Data and File Management using R - Abstract/Outline

Introduction

The workshop covers basic but the essential steps to read, understand and visualise the data before performing any downstream task. This hands-on training consists of learning R packages and functions to import the data from various file formats or sources and cleaning and reshaping the data, followed by data summarisation and manipulation. At the end,will go through the different data visualisation techniques using ggplot2. The workshop will give the participants an opportunity to learn some of the strongest and widely used data manipulation and visualisation R packages such as Dplyr, Tidyverse and ggplot2.

```
sessionInfo()
```

```
## R version 4.2.2 (2022-10-31)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur ... 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRblas.0.dy
lib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRlapack.dy
lib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
##  [1] digest_0.6.31   R6_2.5.1        jsonlite_1.8.4  evaluate_0.20
##  [5] cachem_1.0.6    rlang_1.0.6     cli_3.6.0       rstudioapi_0.14
##  [9] jquerylib_0.1.4 bslib_0.4.2     rmarkdown_2.20  tools_4.2.2
## [13] xfun_0.37       yaml_2.3.7      fastmap_1.1.0   compiler_4.2.2
## [17] htmltools_0.5.4 knitr_1.42      sass_0.4.5
```

List of libraries/ packages to install/import

```
## list of libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(ggplot2)
library(readr)
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(readxl)
library(gdata)
```

```
## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.
```

```
##
```

```
## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.
```

```
##
## Attaching package: 'gdata'
```

```
## The following objects are masked from 'package:data.table':
##
##     first, last
```

```
## The following objects are masked from 'package:dplyr':
##
##     combine, first, last
```

```
## The following object is masked from 'package:stats':
##
##     nobs
```

```
## The following object is masked from 'package:utils':
##
##     object.size
```

```
## The following object is masked from 'package:base':
##
##     startsWith
```

```
library(datasets)
library(assertive)
```

Importing data into R

From flat files

1. How to import the common formats of flat file data with base R functions
2. Using dedicated R packages

```
## Using base R package 'utils'

## Reading from .txt (unformatted text) file

help("read.delim")
?read.delim

data <- read.delim("OpenData500.txt", sep = "\t") ## field separator character. Can also be 'space'.
data <- read.delim("OpenData500.txt", header = T) ## default is header  = True, which will read the top row as column names
data <- read.delim("OpenData500.txt", header = F) ## change header to False which will read the top row as part of the data and will import the data without column names

## Reading from .csv (comma-separated values) files

data <- read.csv("OpenData500.csv")
data <- read.csv("OpenData500.csv", stringsAsFactors = F, sep = ",", check.names = F) ## check description of this

## Data is stored as dataframe which has columns and rows

#data[c(rows), c(columns)]
subdata <- data[c(1:10), c(1:5)] ## first 10 rows and first 5 columns
subdata <- data[,c(1:5)] ## all rows and first 5 columns
subdata <- data[c(1:10),] ## first 10 rows and all columns

#View(data) ## or click on the variable in the 'Data' environment
#head(data) ## head prints the top5 rows from all columns - Some variables are not reader friendly
glimpse(data)
```

```
## Rows: 65
## Columns: 24
## $ company_name             <chr> "ADG Engineers (Aust) Pty Ltd", "Advance Cair…
## $ url                      <chr> "https://www.adgce.com/", "www.advancecairns.…
## $ poa                      <int> 4066, 4870, 5067, 2072, 5067, 2607, 2537, 200…
## $ city                     <chr> "Brisbane", "Cairns", "Adelaide", "Sydney", "…
## $ state                    <chr> "QLD", "QLD", "SA", "NSW", "SA", "ACT", "NSW"…
## $ country                  <chr> "au", "au", "au", "au", "au", "au", "au", "au…
## $ year_founded             <int> 2002, 2000, 1980, 2015, 1932, 2014, 2008, 194…
## $ full_time_employees_low  <int> 51, 1, 11, 1, 51, 11, 1, 1001, 1, 11, 11, 11,…
## $ full_time_employees_high <chr> "200", "10", "50", "10", "200", "50", "10", "…
## $ company_type             <chr> "Private", "Nonprofit", "Private", "Private",…
## $ company_category         <chr> "Construction", "Economic Development and Adv…
## $ revenue_source           <chr> "Consulting, Government contract", "Governmen…
## $ business_model           <chr> "Business to Business", "Business to Governme…
## $ social_impact            <chr> "Citizen engagement and participation, Public…
## $ description              <chr> "We are not limited by discipline, restricted…
## $ description_short        <chr> "We believe project success is created throug…
## $ source_count_low         <int> 1, 101, NA, 11, 11, NA, 1, 11, 11, 1, 11, 11,…
## $ source_count_high        <int> 10, NA, NA, 50, 50, NA, 10, 50, 50, 10, 50, 5…
## $ data_types               <chr> "Business, Economics, Energy, Environment, Fi…
## $ data_comments            <chr> "At this moment in time the required data tha…
## $ example_uses             <chr> "We use BIM (Building information Model) plat…
## $ data_impacts             <chr> "Cost efficiency', 'New or improved product/s…
## $ requested_data           <chr> "Australian Open BIM standards, Once this dat…
## $ data_sources             <chr> "New South Wales Government (NSW Land and Pro…
```

```
summary(data)
```

```
##  company_name           url                 poa           city
##  Length:65          Length:65          Min.   :2000   Length:65
##  Class :character   Class :character   1st Qu.:2122   Class :character
##  Mode  :character   Mode  :character   Median :3000   Mode  :character
##                                        Mean   :3545
##                                        3rd Qu.:4152
##                                        Max.   :7253
##
##     state              country           year_founded  full_time_employees_low
##  Length:65          Length:65          Min.   :1896   Min.   :    1.0
##  Class :character   Class :character   1st Qu.:1985   1st Qu.:    1.0
##  Mode  :character   Mode  :character   Median :2001   Median :   11.0
##                                        Mean   :1992   Mean   :  375.6
##                                        3rd Qu.:2010   3rd Qu.:   11.0
##                                        Max.   :2015   Max.   :10001.0
##
##  full_time_employees_high company_type       company_category
##  Length:65                Length:65          Length:65
##  Class :character         Class :character   Class :character
```

```
##  Mode  :character         Mode  :character   Mode  :character
##
##
##
##
##  revenue_source     business_model     social_impact      description
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  description_short  source_count_low source_count_high  data_types
##  Length:65          Min.   :  1.00   Min.   : 10.00    Length:65
##  Class :character   1st Qu.:  1.00   1st Qu.: 10.00    Class :character
##  Mode  :character   Median : 11.00   Median : 50.00    Mode  :character
##                     Mean   : 30.45   Mean   : 38.84
##                     3rd Qu.: 51.00   3rd Qu.: 50.00
##                     Max.   :101.00   Max.   :100.00
##                     NA's   :10       NA's   :22
##  data_comments      example_uses       data_impacts       requested_data
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  data_sources
##  Length:65
##  Class :character
##  Mode  :character
##
##
##
##
```

```
## table() shows what are the categories in any variable and number of entries for each category
table(data$company_type)
```

```
##
##       Government          Nonprofit            Private             Public
##                1                 13                  45                  5
## Social Enterprise
##                1
```

```
table(data$city) ## ask students
```

```
##
##                                          Adelaide
##                            1                     3
##                     Armidale                  Bass
##                            1                     1
##                     Brisbane                Brisbane
##                            9                     2
##                       Cairns                 Canberra
##                            1                     7
##                    Canberra                 Chermside
##                            1                     1
##             Currumbin Valley              George Town
##                            1                     1
##                       Hobart Kunda Park (Sunshine Coast)
##                            2                     1
##                    melbourne                Melbourne
##                            1                     3
##      Molendinar, Gold Coast              North Sydney
##                            1                     2
##                        Perth                   PERTH
##                            4                     1
##                      Pyrmont                  Sydney
##                            1                    15
##                  Tuross Head                 Virtual
##                            1                     1
##                     West End                Willawong
##                            1                     1
##                      Wodonga
##                            1
```

```
table(data$state) ## ask students - which state has most companies
```

```
##
## ACT NSW QLD  SA TAS VIC  WA
##   9  20  19   3   3   6   5
```

```
table(data$business_model) ## ask students - which business model was most observed in the data
```

```
##
##                                                                  Business to Bus
iness
##
11
##                     Business to Business, Business to Consumer, Business to Gover
nment
##
24
## Business to Business, Business to Consumer, Business to Government, Online to Of
fline
##
1
##                                           Business to Business, Business to Gover
nment
##
15
##                                                                   Business to Con
sumer
##
6
##                                                               Business to Gover
nment
##
2
##                                           Business to Government, Industry Associ
ation
##
1
##                                                                       Community e
vents
##
1
##       Membership organisation representing medical practitioners and medical stu
dents
##
1
##                                                          Non government service del
ivery
##
1
##                                                                Organisation to comm
unity
##
1
##                                                                 Research Infrastru
cture
##
1
```

```
## Checking and changing variable type
summary(data) ## check for 'full_time_employees'
```

```
##  company_name           url                 poa           city
##  Length:65          Length:65          Min.   :2000   Length:65
##  Class :character   Class :character   1st Qu.:2122   Class :character
##  Mode  :character   Mode  :character   Median :3000   Mode  :character
##                                        Mean   :3545
##                                        3rd Qu.:4152
##                                        Max.   :7253
##
##     state              country           year_founded  full_time_employees_low
##  Length:65          Length:65          Min.   :1896   Min.   :    1.0
##  Class :character   Class :character   1st Qu.:1985   1st Qu.:    1.0
##  Mode  :character   Mode  :character   Median :2001   Median :   11.0
##                                        Mean   :1992   Mean   :  375.6
##                                        3rd Qu.:2010   3rd Qu.:   11.0
##                                        Max.   :2015   Max.   :10001.0
##
##  full_time_employees_high company_type       company_category
##  Length:65                Length:65          Length:65
##  Class :character         Class :character   Class :character
##  Mode  :character         Mode  :character   Mode  :character
##
##
##
##
##  revenue_source     business_model     social_impact      description
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  description_short  source_count_low source_count_high  data_types
##  Length:65          Min.   :  1.00   Min.   : 10.00    Length:65
##  Class :character   1st Qu.:  1.00   1st Qu.: 10.00    Class :character
##  Mode  :character   Median : 11.00   Median : 50.00    Mode  :character
##                     Mean   : 30.45   Mean   : 38.84
##                     3rd Qu.: 51.00   3rd Qu.: 50.00
##                     Max.   :101.00   Max.   :100.00
##                     NA's   :10       NA's   :22
##  data_comments      example_uses       data_impacts       requested_data
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
```

```
##
##
##  data_sources
##  Length:65
##  Class :character
##  Mode  :character
##
##
##
##
```

```
class(data$full_time_employees_high)
```

```
## [1] "character"
```

```
data$full_time_employees_high <- as.integer(data$full_time_employees_high)
```

```
## Warning: NAs introduced by coercion
```

```
summary(data) ## check for 'full_time_employees'
```

```
##  company_name           url                 poa           city
##  Length:65          Length:65          Min.   :2000   Length:65
##  Class :character   Class :character   1st Qu.:2122   Class :character
##  Mode  :character   Mode  :character   Median :3000   Mode  :character
##                                        Mean   :3545
##                                        3rd Qu.:4152
##                                        Max.   :7253
##
##     state              country           year_founded  full_time_employees_low
##  Length:65          Length:65          Min.   :1896   Min.   :    1.0
##  Class :character   Class :character   1st Qu.:1985   1st Qu.:    1.0
##  Mode  :character   Mode  :character   Median :2001   Median :   11.0
##                                        Mean   :1992   Mean   :  375.6
##                                        3rd Qu.:2010   3rd Qu.:   11.0
##                                        Max.   :2015   Max.   :10001.0
##
##  full_time_employees_high company_type       company_category
##  Min.   :   10.0          Length:65          Length:65
##  1st Qu.:   10.0          Class :character   Class :character
##  Median :   50.0          Mode  :character   Mode  :character
##  Mean   :  608.1
##  3rd Qu.:   50.0
##  Max.   :10000.0
##  NA's   :2
##  revenue_source     business_model     social_impact      description
```

```
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  description_short  source_count_low source_count_high  data_types
##  Length:65          Min.   :  1.00   Min.   : 10.00    Length:65
##  Class :character   1st Qu.:  1.00   1st Qu.: 10.00    Class :character
##  Mode  :character   Median : 11.00   Median : 50.00    Mode  :character
##                     Mean   : 30.45   Mean   : 38.84
##                     3rd Qu.: 51.00   3rd Qu.: 50.00
##                     Max.   :101.00   Max.   :100.00
##                     NA's   :10       NA's   :22
##  data_comments      example_uses       data_impacts       requested_data
##  Length:65          Length:65          Length:65          Length:65
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  data_sources
##  Length:65
##  Class :character
##  Mode  :character
##
##
##
##
```

```
## Observations with lowest or highest count
dat <- table(data$state)
class(dat)
```

```
## [1] "table"
```

```
dat <- as.data.frame(dat)
dat[which.min(dat$Freq),] ## which state has minimum companies
```

```
##   Var1 Freq
## 4   SA    3
```

```
dat[which.max(dat$Freq),] ## ask students - which state has maximum companies
```

```
##   Var1 Freq
## 2  NSW   20
```

```
##########################################################################
##########################################################################

## Using specific R package 'readr' - need to install

data <- read_csv("OpenData500.csv")
```

```
## Rows: 65 Columns: 24
## ── Column specification ────────────────────────────────────────────────
─
## Delimiter: ","
## chr (18): company_name, url, city, state, country, company_type, company_cat...
## dbl  (5): poa, year_founded, full_time_employees_low, source_count_low, sour...
## num  (1): full_time_employees_high
##
## 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 messag
e.
```

```
data <- read_tsv("OpenData500.txt")
```

```
## Rows: 65 Columns: 24
## ── Column specification ────────────────────────────────────────────────
─
## Delimiter: "\t"
## chr (18): company_name, url, city, state, country, company_type, company_cat...
## dbl  (5): poa, year_founded, full_time_employees_low, source_count_low, sour...
## num  (1): full_time_employees_high
##
## 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 messag
e.
```

```
column_names <- colnames(data)

## Through skip and n_max you can control which part of your flat file you're actua
lly importing into R.
## import observations 7, 8, 9, 10 and 11
data <- read_tsv("OpenData500.txt", skip = 6, n_max = 5) ## Once you skip some line
s, you also skip the first line that can contain column names!
```

```
## New names:
## Rows: 5 Columns: 24
## ── Column specification
## ──────────────────────────────────────────────────────── Delimiter: "\t" chr
## (18): Almighty God Blessing Family Day Care pty ltd, ...2, Canberra, ACT... dbl
## (6): 2607, 2014, 11, 50, ...17, ...18
## 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.
## • `` -> `...2`
## • `` -> `...17`
## • `` -> `...18`
## • `` -> `...19`
## • `` -> `...20`
## • `` -> `...21`
## • `` -> `...22`
## • `-` -> `-...23`
## • `-` -> `-...24`
```

```
data <- read_tsv("OpenData500.txt", skip = 6, n_max = 5, col_names = column_names)
## provide the column names separately.
```

```
## Rows: 5 Columns: 24
## ── Column specification ────────────────────────────────────────────────────
─
## Delimiter: "\t"
## chr (18): company_name, url, city, state, country, company_type, company_cat...
## dbl  (6): poa, year_founded, full_time_employees_low, full_time_employees_hi...
##
## 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 messag
e.
```

```
## Using specific R package 'data.table' - need to install
?fread
data <- fread("OpenData500.csv")
## drop and select, to drop or select variables/columns of interest
data <- fread("OpenData500.csv", drop = 2:6)
data <- fread("OpenData500.csv", select = c(1, 5:10))
```

