

Congratulations! You passed!

TO PASS 80% or higher

Keep Learning

GRADE
100%

Weekly challenge 3

LATEST SUBMISSION GRADE

100%

1.

1 / 1 point

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 can create row names
- ☐ Tibbles can automatically change the names of variables
- ☒ 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.

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?

1 / 1 point

- ☐ colnames()
- ☒ head()
- ☐ preview()
- ☐ print()



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 `glimpse()` function to get a quick summary of the dataset. Write the code chunk that will give you this summary.

1 / 1 point

```
1 glimpse(ToothGrowth )
```

Run

Reset

```
Observations: 60
Variables: 3
$ len <dbl> 4.2, 11.5, 7.3, 5.8, 6.4, 10.0, 11.2, 11.2, 5.2, 7.0, 16.5, 16...
$ supp <fctr> VC, VC, VC, VC, VC, VC, VC, VC, VC, VC, VC, VC, VC, VC, V...
$ dose <dbl> 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 1.0, 1.0, 1.0...
```

How many variables does the `ToothGrowth` dataset contain?

- ☐ 5
- ☒ 3
- ☐ 2
- ☐ 4

✓ **Correct**

The code chunk `glimpse(ToothGrowth)` gives you a quick summary of the dataset. Inside the parentheses of the `glimpse()` function is the name of the dataset you want to view. The code returns a summary of the number of rows and columns in the dataset. It also shows the names of the columns and the type of data they contain. The `ToothGrowth` dataset contains 3 variables.

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?

1 / 1 point

- ☐ `rename(penguins)`
- ☐ `select(penguins)`
- ☒ `clean_names(penguins)`
- ☐ `drop_na(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 data. They write the following code:

1 / 1 point

```
penguins %>%
```

The variable `species` includes three penguin species: Adelie, Chinstrap, and Gentoo. What code chunk does the analyst add to create a data frame that only includes the Gentoo species?

- ☐ `filter(species <- "Gentoo")`
- ☐ `filter(Gentoo == species)`
- ☐ `filter(species == "Adelie")`
- ☒ `filter(species == "Gentoo")`



Correct

The code chunk is `filter(species == "Gentoo")`. The `filter` function allows the data analyst to specify which part of the data they want to view. Two equal signs in an argument mean "exactly equal to." Using this operator instead of the assignment operator `<-` calls only the data about Gentoo penguins to the dataset.

6. You are working with the penguins dataset. You want to use the `summarize()` and `mean()` functions to find the mean value for the variable `body_mass_g`. You write the following code:

1 / 1 point

```
penguins %>%
```

```
drop_na() %>%
```

```
group_by(species) %>%
```

Add the code chunk that lets you find the mean value for the variable `body_mass_g`.

```
1 summarise(mean = mean(body_mass_g))
```

Run

Reset

```
# A tibble: 3 <U+00D7> 2
  species    mean
  <chr>    <dbl>
1 Adelie 3706.164
2 Chinstrap 3733.088
3 Gentoo 5092.437
```

What is the mean body mass in g for the Adelie species?

- ☐ 4207.433
- ☒ 3706.164
- ☐ 5092.437
- ☐ 3733.088



Correct

The code chunk `summarize(mean(body_mass_g))` lets you find the mean value for the variable `body_mass_g`. The correct code is `penguins %>% drop_na() %>% group_by(species) %>% summarize(mean(body_mass_g))`. 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 `mean()` to calculate the mean value for body mass. The mean body mass for the Adelie species is 3706.164g.

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

1 / 1 point

- ☐ `mutate(salary_data, rate = wages * 40)`
- ☐ `mutate(salary_data, wages = rate + 40)`
- ☐ `mutate(wages = rate * 40)`
- ☒ `mutate(salary_data, wages = rate * 40)`



Correct

The code chunk is `mutate(salary_data, wages = rate * 40)`. The analyst can use the `mutate()` function to create a new column called `wages` that includes data from the `rate` column multiplied by 40. 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?

1 / 1 point

- ☒ `unite(retail, "price", price_dollars, price_cents, sep=".")`
- ☐ `unite(retail, "price", price_cents, sep=".")`
- ☐ `unite(retail, "price", price_dollars, price_cents)`
- ☐ `unite(retail, price_dollars, price_cents, sep=".")`



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.

1 / 1 point

A data analyst is using statistical measures to get a better understanding of their data. What function can they use to determine how strongly related are two of the variables?

- ☐ `sd()`
- ☒ `cor()`
- ☐ `bias()`
- ☐ `mean()`



Correct

The `cor()` returns the correlation between two variables. Correlation shows us how strong the relationship is between two variables .

10. A data analyst wants to find out how much the predicted outcome and the actual outcome of their data model differ. What function can they use to quickly measure this?

1 / 1 point

- ☒ `bias()`
- ☐ `mean()`
- ☐ `sd()`
- ☐ `cor()`



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.