Joins

Ed Gonzalez

```
install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/tz/sh20cj15711657_9_1d4v6m00000gn/T//RtmpJR5yYm/downloaded_packages
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
baseball01 <- read_csv("/Users/ed/Downloads/baseball01.csv")
```

```
## Rows: 5 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Name
## dbl (3): Hits, Dingers, RBI
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
baseball02 <- read_csv("/Users/ed/Downloads/baseball02.csv")
```

```
## Rows: 4 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Name, Color, Position
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
ljbaseball <- left_join(baseball01, baseball02)
```

```
## Joining, by = "Name"
```

```
ljbaseball
```

```
## # A tibble: 5 x 6
##   Name    Hits Dingers   RBI Color Position
##   <chr>  <dbl>   <dbl> <dbl> <chr> <chr>
## 1 Tinker   100      15    60 blue  SS
## 2 Evers    120      20    70 <NA>  <NA>
## 3 Chance   110      30    75 <NA>  <NA>
## 4 Cobb     150      30    80 red   2B
## 5 Ruth      90      40    90 <NA>  <NA>
```

With the left_join function, we get the above results. We now have a table with all of the Names from baseball01, but with the added variables in baseball02 with "NA" in cells that didn't contain any information.

```
ijbaseball <- inner_join(baseball01, baseball02)
```

```
## Joining, by = "Name"
```

```
ijbaseball
```

```
## # A tibble: 2 x 6
##   Name    Hits Dingers   RBI Color Position
##   <chr>  <dbl>   <dbl> <dbl> <chr> <chr>
## 1 Tinker   100      15    60 blue  SS
## 2 Cobb     150      30    80 red   2B
```

Inner_join function grabs the Names that baseball01/02 have in common. In this case, we have Tinker and Cobb as the only rows as they contain information for all of the variables in both data sets.

```
fjbaseball <- full_join(baseball01, baseball02)
```

```
## Joining, by = "Name"
```

```
fjbaseball
```

```
## # A tibble: 7 x 6
##   Name    Hits Dingers   RBI Color  Position
##   <chr>  <dbl>   <dbl> <dbl> <chr>  <chr>
## 1 Tinker   100      15    60 blue   SS
## 2 Evers    120      20    70 <NA>   <NA>
## 3 Chance   110      30    75 <NA>   <NA>
## 4 Cobb     150      30    80 red    2B
## 5 Ruth      90      40    90 <NA>   <NA>
## 6 Mantle    NA      NA    NA green  CF
## 7 Maris     NA      NA    NA yellow RF
```

With full_join, it is fairly self-explanatory as it joins the data sets entirely and inserts "NA" in any cells that doesn't contain data from either set.