From excel files

```
## Using specific R package 'readxl' - need to install

excel_sheets("Dataset.xlsx") ## to print the names of sheets in the excel file
```

```
## [1] "OpenData500"    "TrafficOffence"
```

```
?read_excel
traffic_dat <- read_excel(path = "Dataset.xlsx", sheet = "TrafficOffence") ## impor
t the specific data sheet from excel file by giving name of the sheet
traffic_dat <- read_excel(path = "Dataset.xlsx", sheet = 2) ## import the specific
data sheet from excel file by giving number of the sheet
traffic_dat <- read_excel(path = "Dataset.xlsx", sheet = 2, range = "A1:K1501", col
_names = T) ## A cell range to read from.
glimpse(traffic_dat)
```

```
## Rows: 1,500
## Columns: 11
## $ Offence_Month  <dttm> 2022-03-01, 2022-03-01, 2022-03-01, 2022-03-01, 2022-0…
## $ Rego_State     <chr> "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT",…
## $ Clt_Catg       <chr> "Diplomatic", "Diplomatic", "Diplomatic", "Diplomatic",…
## $ Camera_Type    <chr> "FIXED ONLY SPEED CAMERA", "FIXED ONLY SPEED CAMERA", "…
## $ Location_Code  <dbl> 1035, 1027, 3048, 230, 3016, 3038, 3014, 251, 1, 93, 11…
## $ Location_Desc  <chr> "TUGGERANONG PARKWAY NEAR COTTER ROAD OVERPASS SOUTHBOU…
## $ Offence_Desc   <chr> "20 Non-school zone exceed speed limit by <= 15km/h", "…
## $ Sum_Pen_Amt    <dbl> 301, 602, 1385, 301, 975, 325, 325, 301, 1566, 301, 444…
## $ Sum_Inf_Count  <dbl> 1, 2, 1, 1, 3, 1, 1, 1, 2, 1, 1, 1, 13, 2, 1, 1, 5, 5, …
## $ Sum_With_Amt   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ Sum_With_Count <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
```

```
summary(traffic_dat)
```

```
##  Offence_Month                  Rego_State          Clt_Catg
##  Min.   :2010-07-01 00:00:00   Length:1500        Length:1500
##  1st Qu.:2010-07-01 00:00:00   Class :character   Class :character
##  Median :2022-03-01 00:00:00   Mode  :character   Mode  :character
##  Mean   :2017-07-05 10:38:24
##  3rd Qu.:2022-03-01 00:00:00
##  Max.   :2022-11-01 00:00:00
##
##  Camera_Type        Location_Code  Location_Desc      Offence_Desc
##  Length:1500        Min.   :   1   Length:1500        Length:1500
##  Class :character   1st Qu.: 114   Class :character   Class :character
##  Mode  :character   Median :1016   Mode  :character   Mode  :character
##                     Mean   :1001
##                     3rd Qu.:1030
##                     Max.   :3077
##
##   Sum_Pen_Amt      Sum_Inf_Count   Sum_With_Amt    Sum_With_Count
##  Min.   :     0   Min.   :   0   Min.   :   0.0   Min.   : 0.0000
##  1st Qu.:   314   1st Qu.:   1   1st Qu.:   0.0   1st Qu.: 0.0000
##  Median :   745   Median :   2   Median :   0.0   Median : 0.0000
##  Mean   :  4093   Mean   :  13   Mean   : 112.3   Mean   : 0.3287
##  3rd Qu.:  1911   3rd Qu.:   4   3rd Qu.:   0.0   3rd Qu.: 0.0000
##  Max.   :523439   Max.   :1739   Max.   :9015.0   Max.   :19.0000
##  NA's   :9                       NA's   :6
```

```
## Using specific R package 'gdata' - need to install
?read.xls
## Converting the Excel file to a .csv file using a Perl script, and then reading t
hat .csv file with the read.csv() function that is loaded by default in R, through
the utils package.
traffic_dat <- read.xls("Dataset.xlsx", sheet = 2)

##Finish the read.xls() call that reads data from the second sheet of excel file "D
ataset.xls", skip the first 50 rows of the sheet. Make sure to set header appropria
tely and that the strings are not imported as factors.
traffic_dat <- read.xls("Dataset.xlsx", sheet = 2, skip = 50, header = F, stringsAs
Factors = F) ## first row was not considered as column headers. In this case, we ha
ve to provide the column headers ourselves (a vector of column names).
glimpse(traffic_dat)
```

```
## Rows: 1,451
## Columns: 11
## $ V1  <chr> "Mar-22", "Mar-22", "Mar-22", "Mar-22", "Mar-22", "Mar-22", "Mar-2…
## $ V2  <chr> "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "AC…
## $ V3  <chr> "OTHER ORGN", "OTHER ORGN", "OTHER ORGN", "OTHER ORGN", "OTHER ORG…
## $ V4  <chr> "FIXED ONLY SPEED CAMERA", "FIXED ONLY SPEED CAMERA", "FIXED ONLY …
## $ V5  <int> 1027, 1035, 1031, 1031, 1027, 1030, 1030, 1029, 1032, 119, 187, 30…
## $ V6  <chr> "BARTON HIGHWAY BETWEEN GUNGAHLIN DRIVE AND ELLENBOROUGH STREET", …
## $ V7  <chr> "20 Non-School Zone Exceed Speed Limit > 15 But <= 30 Km/H", "20 N…
## $ V8  <int> 1980, 1265, 2530, 3960, 6325, 3795, 1980, 5060, 1265, 3795, 1265, …
## $ V9  <int> 1, 1, 2, 2, 5, 3, 1, 4, 1, 3, 1, 2, 2, 1, 1, 2, 1, 1, 2, 1, 1, 1, …
## $ V10 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ V11 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
```

