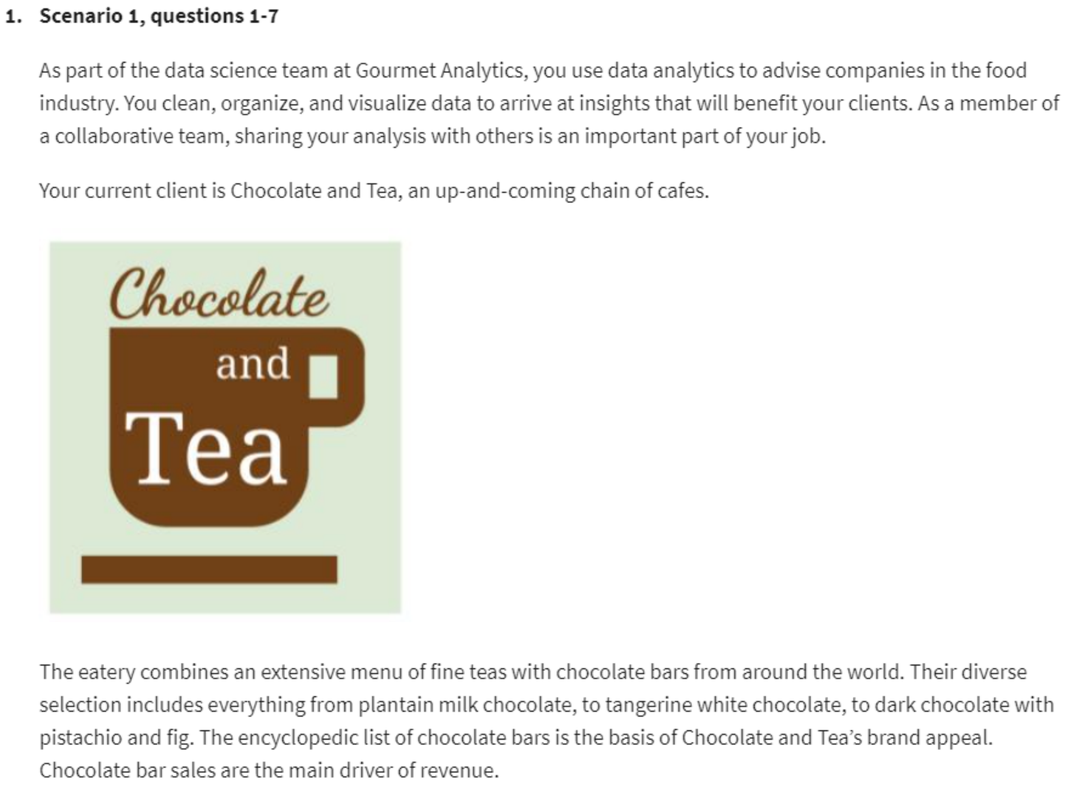
○ Convert their .rmd file

○ Knit their .rmd file

○ Create an R notebook

✓ **Correct**
If an analyst creates the same kind of document over and over or customizes the appearance of a final report, a template can save them time.

Course Challenge



1. Scenario 1, questions 1-7

As part of the data science team at Gourmet Analytics, you use data analytics to advise companies in the food industry. You clean, organize, and visualize data to arrive at insights that will benefit your clients. As a member of a collaborative team, sharing your analysis with others is an important part of your job.

Your current client is Chocolate and Tea, an up-and-coming chain of cafes.

The eatery combines an extensive menu of fine teas with chocolate bars from around the world. Their diverse selection includes everything from plantain milk chocolate, to tangerine white chocolate, to dark chocolate with pistachio and fig. The encyclopedic list of chocolate bars is the basis of Chocolate and Tea's brand appeal. Chocolate bar sales are the main driver of revenue.

Chocolate and Tea aims to serve chocolate bars that are highly rated by professional critics. They also continually adjust the menu to make sure it reflects the global diversity of chocolate production. The management team regularly updates the chocolate bar list in order to align with the latest ratings and to ensure that the list contains bars from a variety of countries.

They've asked you to collect and analyze data on the latest chocolate ratings. In particular, they'd like to know which countries produce the highest-rated bars of super dark chocolate (a high percentage of cocoa). This data will help them create their next chocolate bar menu.

Your team has received a dataset that features the latest ratings for thousands of chocolates from around the world. Click here to access the dataset. Given the data and the nature of the work you will do for your client, your team agrees to use R for this project.

