Lecture Assignment 19

Viraj Vijaywargiya

2022-10-05

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Part 19.4.4

Question 1

if statement is used to test a single condition, or an object of length 1. Whereas, ifelse() is used for multiple conditions, chaining multiple if statements together, meaning that it also works with vectors with length greater than 1.

```
x <- 1
if (x == 1) {
  "True"
} else {
  "False"
}
```

```
## [1] "True"
```

```
x <- 3
if (x == 1) {
  "True"
} else if (x == 0){
  "False"
} else {
  "Invalid"
}
```

```
## [1] "Invalid"
```

Question 2

```
GreetFunc <- function(){
  hr <- lubridate::hour(lubridate::now())
  if (dplyr::between(hr, 12, 18)){
    print("good afternoon")
  } else if(dplyr::between(hr, 18, 24)){
    print("good evening")
  } else {
    print("good morning")
  }
}

GreetFunc()
```

```
## [1] "good morning"
```

Question 3

```
fizzbuzz <- function(n){
  if (n %% 3 == 0 && n %% 5 == 0){
    "fizzbuzz"
  } else if (n %% 3 == 0) {
    "fizz"
  } else if (n %% 5 == 0) {
    "buzz"
  } else {
    n
  }
}

fizzbuzz(15)
```

```
## [1] "fizzbuzz"
```

```
fizzbuzz(18)
```

```
## [1] "fizz"
```

```
fizzbuzz(35)
```

```
## [1] "buzz"
```

```
fizzbuzz(16)
```

```
## [1] 16
```

Question 4

```
temp <- function(t){
  cut(t, breaks = c(-Inf,0,10,20,30,Inf),
      labels = c('freezing','cold','cool','warm','hot'))
}

temp(c(-5,0,5,10,15,20,25,30,35))
```

```
## [1] freezing freezing cold     cold     cool     cool     warm     warm
## [9] hot
## Levels: freezing cold cool warm hot
```

We would add the argument, right = FALSE, to cut() to indicate intervals should be closed if $<$ is used instead of <=. Because we have multiple conditions, vector of length greater than 1, cut() has an advantage over "if" if we have many values in temp.

Part 19.5.5

Question 1

```
commas <- function(...) stringr::str_c(..., collapse = ", ")
#commas(letters, collapse = "-")
```

With the way commas() is defined above, “commas(letters, collapse =”-“)” gives an error. This is because collapse is set to “,” in the inital definition of the commas() function. The same named argument is given twice, which is an error.