From web

```
## Downloading the .csv data from web

url_csv <- "https://data.gov.au/data/dataset/f51453c9-323e-4b4c-808e-52b635e99e8c/resource/5c0af3fd-2609-4e5e-84ca-7c8674697381/download/od500aucompany.csv" ##you can use the standard importing functions with https:// connections since R version 3.2.2.
data <- read.csv(url_csv)
data <- read_csv(url_csv)
```

```
## Rows: 65 Columns: 24
## ── Column specification ────────────────────────────────────────────────────────
─
## Delimiter: ","
## chr (18): company_name, url, city, state, country, company_type, company_cat...
## dbl  (5): poa, year_founded, full_time_employees_low, source_count_low, sour...
## num  (1): full_time_employees_high
##
## 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 messag
e.
```

```
glimpse(data)
```

```
## Rows: 65
## Columns: 24
## $ company_name             <chr> "ADG Engineers (Aust) Pty Ltd", "Advance Cair…
## $ url                      <chr> "https://www.adgce.com/", "www.advancecairns.…
## $ poa                      <dbl> 4066, 4870, 5067, 2072, 5067, 2607, 2537, 200…
## $ city                     <chr> "Brisbane", "Cairns", "Adelaide", "Sydney", "…
## $ state                    <chr> "QLD", "QLD", "SA", "NSW", "SA", "ACT", "NSW"…
## $ country                  <chr> "au", "au", "au", "au", "au", "au", "au", "au…
## $ year_founded             <dbl> 2002, 2000, 1980, 2015, 1932, 2014, 2008, 194…
## $ full_time_employees_low  <dbl> 51, 1, 11, 1, 51, 11, 1, 1001, 1, 11, 11, 11,…
## $ full_time_employees_high <dbl> 200, 10, 50, 10, 200, 50, 10, 5000, 10, 50, 5…
## $ company_type             <chr> "Private", "Nonprofit", "Private", "Private",…
## $ company_category         <chr> "Construction", "Economic Development and Adv…
## $ revenue_source           <chr> "Consulting, Government contract", "Governmen…
## $ business_model           <chr> "Business to Business", "Business to Governme…
## $ social_impact            <chr> "Citizen engagement and participation, Public…
## $ description              <chr> "We are not limited by discipline, restricted…
## $ description_short        <chr> "We believe project success is created throug…
## $ source_count_low         <dbl> 1, 101, NA, 11, 11, NA, 1, 11, 11, 1, 11, 11,…
## $ source_count_high        <dbl> 10, NA, NA, 50, 50, NA, 10, 50, 50, 10, 50, 5…
## $ data_types               <chr> "Business, Economics, Energy, Environment, Fi…
## $ data_comments            <chr> "At this moment in time the required data tha…
## $ example_uses             <chr> "We use BIM (Building information Model) plat…
## $ data_impacts             <chr> "Cost efficiency', 'New or improved product/s…
## $ requested_data           <chr> "Australian Open BIM standards, Once this dat…
## $ data_sources             <chr> "New South Wales Government (NSW Land and Pro…
```

```
## Downloading the .xls data from web
## readxl and gdata are the two packages we just used to read the excel data. gdata
can read the excel files from internet as well.

url_xls <- "https://d28rz98at9flks.cloudfront.net/83173/ElectricityTransmissionSubs
tations_v2.xls"
download.file(url_xls, destfile = "electric.xls")
dat_electric <- read_excel(path = "electric.xls")
```

Writing data into files

```
write.csv(dat_electric, "dat_electric.csv", row.names = F, sep = ",")
```

```
## Warning in write.csv(dat_electric, "dat_electric.csv", row.names = F, sep =
## ","): attempt to set 'sep' ignored
```

```
write.table(dat_electric, "dat_electric.txt", row.names = F)
write_tsv(dat_electric, "dat_electric.txt")
```

Data manipulation with dplyr

```
## Working on R dataset

data()

data("diamonds")
force(diamonds)
```

```
## # A tibble: 53,940 × 10
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
## # … with 53,930 more rows
```

