

# Activity 8: Computational Thinking 2

Joshua Holt and Sora Feeney

## 1. Conditionals

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
v dplyr      1.1.4      v readr      2.1.5
v forcats    1.0.0      v stringr    1.5.1
v ggplot2    4.0.0      v tibble     3.3.0
v lubridate  1.9.4      v tidyr      1.3.1
v purrr      1.1.0
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()     masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
```

```
library(here)
```

here() starts at /Users/josh1/Desktop/GitAndGithub-DataSci-172026/ds4eeb\_Lab4/Activity 8 Att

### Q1.1)

```
x <- 11

# Check if the value of x is greater than 10
if(x > 10)
{
```

```

    print(paste(x, "is greater than 10"))
  } else
  {
    print(paste(x, "is less than or equal to 10"))
  }

```

```
[1] "11 is greater than 10"
```

The output message states “11 is greater than 10”

### Q1.2)

```

y <- c(-2, 42, 0, 10)

for (i in 1:length(y)) {
  if (y[i] < 0) {
    print("value is less than 0")
  } else {
    if (y[i] == 0) {
      print("value is 0!!!")
    } else {
      # don't need if (y[i]>0) because that's the only option left
      print("value is greater than 0")
    }
  }
}

```

```

[1] "value is less than 0"
[1] "value is greater than 0"
[1] "value is 0!!!"
[1] "value is greater than 0"

```

### Q1.3)

```

library(lterdatasampler)
library(tidyverse)

```

```
?nwt_pikas
```

Stress was measured by observing the amount of glucocorticoid metabolite found in pika feces. The stress variable is “concentration\_pg\_m” which denotes glucocorticoid metabolite (GCM) concentration in picogram GCM/gram dry pika feces.

Q1.4)

A row represents an individual pika poop

Q1.5)

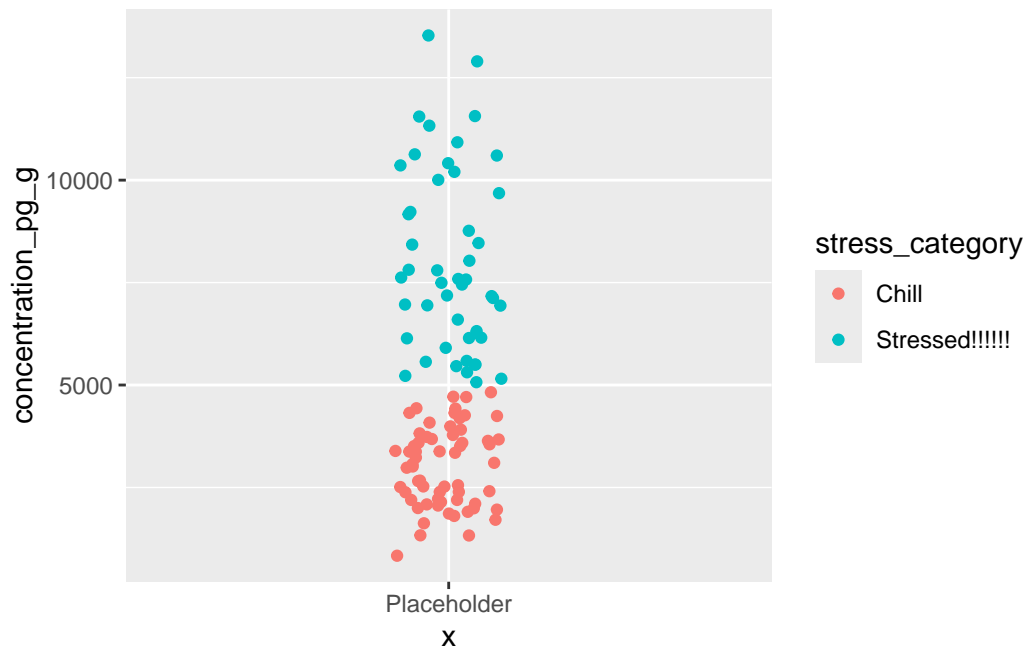
```
nwt_pikas_categ <- nwt_pikas %>%
  # Call the new column stress_category
  mutate(stress_category = case_when(
    # When the value is > 5000, make the new column's value "Stressed!!!!"
    concentration_pg_g > 5000 ~ "Stressed!!!!!!",
    # Otherwise, make the new column's value "Chill"
    .default = "Chill"
  ))

# Check out the first 6 rows, but remove the utm columns just for visibility
head(nwt_pikas_categ %>% select(-c(utm_easting, utm_northing)))
```

```
# A tibble: 6 x 7
  date      site      station sex concentration_pg_g elev_m stress_category
<date>    <fct>    <fct> <fct>          <dbl>   <dbl> <chr>
1 2018-06-08 Cable Gate Cable G~ male          11563.   3343. Stressed!!!!!!
2 2018-06-08 Cable Gate Cable G~ male          10629.   3353. Stressed!!!!!!
3 2018-06-08 Cable Gate Cable G~ male          10924.   3358. Stressed!!!!!!
4 2018-06-13 West Knoll West Kn~ male          10414.   3578. Stressed!!!!!!
5 2018-06-13 West Knoll West Kn~ male          13531.   3584. Stressed!!!!!!
6 2018-06-13 West Knoll West Kn~ <NA>           7799.   3595. Stressed!!!!!!
```

```
# make scatterplot with color
nwt_pikas_categ |>
  ggplot(aes(x = "Placeholder",
             y = concentration_pg_g,
             #color mapped to stress category
```

```
color = stress_category)) +  
geom_jitter(width = 0.1)
```



## 2. DIY a for loop and an if else statement / case when

Q2.1)

We chose to use the iris data set.

Q2.2)

We are using the Iris data set to find the median value for each numeric column. In this data set, the columns represent different measurements/variables of flowers. This is taking into account all flower species.

Q2.3)

```
#Get the column names of the dataframe, store in a vector "names"
names <- colnames(iris)

# Creating for loop
for (i in 1:4) {

  # Print out the "i"th element of the vector to print alongside the output
  print(names[i])

  # This prints out column i
  print(median(iris[,i]))
}
```

```
[1] "Sepal.Length"
[1] 5.8
[1] "Sepal.Width"
[1] 3
[1] "Petal.Length"
[1] 4.35
[1] "Petal.Width"
[1] 1.3
```

## Q2.4)

We are comparing all values in the Sepal Length column to the calculated median of Sepal Length across all individuals. Then print out the final conclusion of whether the values in each cell are less than, equal to, or greater than the median of Sepal Length.

## Q2.5)

```
#Redifing data frame to only include Sepal Length column
S.L.Med <- iris$Sepal.Length

#For 1 through the length of the vector "S.L.Med"
for (i in 1:length(S.L.Med)){
  #Check if the cell value is NA
  if(!is.na(S.L.Med[i])){
    #Check the value using nested if-else statements
    if(S.L.Med[i] < 5.8){
      #If the element is less than the previously calculated median
```

```
print("Sepal Length is less than the median (5.8 cm)")
} else {
    if (S.L.Med[i] == 5.8) {
        #If the element is exactly equal to previously calculated median
        print("Sepal Length is equal to the median (5.8 cm)")
    } else {
        #If the element is exactly greater to previously calculated median
        #Don't need if statement because it is the only option left
        print("Sepal Length is greater than the median (5.8 cm)")
    }
}
}
```

[illegible]

[illegible]

[illegible]



[illegible]