Pipe operator

12/05/2020

Necessary packages

```
library(magrittr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   0.8.5
## v tidyr   1.0.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract()   masks magrittr::extract()
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x purrr::set_names() masks magrittr::set_names()
```

```
library(stringr)
```

1. Explore the outputs of the following functions.

```
x <- c(1,2)
sum(x, 3)
x %>% sum(3)
sum(x, 3) == x %>% sum(3)
```

```
seq(3, 10, 2)
3 %>% seq(10, 2)
10 %>% seq(3, ., 2)
```

```
# method 1
filter(iris, Sepal.Length >= 7.0)
# method 2 - using pipe
iris %>% filter(Sepal.Length >= 7.0)
```

```
# method 1
ir <- as_tibble(iris)
select(ir, Species)
# method 2 - using pipe
iris %>% as_tibble() %>% select(Species)
```

2. Rewrite the following code using the pipe operator.

```
str_c("good", sample(c("health", "food", "work", "day"), 1))
```

```
[1] "goodwork"
```

3. Rewrite the following code using the pipe operator.

```
summarize(filter(iris, Species=="setosa"), median(Sepal.Length))
```

```
  median(Sepal.Length)
1                    5
```