Transforming Data with dplyr

```
## There are verbs that can be use to manipulate both, the rows and columns of a da
taframe. These verbs include select, filter, arrange, and mutate.

## Working with columns
### Extracting the columns
glimpse(diamonds)
```

```
## Rows: 53,940
## Columns: 10
## $ carat   <dbl> 0.23, 0.21, 0.23, 0.29, 0.31, 0.24, 0.24, 0.26, 0.22, 0.23, 0.…
## $ cut     <ord> Ideal, Premium, Good, Premium, Good, Very Good, Very Good, Ver…
## $ color   <ord> E, E, E, I, J, J, I, H, E, H, J, J, F, J, E, E, I, J, J, J, I,…
## $ clarity <ord> SI2, SI1, VS1, VS2, SI2, VVS2, VVS1, SI1, VS2, VS1, SI1, VS1, …
## $ depth   <dbl> 61.5, 59.8, 56.9, 62.4, 63.3, 62.8, 62.3, 61.9, 65.1, 59.4, 64…
## $ table   <dbl> 55, 61, 65, 58, 58, 57, 57, 55, 61, 61, 55, 56, 61, 54, 62, 58…
## $ price   <int> 326, 326, 327, 334, 335, 336, 336, 337, 337, 338, 339, 340, 34…
## $ x       <dbl> 3.95, 3.89, 4.05, 4.20, 4.34, 3.94, 3.95, 4.07, 3.87, 4.00, 4.…
## $ y       <dbl> 3.98, 3.84, 4.07, 4.23, 4.35, 3.96, 3.98, 4.11, 3.78, 4.05, 4.…
## $ z       <dbl> 2.43, 2.31, 2.31, 2.63, 2.75, 2.48, 2.47, 2.53, 2.49, 2.39, 2.…
```

```
summary(diamonds)
```

```
##      carat               cut        color        clarity          depth      
##  Min.   :0.2000   Fair      : 1610   D: 6775   SI1     :13065   Min.   :43.00  
##  1st Qu.:0.4000   Good      : 4906   E: 9797   VS2     :12258   1st Qu.:61.00  
##  Median :0.7000   Very Good:12082   F: 9542   SI2     : 9194   Median :61.80  
##  Mean   :0.7979   Premium   :13791   G:11292   VS1     : 8171   Mean   :61.75  
##  3rd Qu.:1.0400   Ideal     :21551   H: 8304   VVS2    : 5066   3rd Qu.:62.50  
##  Max.   :5.0100                     I: 5422   VVS1    : 3655   Max.   :79.00  
##                                     J: 2808   (Other): 2531                  
##      table           price             x                y         
##  Min.   :43.00   Min.   :  326   Min.   : 0.000   Min.   : 0.000  
##  1st Qu.:56.00   1st Qu.:  950   1st Qu.: 4.710   1st Qu.: 4.720  
##  Median :57.00   Median : 2401   Median : 5.700   Median : 5.710  
##  Mean   :57.46   Mean   : 3933   Mean   : 5.731   Mean   : 5.735  
##  3rd Qu.:59.00   3rd Qu.: 5324   3rd Qu.: 6.540   3rd Qu.: 6.540  
##  Max.   :95.00   Max.   :18823   Max.   :10.740   Max.   :58.900  
##                                                                   
##        z         
##  Min.   : 0.000  
##  1st Qu.: 2.910  
##  Median : 3.530  
##  Mean   : 3.539  
##  3rd Qu.: 4.040  
##  Max.   :31.800  
##
```

```
?select ## Keep or drop columns using their names and types

dat <-diamonds %>%
  select(c(3:6)) ## either using column number with range

dat <- diamonds %>%
  select(c(color, clarity, depth, table)) ## or using column names

dat <- diamonds %>%
  select(color:table) ## or using column names with range

dat <- diamonds %>%
  select(ends_with("color")) ## select all columns which ends with the string 'colo
r'

dat <- diamonds %>%
  select(!(color:table)) ## select all columns except some of them. You can use col
umn number or names or range.

## There are other selection helpers that can be used within select() to extract di
fferent columns.

### Mutating the columns
?mutate ##Create, modify, and delete columns

dat <- diamonds %>%
  mutate("Table.Price" = table*price) ## added a new column in the dataset
dat
```

```
## # A tibble: 53,940 × 11
##    carat cut       color clarity depth table price     x     y     z Table.Price
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>       <dbl>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43       17930
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31       19886
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31       21255
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63       19372
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75       19430
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48       19152
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47       19152
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53       18535
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49       20557
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39       20618
## # … with 53,930 more rows
```

```
dat <- diamonds %>%
  mutate("Color.Cut" = paste0(color,".",cut)) ## added a new column in the dataset
dat
```

```
## # A tibble: 53,940 × 11
##    carat cut       color clarity depth table price     x     y     z Color.Cut
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl> <chr>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43 E.Ideal
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31 E.Premium
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31 E.Good
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63 I.Premium
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75 J.Good
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48 J.Very Good
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47 I.Very Good
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53 H.Very Good
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49 E.Fair
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39 H.Very Good
## # … with 53,930 more rows
```

```
dat <- diamonds %>%
  mutate("Table.Price" = table*price,
         .keep = "none") ## only kept the new column and didn't include the previou
s one
dat
```

```
## # A tibble: 53,940 × 1
##    Table.Price
##          <dbl>
##  1       17930
##  2       19886
##  3       21255
##  4       19372
##  5       19430
##  6       19152
##  7       19152
##  8       18535
##  9       20557
## 10       20618
## # … with 53,930 more rows
```

```
### Working with rows

## Filtering the rows
?dplyr::filter ## Keep rows that match a condition

summary(diamonds)
```

```
##      carat               cut        color        clarity          depth
##  Min.   :0.2000   Fair      : 1610   D: 6775   SI1     :13065   Min.   :43.00
##  1st Qu.:0.4000   Good      : 4906   E: 9797   VS2     :12258   1st Qu.:61.00
##  Median :0.7000   Very Good:12082   F: 9542   SI2     : 9194   Median :61.80
##  Mean   :0.7979   Premium   :13791   G:11292   VS1     : 8171   Mean   :61.75
##  3rd Qu.:1.0400   Ideal     :21551   H: 8304   VVS2    : 5066   3rd Qu.:62.50
##  Max.   :5.0100                     I: 5422   VVS1    : 3655   Max.   :79.00
##                                     J: 2808   (Other): 2531
##      table           price             x                y
##  Min.   :43.00   Min.   :  326   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:56.00   1st Qu.:  950   1st Qu.: 4.710   1st Qu.: 4.720
##  Median :57.00   Median : 2401   Median : 5.700   Median : 5.710
##  Mean   :57.46   Mean   : 3933   Mean   : 5.731   Mean   : 5.735
##  3rd Qu.:59.00   3rd Qu.: 5324   3rd Qu.: 6.540   3rd Qu.: 6.540
##  Max.   :95.00   Max.   :18823   Max.   :10.740   Max.   :58.900
##
##        z
##  Min.   : 0.000
##  1st Qu.: 2.910
##  Median : 3.530
##  Mean   : 3.539
##  3rd Qu.: 4.040
##  Max.   :31.800
##
```

```
dat <- diamonds %>%
  filter(carat >1) ## selecting only those diamonds which are greater than 1 carat

dat <- diamonds %>%
  filter(cut == "Ideal") ## selecting only those diamonds which have ideal cut

dat <- diamonds %>%
  filter(carat >1) %>%
  filter(cut == "Ideal") ## selecting only those diamonds which are greater than 1
carat and have ideal cut

summary(dat)
```

```
##      carat               cut        color       clarity         depth
##  Min.   :1.010   Fair      :   0   D: 358   SI1     :1389   Min.   :43.00
##  1st Qu.:1.060   Good      :   0   E: 531   SI2     :1298   1st Qu.:61.30
##  Median :1.200   Very Good:   0   F: 855   VS2     :1199   Median :61.80
##  Mean   :1.315   Premium   :   0   G:1418   VS1     : 854   Mean   :61.73
##  3rd Qu.:1.510   Ideal     :5662   H:1145   VVS2    : 496   3rd Qu.:62.30
##  Max.   :3.500                    I: 876   VVS1    : 209   Max.   :66.70
##                                   J: 479   (Other): 217
##      table           price             x              y               z
##  Min.   :43.00   Min.   : 2416   Min.   :0.00   Min.   :0.000   Min.   :0.000
##  1st Qu.:55.00   1st Qu.: 5520   1st Qu.:6.57   1st Qu.:6.580   1st Qu.:4.050
##  Median :56.00   Median : 7655   Median :6.81   Median :6.820   Median :4.210
##  Mean   :56.26   Mean   : 8674   Mean   :6.99   Mean   :6.994   Mean   :4.314
##  3rd Qu.:57.00   3rd Qu.:10994   3rd Qu.:7.36   3rd Qu.:7.360   3rd Qu.:4.550
##  Max.   :62.00   Max.   :18806   Max.   :9.65   Max.   :9.590   Max.   :6.030
##
```

```
dat <- diamonds %>%
  filter(!carat >1) ## selecting only those diamonds which are not greater than 1 c
arat

summary(dat)
```

```
##      carat               cut        color       clarity         depth
##  Min.   :0.2000   Fair      :  959   D:5454   VS2     :8575   Min.   :43.00
##  1st Qu.:0.3300   Good      : 3327   E:7905   SI1     :8483   1st Qu.:61.10
##  Median :0.5000   Very Good: 8201   F:6941   VS1     :5935   Median :61.80
##  Mean   :0.5312   Premium   : 8062   G:7573   SI2     :4297   Mean   :61.74
##  3rd Qu.:0.7100   Ideal     :15889   H:4667   VVS2    :4090   3rd Qu.:62.50
##  Max.   :1.0000                     I:2742   VVS1    :3259   Max.   :79.00
##                                     J:1156   (Other):1799
##      table           price             x               y
##  Min.   :44.00   Min.   :  326   Min.   :0.000   Min.   : 0.000
##  1st Qu.:56.00   1st Qu.:  775   1st Qu.:4.460   1st Qu.: 4.470
##  Median :57.00   Median : 1262   Median :5.070   Median : 5.070
##  Mean   :57.27   Mean   : 1787   Mean   :5.104   Mean   : 5.112
##  3rd Qu.:59.00   3rd Qu.: 2470   3rd Qu.:5.710   3rd Qu.: 5.720
##  Max.   :79.00   Max.   :16469   Max.   :6.820   Max.   :31.800
##
##        z
##  Min.   : 0.000
##  1st Qu.: 2.750
##  Median : 3.130
##  Mean   : 3.154
##  3rd Qu.: 3.530
##  Max.   :31.800
##
```

```
## you can use other conditions in the filter() verb.

## Arranging the rows
dat <- diamonds %>%
  arrange(carat) ## arrange the column 'carat' in ascending order and arranged the
whole data as well

dat <- diamonds %>%
  arrange(desc(carat)) ## arrange the column 'carat' in descending order and arrang
ed the whole data as well

dat <- diamonds %>%
  arrange(carat, cut) ## arrange the column 'carat' and then cut in ascending order
and arranged the whole data as well

dat
```

```
## # A tibble: 53,940 × 10
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1   0.2 Very Good E     VS2      63.4    59   367  3.74  3.71  2.36
##  2   0.2 Premium   E     SI2      60.2    62   345  3.79  3.75  2.27
##  3   0.2 Premium   E     VS2      59.8    62   367  3.79  3.77  2.26
##  4   0.2 Premium   E     VS2      59      60   367  3.81  3.78  2.24
##  5   0.2 Premium   E     VS2      61.1    59   367  3.81  3.78  2.32
##  6   0.2 Premium   E     VS2      59.7    62   367  3.84  3.8   2.28
##  7   0.2 Premium   F     VS2      62.6    59   367  3.73  3.71  2.33
##  8   0.2 Premium   D     VS2      62.3    60   367  3.73  3.68  2.31
##  9   0.2 Premium   D     VS2      61.7    60   367  3.77  3.72  2.31
## 10   0.2 Ideal     E     VS2      59.7    55   367  3.86  3.84  2.3
## # … with 53,930 more rows
```

```
## Question for students
## From the diamonds dataset, do the following together
##  1. select the columns 'carat', 'color', 'clarity', 'price'
##  2. add a new variable 'Price.in.Cents' showing the price of each diamond in cen
ts
##  3. filter for diamonds with a size of at least 1.5 carat
##  4. arrange diamonds in descending order of their price in cents
```

Aggregating Data

```
## how to aggregate your data to make it more interpretable, including count, group
_by, summarize, and top_n

glimpse(diamonds)
```

```
## Rows: 53,940
## Columns: 10
## $ carat   <dbl> 0.23, 0.21, 0.23, 0.29, 0.31, 0.24, 0.24, 0.26, 0.22, 0.23, 0.…
## $ cut     <ord> Ideal, Premium, Good, Premium, Good, Very Good, Very Good, Ver…
## $ color   <ord> E, E, E, I, J, J, I, H, E, H, J, J, F, J, E, E, I, J, J, J, I,…
## $ clarity <ord> SI2, SI1, VS1, VS2, SI2, VVS2, VVS1, SI1, VS2, VS1, SI1, VS1, …
## $ depth   <dbl> 61.5, 59.8, 56.9, 62.4, 63.3, 62.8, 62.3, 61.9, 65.1, 59.4, 64…
## $ table   <dbl> 55, 61, 65, 58, 58, 57, 57, 55, 61, 61, 55, 56, 61, 54, 62, 58…
## $ price   <int> 326, 326, 327, 334, 335, 336, 336, 337, 337, 338, 339, 340, 34…
## $ x       <dbl> 3.95, 3.89, 4.05, 4.20, 4.34, 3.94, 3.95, 4.07, 3.87, 4.00, 4.…
## $ y       <dbl> 3.98, 3.84, 4.07, 4.23, 4.35, 3.96, 3.98, 4.11, 3.78, 4.05, 4.…
## $ z       <dbl> 2.43, 2.31, 2.31, 2.63, 2.75, 2.48, 2.47, 2.53, 2.49, 2.39, 2.…
```

```
## using count()
?count ##Count the observations in each group

diamonds %>%
  count(color) ## count the number of diamonds with each color
```

```
## # A tibble: 7 × 2
##   color     n
##   <ord> <int>
## 1 D      6775
## 2 E      9797
## 3 F      9542
## 4 G     11292
## 5 H      8304
## 6 I      5422
## 7 J      2808
```

```
diamonds %>%
  count(color, sort = T) ## count the number of diamonds with each color and displa
y in ascending order
```

```
## # A tibble: 7 × 2
##   color     n
##   <ord> <int>
## 1 G     11292
## 2 E      9797
## 3 F      9542
## 4 H      8304
## 5 D      6775
## 6 I      5422
## 7 J      2808
```

```
diamonds %>%
  count(color, wt = price, sort = T) ## count the number of diamonds per each color weighted by price
```

```
## # A tibble: 7 × 2
##   color        n
##   <ord>    <int>
## 1 G     45158240
## 2 H     37257301
## 3 F     35542866
## 4 E     30142944
## 5 I     27608146
## 6 D     21476439
## 7 J     14949281
```

```
## using summarise()
?summarise ## Summarise each group down to one row. it will summarise the data columns and generate the new dataframe with the results

dat <- diamonds %>%
  summarise(min.price = min(price))
dat
```

```
## # A tibble: 1 × 1
##   min.price
##       <int>
## 1       326
```

```
dat <- diamonds %>%
  summarise(min.price = min(price),
            max.depth = max(depth))
dat
```

```
## # A tibble: 1 × 2
##   min.price max.depth
##       <int>     <dbl>
## 1       326        79
```

```
## Question for students

## From the diamonds dataset, generate a dataframe showing:
##  1. minimum diamond size in carat (min.size)
##  2. average depth for diamonds (avg.depth)
##  3. sum of price of all diamonds (sum.price)


## using group_by()
?group_by ## Group by one or more variables before summarising

dat <- diamonds %>%
  summarise(min.price = min(price))
dat
```

```
## # A tibble: 1 × 1
##   min.price
##       <int>
## 1       326
```

```
dat <- diamonds %>%
  group_by(carat) %>%
  summarise(min.price = min(price))
dat
```

```
## # A tibble: 273 × 2
##    carat min.price
##    <dbl>     <int>
##  1  0.2        345
##  2  0.21       326
##  3  0.22       337
##  4  0.23       326
##  5  0.24       336
##  6  0.25       357
##  7  0.26       337
##  8  0.27       361
##  9  0.28       360
## 10  0.29       334
## # … with 263 more rows
```

```
dat <- diamonds %>%
  group_by(color) %>%
  summarise(min.price = min(price))
dat
```

```
## # A tibble: 7 × 2
##   color min.price
##   <ord>     <int>
## 1 D           357
## 2 E           326
## 3 F           342
## 4 G           354
## 5 H           337
## 6 I           334
## 7 J           335
```

```
dat <- diamonds %>%
  group_by(carat) %>%
  summarise(min.price = min(price),
            avg.depth = mean(depth))
dat
```

```
## # A tibble: 273 × 3
##    carat min.price avg.depth
##    <dbl>     <int>     <dbl>
##  1  0.2        345      61.1
##  2  0.21       326      60.5
##  3  0.22       337      61.6
##  4  0.23       326      61.4
##  5  0.24       336      61.6
##  6  0.25       357      61.6
##  7  0.26       337      61.7
##  8  0.27       361      61.6
##  9  0.28       360      61.5
## 10  0.29       334      61.4
## # … with 263 more rows
```

```
## Question for students

## From the diamonds dataset, generate a dataframe showing:
##  1. group the data by diamonds clarity
##  2. minimum diamond size in carat (min.size)
##  3. average depth for diamonds (avg.depth)
##  4. average price of diamonds (avg.price)


## using top_n()
?top_n ## Select top (or bottom) n rows (by value)

dat <- diamonds %>%
  top_n(1, carat)
dat ## returned a top row of whole data with the largest value of carat
```

```
## # A tibble: 1 × 10
##   carat cut   color clarity depth table price     x     y     z
##   <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  5.01 Fair  J     I1       65.5    59 18018  10.7  10.5  6.98
```

```
dat <- diamonds %>%
  top_n(5, carat)
dat ## returned the top 5 rows of whole data with the largest values of carat
```

```
## # A tibble: 5 × 10
##   carat cut     color clarity depth table price     x     y     z
##   <dbl> <ord>   <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  4.01 Premium I     I1       61      61 15223  10.1 10.1   6.17
## 2  4.01 Premium J     I1       62.5    62 15223  10.0  9.94  6.24
## 3  4.13 Fair    H     I1       64.8    61 17329  10    9.85  6.43
## 4  5.01 Fair    J     I1       65.5    59 18018  10.7 10.5   6.98
## 5  4.5  Fair    J     I1       65.8    58 18531  10.2 10.2   6.72
```

```
dat <- diamonds %>%
  group_by(color) %>% ## do the same thing but with diamonds of each color
  top_n(1, carat)
dat ## returned the top row of whole data with the largest value of carat within ea
ch color
```

```
## # A tibble: 7 × 10
## # Groups:   color [7]
##   carat cut     color clarity depth table price     x     y     z
##   <dbl> <ord>   <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  3.01 Premium F     I1       62.2    56  9925  9.24  9.13  5.73
## 2  3.05 Premium E     I1       60.9    58 10453  9.26  9.25  5.66
## 3  3.01 Premium G     SI2      59.8    58 14220  9.44  9.37  5.62
## 4  4.01 Premium I     I1       61      61 15223 10.1  10.1   6.17
## 5  3.4  Fair    D     I1       66.8    52 15964  9.42  9.34  6.27
## 6  4.13 Fair    H     I1       64.8    61 17329 10     9.85  6.43
## 7  5.01 Fair    J     I1       65.5    59 18018 10.7  10.5   6.98
```