A teammate asks you about the benefits of using R for the project. You mention that R can quickly process lots of data and create high quality data visualizations. What is another benefit of using R for the project?

○ Choose a topic for analysis

● Easily reproduce and share an analysis

○ Automatically clean data

○ Define a problem and ask the right questions

✓ **Correct**
Another benefit of using R for the project is that it can easily reproduce and share an analysis.

2. **Scenario 1, continued**

Before you begin working with your data, you need to import it and save it as a data frame. To get started, you open your RStudio workspace and load the tidyverse library. You upload a .csv file containing the data to RStudio and store it in a project folder named flavors_of_cacao.csv.

You use the read_csv() function to import the data from the .csv file. Assume that the name of the data frame is bars_df and the .csv file is in the working directory. What code chunk lets you create the data frame?

● `bars_df <- read_csv("flavors_of_cacao.csv")`

○ `bars_df + read_csv("flavors_of_cacao.csv")`

○ `read_csv("flavors_of_cacao.csv") + bars_df`

○ `bars_df %>% read_csv("flavors_of_cacao.csv")`

✓ **Correct**
The code chunk `bars_df <- read_csv("flavors_of_cacao.csv")` lets you create the data frame. In this code chunk:

- bars_df is the name of the data frame that will store the data.
- <- is the assignment operator to assign values to the data frame.
- read_csv() is the function that will import the data to the data frame.
- "flavors_of_cacao.csv" is the file name that read.csv() function takes for its argument.

3. **Scenario 1, continued**

Now that you've created a data frame, you want to find out more about how the data is organized. The data frame has hundreds of rows and lots of columns.

Assume the name of your data frame is flavors_df. What code chunk lets you get a glimpse of the contents of the data frame?

○ `glimpse <- flavors_df`

○ `glimpse = flavors_df`

○ `glimpse %>% flavors_df`

● `glimpse(flavors_df)`

✓ Correct

You write the code chunk `glimpse(flavors_df)`. In this code chunk:

- `glimpse()` is the function that will give you a glimpse of the contents of the data frame, and give you high-level information like column names and the type of data contained in those columns.
- `flavors_df` is the name of the data frame that the glimpse() function takes for its argument.

4. **Scenario 1, continued**

Next, you begin to clean your data. When you check out the column headings in your data frame you notice that the first column is named *Company...Maker.if.known.* (Note: The period after *known* is part of the variable name.) For the sake of clarity and consistency, you decide to rename this column *Company* (without a period at the end).

Assume the first part of your code chunk is:

`flavors_df %>%`

What code chunk do you add to change the column name?

○ `rename(Company...Maker.if.known. <- Company)`

○ `rename(Company <- Company...Maker.if.known.)`

● `rename(Company = Company...Maker.if.known.)`

○ `rename(Company...Maker.if.known. = Company)`

✓ Correct

You write the code chunk `rename(Company = Company...Maker.if.known.)`.

In this code chunk:

- `rename()` is the function that will change the name of your column.
- Inside the parentheses of the function, write the new name (`Company`), then an equals sign, then the name you want to change (`Company...Maker.if.known.`).

5. After previewing and cleaning your data, you determine what variables are most relevant to your analysis. Your main focus is on *Rating*, *Cocoa.Percent*, and *Company.Location*. You decide to use the select() function to create a new data frame with only these three variables.

   Assume the first part of your code is:

   `trimmed_flavors_df <- flavors_df %>%`

   Add the code chunk that lets you select the three variables.

```
select(Rating, Cocoa.Percent, Company.Location)
```

Run

Reset

What company location appears in row 1 of your tibble?

○ Canada

○ Colombia

○ Scotland

● France

✓ **Correct**

You add the code chunk `select(Rating, Cocoa.Percent, Company.Location)` to select the three variables. The correct code is `trimmed_flavors_df <- flavors_df %>% select(Rating, Cocoa.Percent, Company.Location)`. In this code chunk:

- The select() function lets you select specific variables for your new data frame.
- select() takes the names of the variables you want to choose as its argument: Rating, Cocoa.Percent, Company.Location.

The company location France appears in row 1 of your tibble.

6. Next, you select the basic statistics that can help your team better understand the ratings system in your data.

   Assume the first part of your code is:

   `trimmed_flavors_df %>%`

   You want to use the summarize() and sd() functions to find the standard deviation of the rating for your data. Add the code chunk that lets you find the standard deviation for the variable *Rating.*

```
summarize(sd_rating=sd(Rating))
```

Run

Reset

What is the standard deviation of the rating?