Joining Data with dplyr

Joining Tables

```
## Mutating joins

## Mutating joins add columns from y to x, matching observations based on the keys.
There are four mutating joins: the inner join, and the three outer joins.
```

Inner Join

```
## using inner_join
?inner_join ## only keeps observations from x that have a matching key in y. Means keep the common observations only and exclude the unmatched rows in both datasets

colors <- read.csv("starwars_colors.csv", sep = ",", stringsAsFactors = F, check.names = F)
body <- read.csv("starwars_body.csv", sep = ",", stringsAsFactors = F, check.names = F)
species <- read.csv("starwars_species.csv", sep = ",", stringsAsFactors = F, check.names = F)
year <- read.csv("starwars_year.csv", sep = ",", stringsAsFactors = F, check.names = F)

## to join the two datasets, their should be a common column in the both datasets
glimpse(colors)
```

```
## Rows: 53
## Columns: 4
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ hair_color     <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown…
## $ skin_color     <chr> "fair", "gold", "white, blue", "white", "light", "light…
## $ eye_color      <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "bl…
```

```
glimpse(body)
```

```
## Rows: 66
## Columns: 3
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
```

```
inner <- body %>%
  inner_join(colors, by = c("character_name" = "character_name"))
head(inner)
```

```
##   character_name height mass  hair_color  skin_color eye_color
## 1 Luke Skywalker    172   77       blond        fair      blue
## 2          C-3PO    167   75        <NA>        gold    yellow
## 3          R2-D2     96   32        <NA> white, blue       red
## 4    Darth Vader    202  136        none       white    yellow
## 5    Leia Organa    150   49       brown       light     brown
## 6      Owen Lars    178  120 brown, grey       light      blue
```

```
inner <- body %>%
  inner_join(colors, by = "character_name")
head(inner)
```

```
##   character_name height mass  hair_color  skin_color eye_color
## 1 Luke Skywalker    172   77       blond        fair      blue
## 2          C-3PO    167   75        <NA>        gold    yellow
## 3          R2-D2     96   32        <NA> white, blue       red
## 4    Darth Vader    202  136        none       white    yellow
## 5    Leia Organa    150   49       brown       light     brown
## 6      Owen Lars    178  120 brown, grey       light      blue
```

```
glimpse(species)
```

```
## Rows: 69
## Columns: 3
## $ name      <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Leia Org…
## $ homeworld <chr> "Tatooine", "Tatooine", "Naboo", "Tatooine", "Alderaan", "Ta…
## $ species   <chr> "Human", "Droid", "Droid", "Human", "Human", "Human", "Human…
```

```
inner <- body %>%
  inner_join(colors, by = c("character_name" = "character_name")) %>%
  inner_join(species, by = c("character_name" = "name"))
head(inner)
```

```
##   character_name height mass  hair_color  skin_color eye_color homeworld
## 1 Luke Skywalker    172   77       blond        fair      blue  Tatooine
## 2          C-3PO    167   75        <NA>        gold    yellow  Tatooine
## 3          R2-D2     96   32        <NA> white, blue       red     Naboo
## 4    Darth Vader    202  136        none       white    yellow  Tatooine
## 5    Leia Organa    150   49       brown       light     brown  Alderaan
## 6      Owen Lars    178  120 brown, grey       light      blue  Tatooine
##   species
## 1   Human
## 2   Droid
## 3   Droid
## 4   Human
## 5   Human
## 6   Human
```

```
glimpse(year)
```

```
## Rows: 60
## Columns: 5
## $ ID         <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Leia Or…
## $ birth_year <dbl> 19.0, 112.0, 33.0, 41.9, 19.0, 52.0, 47.0, NA, 24.0, 57.0, …
## $ sex        <chr> "male", "none", "none", "male", "female", "male", "female",…
## $ gender     <chr> "masculine", "masculine", "masculine", "masculine", "femini…
## $ species    <chr> "Human", "Droid", "Droid", "Human", "Human", "Human", "Huma…
```

```
inner <- body %>%
  inner_join(colors, by = c("character_name" = "character_name")) %>%
  inner_join(species, by = c("character_name" = "name")) %>%
  inner_join(year, by = c("character_name" = "ID"))
head(inner) ## two species column
```

```
##   character_name height mass  hair_color  skin_color eye_color homeworld
## 1 Luke Skywalker    172   77       blond        fair      blue  Tatooine
## 2          C-3PO    167   75        <NA>        gold    yellow  Tatooine
## 3          R2-D2     96   32        <NA> white, blue       red     Naboo
## 4    Darth Vader    202  136        none       white    yellow  Tatooine
## 5    Leia Organa    150   49       brown       light     brown  Alderaan
## 6      Owen Lars    178  120 brown, grey       light      blue  Tatooine
##   species.x birth_year    sex    gender species.y
## 1     Human       19.0   male masculine     Human
## 2     Droid      112.0   none masculine     Droid
## 3     Droid       33.0   none masculine     Droid
## 4     Human       41.9   male masculine     Human
## 5     Human       19.0 female  feminine     Human
## 6     Human       52.0   male masculine     Human
```

```
inner <- body %>%
  inner_join(colors, by = c("character_name" = "character_name")) %>%
  inner_join(species, by = c("character_name" = "name")) %>%
  inner_join(year, by = c("character_name" = "ID"), suffix = c("_body", "_year"))
head(inner) ## two species column
```

```
##   character_name height mass  hair_color  skin_color eye_color homeworld
## 1 Luke Skywalker    172   77       blond        fair      blue  Tatooine
## 2          C-3PO    167   75        <NA>        gold    yellow  Tatooine
## 3          R2-D2     96   32        <NA> white, blue       red     Naboo
## 4    Darth Vader    202  136        none       white    yellow  Tatooine
## 5    Leia Organa    150   49       brown       light     brown  Alderaan
## 6      Owen Lars    178  120 brown, grey       light      blue  Tatooine
##   species_body birth_year    sex    gender species_year
## 1        Human       19.0   male masculine        Human
## 2        Droid      112.0   none masculine        Droid
## 3        Droid       33.0   none masculine        Droid
## 4        Human       41.9   male masculine        Human
## 5        Human       19.0 female  feminine        Human
## 6        Human       52.0   male masculine        Human
```

Outer Joins

```
## Three outer joins which keep the observation that is present in at least one of the dataframe

## using left_join
?left_join ## keeps all observations in x

glimpse(colors)
```

```
## Rows: 53
## Columns: 4
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ hair_color     <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown…
## $ skin_color     <chr> "fair", "gold", "white, blue", "white", "light", "light…
## $ eye_color      <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "bl…
```

```
glimpse(body)
```

```
## Rows: 66
## Columns: 3
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
```

```
left <- body %>%
  left_join(colors, by = c("character_name" = "character_name"))
head(left)
```

```
##   character_name height mass  hair_color  skin_color eye_color
## 1 Luke Skywalker    172   77       blond        fair      blue
## 2          C-3PO    167   75        <NA>        gold    yellow
## 3          R2-D2     96   32        <NA> white, blue       red
## 4    Darth Vader    202  136        none       white    yellow
## 5    Leia Organa    150   49       brown       light     brown
## 6      Owen Lars    178  120 brown, grey       light      blue
```

```
body %>%
  left_join(colors, by = c("character_name" = "character_name")) %>%
  filter(is.na(hair_color)) ## is.na() finding the observations that doesn't have a
match
```

```
##    character_name height mass hair_color  skin_color eye_color
## 1           C-3PO    167   75       <NA>        gold    yellow
## 2           R2-D2     96   32       <NA> white, blue       red
## 3           R5-D4     97   32       <NA>  white, red       red
## 4  Shmi Skywalker    163   NA       <NA>        <NA>      <NA>
## 5      Darth Maul    175   80       <NA>        <NA>      <NA>
## 6     Bib Fortuna    180   NA       <NA>        <NA>      <NA>
## 7     Ayla Secura    178   55       <NA>        <NA>      <NA>
## 8        Dud Bolt     94   45       <NA>        <NA>      <NA>
## 9         Gasgano    122   NA       <NA>        <NA>      <NA>
## 10 Ben Quadinaros    163   65       <NA>        <NA>      <NA>
## 11     Mace Windu    188   84       <NA>        <NA>      <NA>
## 12   Ki-Adi-Mundi    198   82       <NA>        <NA>      <NA>
## 13      Kit Fisto    196   87       <NA>        <NA>      <NA>
## 14      Eeth Koth    171   NA       <NA>        <NA>      <NA>
## 15     Adi Gallia    184   50       <NA>        <NA>      <NA>
## 16    Saesee Tiin    188   NA       <NA>        <NA>      <NA>
```

```
left <- body %>%
  left_join(colors, by = c("character_name" = "character_name")) %>%
  left_join(year, by = c("character_name" = "ID"))
head(left)
```

```
##   character_name height mass  hair_color  skin_color eye_color birth_year
## 1 Luke Skywalker    172   77       blond        fair      blue       19.0
## 2          C-3PO    167   75        <NA>        gold    yellow      112.0
## 3          R2-D2     96   32        <NA> white, blue       red       33.0
## 4    Darth Vader    202  136        none       white    yellow       41.9
## 5    Leia Organa    150   49       brown       light     brown       19.0
## 6      Owen Lars    178  120 brown, grey       light      blue       52.0
##      sex    gender species
## 1   male masculine   Human
## 2   none masculine   Droid
## 3   none masculine   Droid
## 4   male masculine   Human
## 5 female  feminine   Human
## 6   male masculine   Human
```

```
body %>%
  left_join(colors, by = c("character_name" = "character_name")) %>%
  left_join(year, by = c("character_name" = "ID")) %>%
  filter(is.na(birth_year))
```

```
##       character_name height  mass hair_color          skin_color     eye_color
## 1              R5-D4     97  32.0       <NA>          white, red           red
## 2       Roos Tarpals    224  82.0       none                grey        orange
## 3         Rugor Nass    206    NA       none               green        orange
## 4           Ric Olié    183    NA      brown                fair          blue
## 5              Watto    137    NA      black          blue, grey        yellow
## 6            Sebulba    112  40.0       none           grey, red        orange
## 7        Bib Fortuna    180    NA       <NA>                <NA>          <NA>
## 8           Dud Bolt     94  45.0       <NA>                <NA>          <NA>
## 9            Gasgano    122    NA       <NA>                <NA>          <NA>
## 10    Ben Quadinaros    163  65.0       <NA>                <NA>          <NA>
## 11         Kit Fisto    196  87.0       <NA>                <NA>          <NA>
## 12         Eeth Koth    171    NA       <NA>                <NA>          <NA>
## 13        Adi Gallia    184  50.0       <NA>                <NA>          <NA>
## 14       Saesee Tiin    188    NA       <NA>                <NA>          <NA>
## 15       Yarael Poof    264    NA       none               white        yellow
## 16        Mas Amedda    196    NA       none                blue          blue
## 17      Gregar Typho    185  85.0      black                dark         brown
## 18             Cordé    157    NA      brown               light         brown
## 19       Cliegg Lars    183    NA      brown                fair          blue
## 20 Poggle the Lesser    183  80.0       none               green        yellow
## 21   Luminara Unduli    170  56.2      black              yellow          blue
## 22     Barriss Offee    166  50.0      black              yellow          blue
## 23             Dormé    165    NA      brown               light         brown
## 24             Dooku    193  80.0      white                fair         brown
## 25        Zam Wesell    168  55.0     blonde fair, green, yellow        yellow
## 26   Dexter Jettster    198 102.0       none               brown        yellow
## 27           Lama Su    229  88.0       none                grey         black
```

```
## 28           Taun We    213    NA       none                grey         black
## 29        Jocasta Nu    167    NA      white                fair          blue
## 30     Ratts Tyerell     79  15.0       none          grey, blue       unknown
## 31            R4-P17     96    NA       none         silver, red     red, blue
## 32        Wat Tambor    193  48.0       none         green, grey       unknown
## 33          San Hill    191    NA       none                grey          gold
## 34          Shaak Ti    178  57.0       none    red, blue, white         black
## 35          Grievous    216 159.0       none        brown, white green, yellow
## 36           Tarfful    234 136.0      brown               brown          blue
## 37   Raymus Antilles    188  79.0      brown               light         brown
## 38         Sly Moore    178  48.0       none                pale         white
## 39        Tion Medon    206  80.0       none                grey         black
## 40              Finn     NA    NA      black                dark          dark
## 41               Rey     NA    NA      brown               light         hazel
## 42       Poe Dameron     NA    NA      brown               light         brown
## 43               BB8     NA    NA       none                none         black
## 44    Captain Phasma     NA    NA    unknown             unknown       unknown
## 45     Padmé Amidala    165  45.0      brown               light         brown
##    birth_year    sex    gender    species
## 1          NA   none masculine      Droid
## 2          NA   male masculine     Gungan
## 3          NA   male masculine     Gungan
## 4          NA   <NA>      <NA>       <NA>
## 5          NA   male masculine  Toydarian
## 6          NA   male masculine        Dug
## 7          NA   male masculine    Twi'lek
## 8          NA   male masculine Vulptereen
## 9          NA   male masculine      Xexto
## 10         NA   male masculine      Toong
## 11         NA   male masculine   Nautolan
## 12         NA   male masculine     Zabrak
## 13         NA female  feminine Tholothian
## 14         NA   male masculine   Iktotchi
## 15         NA   male masculine   Quermian
## 16         NA   <NA>      <NA>       <NA>
## 17         NA   <NA>      <NA>       <NA>
## 18         NA   <NA>      <NA>       <NA>
## 19         NA   <NA>      <NA>       <NA>
## 20         NA   <NA>      <NA>       <NA>
## 21         NA   <NA>      <NA>       <NA>
## 22         NA   <NA>      <NA>       <NA>
## 23         NA   <NA>      <NA>       <NA>
## 24         NA   <NA>      <NA>       <NA>
## 25         NA female  feminine   Clawdite
## 26         NA   male masculine   Besalisk
## 27         NA   male masculine   Kaminoan
## 28         NA   <NA>      <NA>       <NA>
## 29         NA   <NA>      <NA>       <NA>
## 30         NA   <NA>      <NA>       <NA>
## 31         NA   <NA>      <NA>       <NA>
## 32         NA   <NA>      <NA>       <NA>
```

```
## 33          NA   <NA>      <NA>       <NA>
## 34          NA   <NA>      <NA>       <NA>
## 35          NA   <NA>      <NA>       <NA>
## 36          NA   <NA>      <NA>       <NA>
## 37          NA   <NA>      <NA>       <NA>
## 38          NA   <NA>      <NA>       <NA>
## 39          NA   <NA>      <NA>       <NA>
## 40          NA   <NA>      <NA>       <NA>
## 41          NA   <NA>      <NA>       <NA>
## 42          NA   <NA>      <NA>       <NA>
## 43          NA   <NA>      <NA>       <NA>
## 44          NA   <NA>      <NA>       <NA>
## 45          NA   <NA>      <NA>       <NA>
```

```
## using right_join
?right_join ## keeps all observations in y

glimpse(colors)
```

```
## Rows: 53
## Columns: 4
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ hair_color     <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown…
## $ skin_color     <chr> "fair", "gold", "white, blue", "white", "light", "light…
## $ eye_color      <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "bl…
```

```
glimpse(body)
```

```
## Rows: 66
## Columns: 3
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
```

```
right <- body %>%
  right_join(colors, by = c("character_name" = "character_name"))
head(right)
```

```
##   character_name height mass  hair_color  skin_color eye_color
## 1 Luke Skywalker    172   77       blond        fair      blue
## 2          C-3PO    167   75        <NA>        gold    yellow
## 3          R2-D2     96   32        <NA> white, blue       red
## 4    Darth Vader    202  136        none       white    yellow
## 5    Leia Organa    150   49       brown       light     brown
## 6      Owen Lars    178  120 brown, grey       light      blue
```

```
right <- body %>%
  right_join(colors, by = c("character_name" = "character_name")) %>%
  right_join(year, by = c("character_name" = "ID"))
head(right)
```

```
##   character_name height mass  hair_color  skin_color eye_color birth_year
## 1 Luke Skywalker    172   77       blond        fair      blue       19.0
## 2          C-3PO    167   75        <NA>        gold    yellow      112.0
## 3          R2-D2     96   32        <NA> white, blue       red       33.0
## 4    Darth Vader    202  136        none       white    yellow       41.9
## 5    Leia Organa    150   49       brown       light     brown       19.0
## 6      Owen Lars    178  120 brown, grey       light      blue       52.0
##      sex    gender species
## 1   male masculine   Human
## 2   none masculine   Droid
## 3   none masculine   Droid
## 4   male masculine   Human
## 5 female  feminine   Human
## 6   male masculine   Human
```

```
body %>%
  right_join(colors, by = c("character_name" = "character_name")) %>%
  right_join(year, by = c("character_name" = "ID")) %>%