● 0.4780624

○ 0.4458434

○ 0.3720475

○ 0.2951794

Correct

You add the code chunk `summarize(sd(Rating))` to find the standard deviation for the variable Rating. The correct code is `trimmed_flavors_df %>% summarize(sd(Rating))`. In this code chunk:

- The summarize() function lets you display summary statistics. You can use the summarize() function in combination with other functions such as mean(), max(), and min() to calculate specific statistics.
- In this case, you use sd() to calculate the standard deviation statistic for the variable Rating.

The standard deviation of the rating is 0.4780624.

7. After completing your analysis of the rating system, you determine that any rating greater than or equal to 3.5 points can be considered a high rating. You also know that Chocolate and Tea considers a bar to be super dark chocolate if the bar's cocoa percent is greater than or equal to 70%. You decide to create a new data frame to find out which chocolate bars meet these two conditions.

Assume the first part of your code is:

`best_trimmed_flavors_df <- trimmed_flavors_df %>%`

You want to apply the filter() function to the variables *Cocoa.Percent* and *Rating*. Add the code chunk that lets you filter the data frame for chocolate bars that contain at least 70% cocoa and have a rating of at least 3.5 points.

```
filter(Cocoa.Percent >=70, Rating >=3.5)
```

Run

Reset

What rating appears in row 1 of your tibble?

● 3.50

○ 4.00

○ 4.25

○ 3.75

✓ **Correct**

The code chunk `filter(Cocoa.Percent >= 70, Rating >= 3.5)` lets you filter the data frame for chocolate bars that contain at least 70% cocoa and have a rating of at least 3.5 points. The correct code is `best_trimmed_flavors_df <- trimmed_flavors_df %>% filter(Cocoa.Percent >= 70, Rating >= 3.5)`. In this code chunk:

- The filter() function lets you filter your data frame based on specific criteria.
- Cocoa.Percent and Rating refer to the variables you want to filter.
- The >= operator signifies "greater than or equal to."
- The new data frame will show all the values of Cocoa.Percent greater than or equal to 70, and all the values of Rating greater than or equal to 3.5.

The rating 3.50 appears in row 1 of your tibble.

8. Now that you've cleaned and organized your data, you're ready to create some useful data visualizations. Your team assigns you the task of creating a series of visualizations based on requests from the Chocolate and Tea management team. You decide to use ggplot2 to create your visuals.

Assume your first line of code is:

```
ggplot(data = best_trimmed_flavors_df) +
```

You want to use the geom_bar() function to create a bar chart. Add the code chunk that lets you create a bar chart with the variable *Company* on the x-axis.

```
geom_bar(mapping=aes(x=Company))
```

Run

Reset

How many bars does your bar chart display?

● 8

○ 4

○ 10

○ 6

✓ **Correct**

You add the code chunk `geom_bar(mapping = aes(x = Company))` to create a bar chart with the variable Company on the x-axis. The correct code is `ggplot(data = best_trimmed_flavors_df) + geom_bar(mapping = aes(x = Company))`. In this code chunk:

- geom_bar() is the geom function that uses bars to create a bar chart.
- Inside the parentheses of the aes() function, the code `x = Company` maps the x aesthetic to the variable Company.
- Company will appear on the x-axis of the plot.
- By default, R will put a count of the variable Company on the y-axis.

Your bar chart displays 8 bars.

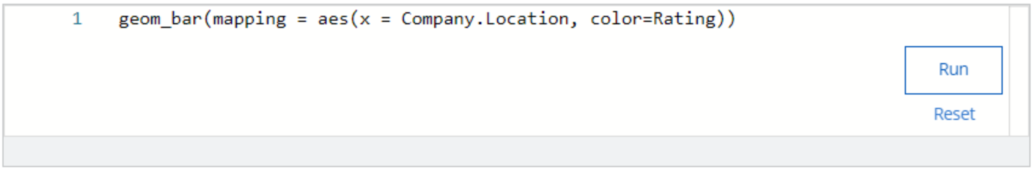
9. Your bar chart reveals the locations that produce the highest rated chocolate bars. To get a better idea of the specific rating for each location, you'd like to highlight each bar.

Assume that you are working with the code chunk:

```
ggplot(data = best_trimmed_flavors_df) +
  geom_bar(mapping = aes(x = Company.Location))
```

Add a code chunk to the second line of code to map the aesthetic *color* to the variable *Rating*.