  filter(is.na(birth_year))
```

```
##      character_name height mass hair_color          skin_color eye_color
## 1             R5-D4     97   32       <NA>          white, red       red
## 2       Roos Tarpals    224   82       none                grey    orange
## 3         Rugor Nass    206   NA       none               green    orange
## 4           Ric Olié    183   NA      brown                fair      blue
## 5              Watto    137   NA      black          blue, grey    yellow
## 6            Sebulba    112   40       none           grey, red    orange
## 7        Yarael Poof    264   NA       none               white    yellow
## 8         Zam Wesell    168   55     blonde fair, green, yellow    yellow
## 9    Dexter Jettster    198  102       none               brown    yellow
## 10           Lama Su    229   88       none                grey     black
## 11 Jek Tono Porkins     NA   NA       <NA>                <NA>      <NA>
## 12     Arvel Crynyd     NA   NA       <NA>                <NA>      <NA>
## 13         Nien Nunb    NA   NA       <NA>                <NA>      <NA>
## 14       Nute Gunray    NA   NA       <NA>                <NA>      <NA>
## 15       Bib Fortuna    NA   NA       <NA>                <NA>      <NA>
## 16          Dud Bolt    NA   NA       <NA>                <NA>      <NA>
## 17           Gasgano    NA   NA       <NA>                <NA>      <NA>
## 18   Ben Quadinaros     NA   NA       <NA>                <NA>      <NA>
## 19         Kit Fisto    NA   NA       <NA>                <NA>      <NA>
## 20         Eeth Koth    NA   NA       <NA>                <NA>      <NA>
## 21        Adi Gallia    NA   NA       <NA>                <NA>      <NA>
## 22       Saesee Tiin    NA   NA       <NA>                <NA>      <NA>
##    birth_year    sex    gender    species
## 1          NA   none masculine      Droid
## 2          NA   male masculine     Gungan
## 3          NA   male masculine     Gungan
## 4          NA   <NA>      <NA>       <NA>
## 5          NA   male masculine  Toydarian
## 6          NA   male masculine        Dug
## 7          NA   male masculine   Quermian
## 8          NA female  feminine   Clawdite
## 9          NA   male masculine   Besalisk
## 10         NA   male masculine   Kaminoan
## 11         NA   male masculine      Human
## 12         NA   male masculine      Human
## 13         NA   male masculine  Sullustan
## 14         NA   male masculine  Neimodian
## 15         NA   male masculine    Twi'lek
## 16         NA   male masculine Vulptereen
## 17         NA   male masculine      Xexto
## 18         NA   male masculine      Toong
## 19         NA   male masculine   Nautolan
## 20         NA   male masculine     Zabrak
## 21         NA female  feminine Tholothian
## 22         NA   male masculine   Iktotchi
```

```
body %>%
  right_join(colors, by = c("character_name" = "character_name")) %>%
  right_join(year, by = c("character_name" = "ID")) %>%
  replace_na(list(hair_color = "Not available", skin_color = "Unknown"))
```

```
##           character_name height mass    hair_color          skin_color
## 1         Luke Skywalker    172   77         blond                fair
## 2                  C-3PO    167   75 Not available                gold
## 3                  R2-D2     96   32 Not available         white, blue
## 4            Darth Vader    202  136          none               white
## 5            Leia Organa    150   49         brown               light
## 6              Owen Lars    178  120   brown, grey               light
## 7     Beru Whitesun lars    165   75         brown               light
## 8                  R5-D4     97   32 Not available          white, red
## 9      Biggs Darklighter    183   84         black               light
## 10        Obi-Wan Kenobi    182   77 auburn, white                fair
## 11      Anakin Skywalker    188   84         blond                fair
## 12        Wilhuff Tarkin    180   NA  auburn, grey                fair
## 13             Chewbacca    228  112         brown             unknown
## 14          Roos Tarpals    224   82          none                grey
## 15            Rugor Nass    206   NA          none               green
## 16              Ric Olié    183   NA         brown                fair
## 17                 Watto    137   NA         black          blue, grey
## 18               Sebulba    112   40          none           grey, red
## 19         Quarsh Panaka    183   NA         black                dark
## 20           Yarael Poof    264   NA          none               white
## 21              Plo Koon    188   80          none              orange
## 22   Bail Prestor Organa    191   NA         black                 tan
## 23            Jango Fett    183   79         black                 tan
## 24            Zam Wesell    168   55        blonde fair, green, yellow
## 25       Dexter Jettster    198  102          none               brown
## 26               Lama Su    229   88          none                grey
## 27             Han Solo     NA   NA Not available             Unknown
## 28               Greedo     NA   NA Not available             Unknown
## 29 Jabba Desilijic Tiure    NA   NA Not available             Unknown
## 30        Wedge Antilles     NA   NA Not available             Unknown
## 31      Jek Tono Porkins     NA   NA Not available             Unknown
## 32                  Yoda     NA   NA Not available             Unknown
## 33             Palpatine     NA   NA Not available             Unknown
## 34             Boba Fett     NA   NA Not available             Unknown
## 35                 IG-88     NA   NA Not available             Unknown
## 36                 Bossk     NA   NA Not available             Unknown
## 37      Lando Calrissian     NA   NA Not available             Unknown
## 38                 Lobot     NA   NA Not available             Unknown
## 39                Ackbar     NA   NA Not available             Unknown
## 40            Mon Mothma     NA   NA Not available             Unknown
## 41          Arvel Crynyd     NA   NA Not available             Unknown
## 42 Wicket Systri Warrick     NA   NA Not available             Unknown
## 43             Nien Nunb     NA   NA Not available             Unknown
```

```
## 44          Qui-Gon Jinn     NA   NA Not available            Unknown
## 45            Nute Gunray     NA   NA Not available            Unknown
## 46         Finis Valorum     NA   NA Not available            Unknown
## 47         Jar Jar Binks     NA   NA Not available            Unknown
## 48        Shmi Skywalker     NA   NA Not available            Unknown
## 49            Darth Maul     NA   NA Not available            Unknown
## 50           Bib Fortuna     NA   NA Not available            Unknown
## 51           Ayla Secura     NA   NA Not available            Unknown
## 52              Dud Bolt     NA   NA Not available            Unknown
## 53               Gasgano     NA   NA Not available            Unknown
## 54        Ben Quadinaros     NA   NA Not available            Unknown
## 55            Mace Windu     NA   NA Not available            Unknown
## 56          Ki-Adi-Mundi     NA   NA Not available            Unknown
## 57             Kit Fisto     NA   NA Not available            Unknown
## 58             Eeth Koth     NA   NA Not available            Unknown
## 59            Adi Gallia     NA   NA Not available            Unknown
## 60           Saesee Tiin     NA   NA Not available            Unknown
##    eye_color birth_year            sex    gender        species
## 1       blue       19.0           male masculine          Human
## 2     yellow      112.0           none masculine          Droid
## 3        red       33.0           none masculine          Droid
## 4     yellow       41.9           male masculine          Human
## 5      brown       19.0         female  feminine          Human
## 6       blue       52.0           male masculine          Human
## 7       blue       47.0         female  feminine          Human
## 8        red         NA           none masculine          Droid
## 9      brown       24.0           male masculine          Human
## 10 blue-gray       57.0           male masculine          Human
## 11      blue       41.9           male masculine          Human
## 12      blue       64.0           male masculine          Human
## 13      blue      200.0           male masculine        Wookiee
## 14    orange         NA           male masculine         Gungan
## 15    orange         NA           male masculine         Gungan
## 16      blue         NA           <NA>      <NA>           <NA>
## 17    yellow         NA           male masculine      Toydarian
## 18    orange         NA           male masculine            Dug
## 19     brown       62.0           <NA>      <NA>           <NA>
## 20    yellow         NA           male masculine       Quermian
## 21     black       22.0           male masculine        Kel Dor
## 22     brown       67.0           male masculine          Human
## 23     brown       66.0           male masculine          Human
## 24    yellow         NA         female  feminine       Clawdite
## 25    yellow         NA           male masculine       Besalisk
## 26     black         NA           male masculine       Kaminoan
## 27      <NA>       29.0           male masculine          Human
## 28      <NA>       44.0           male masculine         Rodian
## 29      <NA>      600.0 hermaphroditic masculine           Hutt
## 30      <NA>       21.0           male masculine          Human
## 31      <NA>         NA           male masculine          Human
## 32      <NA>      896.0           male masculine Yoda's species
## 33      <NA>       82.0           male masculine          Human
```

```
## 34      <NA>       31.5           male masculine          Human
## 35      <NA>       15.0           none masculine          Droid
## 36      <NA>       53.0           male masculine     Trandoshan
## 37      <NA>       31.0           male masculine          Human
## 38      <NA>       37.0           male masculine          Human
## 39      <NA>       41.0           male masculine   Mon Calamari
## 40      <NA>       48.0         female  feminine          Human
## 41      <NA>         NA           male masculine          Human
## 42      <NA>        8.0           male masculine           Ewok
## 43      <NA>         NA           male masculine      Sullustan
## 44      <NA>       92.0           male masculine          Human
## 45      <NA>         NA           male masculine      Neimodian
## 46      <NA>       91.0           male masculine          Human
## 47      <NA>       52.0           male masculine         Gungan
## 48      <NA>       72.0         female  feminine          Human
## 49      <NA>       54.0           male masculine         Zabrak
## 50      <NA>         NA           male masculine        Twi'lek
## 51      <NA>       48.0         female  feminine        Twi'lek
## 52      <NA>         NA           male masculine     Vulptereen
## 53      <NA>         NA           male masculine          Xexto
## 54      <NA>         NA           male masculine          Toong
## 55      <NA>       72.0           male masculine          Human
## 56      <NA>       92.0           male masculine         Cerean
## 57      <NA>         NA           male masculine       Nautolan
## 58      <NA>         NA           male masculine         Zabrak
## 59      <NA>         NA         female  feminine     Tholothian
## 60      <NA>         NA           male masculine       Iktotchi
```