NOTE: the three dots (...) indicate where to add the code chunk.



```
geom_bar(mapping = aes(x = Company.Location, color=Rating))
```

According to your bar chart, which two company locations produce the highest rated chocolate bars?

○ Amsterdam and U.S.A.

● Canada and France

○ Canada and U.S.A.

○ Scotland and France

Correct

You add the code chunk `color = Rating` to the second line of code to map the aesthetic color to the variable Rating. The correct code is `ggplot(data = best_trimmed_flavors_df) + geom_bar(mapping = aes(x = Company.Location, color = Rating))`. In this code chunk:

- Inside the parentheses of the aes() function, after the comma that follows x = Company.Location, write the aesthetic (color), then an equals sign, then the variable (Rating).
- The specific rating of each location will appear as a specific color that outlines each bar of your bar chart.

On your visualization, the legend titled "Rating" shows the color coding for the variable Rating. Lighter blues correspond to higher ratings and darker blues correspond to lower ratings.

According to your bar chart, the two company locations that produce the highest rated chocolate bars are Canada and France.

10. Scenario 2, continued

A teammate creates a new plot based on the chocolate bar data. The teammate asks you to make some revisions to their code.

Assume your teammate shares the following code chunk:

```
ggplot(data = best_trimmed_flavors_df) +

    geom_bar(mapping = aes(x = Company)) +
```

What code chunk do you add to the third line to create wrap around facets of the variable *Company*?

● `facet_wrap(~Company)`

○ `facet(Company)`

○ `facet_wrap(+Company)`

○ `facet_wrap(=Company)`

Correct

You write the code chunk `facet_wrap(~Company)`. In this code chunk:

- `facet_wrap()` is the function that lets you create wrap around facets of a variable.
- Inside the parentheses of the `facet_wrap()` function, type a tilde symbol (~) followed by the name of the variable (`Company`).

11. Scenario 2, continued

Your team has created some basic visualizations to explore different aspects of the chocolate bar data. You’ve volunteered to add titles to the plots. You begin with a scatterplot.

Assume the first part of your code chunk is:

```
ggplot(data = trimmed_flavors_df) +
```

```
     geom_point(mapping = aes(x = Cocoa.Percent, y = Rating)) +
```

What code chunk do you add to the third line to add the title *Suggested Chocolate* to your plot?

○ `labs(Suggested Chocolate)`

○ `labs(Suggested Chocolate = title)`

○ ` labs <- "Suggested Chocolate"`

● `labs(title = “Suggested Chocolate”)`

✓ **Correct**

You write the code chunk **`labs(title = “Suggested Chocolate”)`**. In this code chunk:

- `labs()` is the function that lets you add a title to your plot.
- In the parentheses of the labs() function, write the word title, then an equals sign, then the specific text of the title in quotation marks (`“Suggested Chocolate”`).

12. Scenario 2, continued

Next, you create a new scatterplot to explore the relationship between different variables. You want to save your plot so you can access it later on. You know that the ggsave() function defaults to saving the last plot that you displayed in RStudio, so you’re ready to write the code to save your scatterplot.

Assume your first two lines of code are:

```
ggplot(data = trimmed_flavors_df) +
```

```
  geom_point(mapping = aes(x = Cocoa.Percent, y = Rating)) +
```

What code chunk do you add to the third line to save your plot as a jpeg file with *chocolate* as the file name?

○ `ggsave(“jpeg.chocolate”)`

○ `ggsave(chocolate.jpeg)`

○ `ggsave(“chocolate.png”)`

● `ggsave(“chocolate.jpeg”)`

✓ **Correct**

You add the code chunk **`ggsave(“chocolate.jpeg”)`** to save your plot as a jpeg file with “chocolate” as the file name. In this code chunk:

- Inside the parentheses of the ggsave() function, type a quotation mark followed by the file name (chocolate), then a period, then the type of file format (jpeg), then a closing quotation mark.

13. Scenario 2, continued

As a final step in the analysis process, you create a report to document and share your work. Before you share your work with the management team at Chocolate and Tea, you are going to meet with your team and get feedback. Your team wants the documentation to include all your code and display all your visualizations.

You want to record and share every step of your analysis, let teammates run your code, and display your visualizations. What do you use to document your work?

● An R Markdown notebook

○ A data frame

○ A spreadsheet

○ A database

✓ **Correct**

You use an R Markdown notebook to document your work. The notebook lets you record and share every step of your analysis, lets your teammates run your code, and displays your visualizations.