Full, Semi, and Anti Joins

```
## Three more joining verbs: full-join, semi-join, and anti-join.

## using full_join
?full_join ## keeps all observations in x and y

glimpse(colors)
```

```
## Rows: 53
## Columns: 4
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ hair_color     <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown…
## $ skin_color     <chr> "fair", "gold", "white, blue", "white", "light", "light…
## $ eye_color      <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "bl…
```

```
glimpse(body)
```

```
## Rows: 66
## Columns: 3
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
```

```
full <- body %>%
  full_join(colors, by = c("character_name" = "character_name"))
head(full)
```

```
##   character_name height mass  hair_color  skin_color eye_color
## 1 Luke Skywalker    172   77       blond        fair      blue
## 2          C-3PO    167   75        <NA>        gold    yellow
## 3          R2-D2     96   32        <NA> white, blue       red
## 4    Darth Vader    202  136        none       white    yellow
## 5    Leia Organa    150   49       brown       light     brown
## 6      Owen Lars    178  120 brown, grey       light      blue
```

```
body %>%
  full_join(colors, by = c("character_name" = "character_name")) %>%
  filter(is.na(hair_color)) ## is.na() finding the observations that doesn't have a
match
```

```
##    character_name height mass hair_color  skin_color eye_color
## 1           C-3PO    167   75       <NA>        gold    yellow
## 2           R2-D2     96   32       <NA> white, blue       red
## 3           R5-D4     97   32       <NA>  white, red       red
## 4  Shmi Skywalker    163   NA       <NA>        <NA>      <NA>
## 5      Darth Maul    175   80       <NA>        <NA>      <NA>
## 6     Bib Fortuna    180   NA       <NA>        <NA>      <NA>
## 7     Ayla Secura    178   55       <NA>        <NA>      <NA>
## 8        Dud Bolt     94   45       <NA>        <NA>      <NA>
## 9         Gasgano    122   NA       <NA>        <NA>      <NA>
## 10 Ben Quadinaros    163   65       <NA>        <NA>      <NA>
## 11     Mace Windu    188   84       <NA>        <NA>      <NA>
## 12   Ki-Adi-Mundi    198   82       <NA>        <NA>      <NA>
## 13      Kit Fisto    196   87       <NA>        <NA>      <NA>
## 14      Eeth Koth    171   NA       <NA>        <NA>      <NA>
## 15     Adi Gallia    184   50       <NA>        <NA>      <NA>
## 16    Saesee Tiin    188   NA       <NA>        <NA>      <NA>
```

```
full <- body %>%
  full_join(colors, by = c("character_name" = "character_name")) %>%
  full_join(year, by = c("character_name" = "ID"))
head(full)
```

```
##   character_name height mass  hair_color  skin_color eye_color birth_year
## 1 Luke Skywalker    172   77       blond        fair      blue       19.0
## 2          C-3PO    167   75        <NA>        gold    yellow      112.0
## 3          R2-D2     96   32        <NA> white, blue       red       33.0
## 4    Darth Vader    202  136        none       white    yellow       41.9
## 5    Leia Organa    150   49       brown       light     brown       19.0
## 6      Owen Lars    178  120 brown, grey       light      blue       52.0
##      sex    gender species
## 1   male masculine   Human
## 2   none masculine   Droid
## 3   none masculine   Droid
## 4   male masculine   Human
## 5 female  feminine   Human
## 6   male masculine   Human
```

```
body %>%
  full_join(colors, by = c("character_name" = "character_name")) %>%
  full_join(year, by = c("character_name" = "ID")) %>%
  replace_na(list(hair_color = "Not available", skin_color = "Unknown"))
```

```
##           character_name height  mass    hair_color          skin_color
## 1         Luke Skywalker    172  77.0         blond                fair
## 2                  C-3PO    167  75.0 Not available                gold
## 3                  R2-D2     96  32.0 Not available         white, blue
## 4            Darth Vader    202 136.0          none               white
## 5            Leia Organa    150  49.0         brown               light
## 6              Owen Lars    178 120.0   brown, grey               light
## 7     Beru Whitesun lars    165  75.0         brown               light
## 8                  R5-D4     97  32.0 Not available          white, red
## 9      Biggs Darklighter    183  84.0         black               light
## 10        Obi-Wan Kenobi    182  77.0 auburn, white                fair
## 11      Anakin Skywalker    188  84.0         blond                fair
## 12        Wilhuff Tarkin    180    NA  auburn, grey                fair
## 13             Chewbacca    228 112.0         brown             unknown
## 14          Roos Tarpals    224  82.0          none                grey
## 15            Rugor Nass    206    NA          none               green
## 16              Ric Olié    183    NA         brown                fair
## 17                 Watto    137    NA         black           blue, grey
## 18               Sebulba    112  40.0          none            grey, red
## 19         Quarsh Panaka    183    NA         black                dark
## 20        Shmi Skywalker    163    NA Not available             Unknown
## 21            Darth Maul    175  80.0 Not available             Unknown
## 22           Bib Fortuna    180    NA Not available             Unknown
## 23           Ayla Secura    178  55.0 Not available             Unknown
## 24              Dud Bolt     94  45.0 Not available             Unknown
## 25               Gasgano    122    NA Not available             Unknown
## 26        Ben Quadinaros    163  65.0 Not available             Unknown
## 27            Mace Windu    188  84.0 Not available             Unknown
```

```
## 28          Ki-Adi-Mundi    198  82.0 Not available             Unknown
## 29             Kit Fisto    196  87.0 Not available             Unknown
## 30             Eeth Koth    171    NA Not available             Unknown
## 31            Adi Gallia    184  50.0 Not available             Unknown
## 32           Saesee Tiin    188    NA Not available             Unknown
## 33           Yarael Poof    264    NA          none               white
## 34             Plo Koon    188  80.0          none              orange
## 35           Mas Amedda    196    NA          none                blue
## 36         Gregar Typho    185  85.0         black                dark
## 37                Cordé    157    NA         brown               light
## 38          Cliegg Lars    183    NA         brown                fair
## 39    Poggle the Lesser    183  80.0          none               green
## 40       Luminara Unduli    170  56.2         black              yellow
## 41        Barriss Offee    166  50.0         black              yellow
## 42                Dormé    165    NA         brown               light
## 43                Dooku    193  80.0         white                fair
## 44   Bail Prestor Organa    191    NA         black                 tan
## 45           Jango Fett    183  79.0         black                 tan
## 46           Zam Wesell    168  55.0        blonde fair, green, yellow
## 47      Dexter Jettster    198 102.0          none               brown
## 48              Lama Su    229  88.0          none                grey
## 49              Taun We    213    NA          none                grey
## 50          Jocasta Nu    167    NA         white                fair
## 51        Ratts Tyerell     79  15.0          none          grey, blue
## 52                R4-P17     96    NA          none         silver, red
## 53           Wat Tambor    193  48.0          none         green, grey
## 54             San Hill    191    NA          none                grey
## 55             Shaak Ti    178  57.0          none    red, blue, white
## 56             Grievous    216 159.0          none        brown, white
## 57              Tarfful    234 136.0         brown               brown
## 58      Raymus Antilles    188  79.0         brown               light
## 59            Sly Moore    178  48.0          none                pale
## 60           Tion Medon    206  80.0          none                grey
## 61                Finn     NA    NA         black                dark
## 62                 Rey     NA    NA         brown               light
## 63         Poe Dameron     NA    NA         brown               light
## 64                BB8      NA    NA          none                none
## 65       Captain Phasma     NA    NA       unknown             unknown
## 66        Padmé Amidala    165  45.0         brown               light
## 67              Han Solo     NA    NA Not available             Unknown
## 68               Greedo     NA    NA Not available             Unknown
## 69 Jabba Desilijic Tiure     NA    NA Not available             Unknown
## 70        Wedge Antilles     NA    NA Not available             Unknown
## 71     Jek Tono Porkins     NA    NA Not available             Unknown
## 72                Yoda     NA    NA Not available             Unknown
## 73            Palpatine     NA    NA Not available             Unknown
## 74            Boba Fett     NA    NA Not available             Unknown
## 75                IG-88     NA    NA Not available             Unknown
## 76                Bossk     NA    NA Not available             Unknown
## 77     Lando Calrissian     NA    NA Not available             Unknown
## 78                Lobot     NA    NA Not available             Unknown
```

```
## 79                Ackbar     NA    NA Not available            Unknown
## 80             Mon Mothma     NA    NA Not available            Unknown
## 81           Arvel Crynyd     NA    NA Not available            Unknown
## 82 Wicket Systri Warrick     NA    NA Not available            Unknown
## 83              Nien Nunb     NA    NA Not available            Unknown
## 84           Qui-Gon Jinn     NA    NA Not available            Unknown
## 85            Nute Gunray     NA    NA Not available            Unknown
## 86          Finis Valorum     NA    NA Not available            Unknown
## 87          Jar Jar Binks     NA    NA Not available            Unknown
##        eye_color birth_year           sex    gender        species
## 1           blue       19.0          male masculine          Human
## 2         yellow      112.0          none masculine          Droid
## 3            red       33.0          none masculine          Droid
## 4         yellow       41.9          male masculine          Human
## 5          brown       19.0        female  feminine          Human
## 6           blue       52.0          male masculine          Human
## 7           blue       47.0        female  feminine          Human
## 8            red         NA          none masculine          Droid
## 9          brown       24.0          male masculine          Human
## 10     blue-gray       57.0          male masculine          Human
## 11          blue       41.9          male masculine          Human
## 12          blue       64.0          male masculine          Human
## 13          blue      200.0          male masculine        Wookiee
## 14        orange         NA          male masculine         Gungan
## 15        orange         NA          male masculine         Gungan
## 16          blue         NA          <NA>      <NA>           <NA>
## 17        yellow         NA          male masculine      Toydarian
## 18        orange         NA          male masculine            Dug
## 19         brown       62.0          <NA>      <NA>           <NA>
## 20          <NA>       72.0        female  feminine          Human
## 21          <NA>       54.0          male masculine         Zabrak
## 22          <NA>         NA          male masculine        Twi'lek
## 23          <NA>       48.0        female  feminine        Twi'lek
## 24          <NA>         NA          male masculine     Vulptereen
## 25          <NA>         NA          male masculine          Xexto
## 26          <NA>         NA          male masculine          Toong
## 27          <NA>       72.0          male masculine          Human
## 28          <NA>       92.0          male masculine         Cerean
## 29          <NA>         NA          male masculine       Nautolan
## 30          <NA>         NA          male masculine         Zabrak
## 31          <NA>         NA        female  feminine     Tholothian
## 32          <NA>         NA          male masculine       Iktotchi
## 33        yellow         NA          male masculine       Quermian
## 34         black       22.0          male masculine        Kel Dor
## 35          blue         NA          <NA>      <NA>           <NA>
## 36         brown         NA          <NA>      <NA>           <NA>
## 37         brown         NA          <NA>      <NA>           <NA>
## 38          blue         NA          <NA>      <NA>           <NA>
## 39        yellow         NA          <NA>      <NA>           <NA>
## 40          blue         NA          <NA>      <NA>           <NA>
## 41          blue         NA          <NA>      <NA>           <NA>
```

```
## 42         brown          NA           <NA>      <NA>           <NA>
## 43         brown          NA           <NA>      <NA>           <NA>
## 44         brown        67.0           male masculine          Human
## 45         brown        66.0           male masculine          Human
## 46        yellow          NA         female  feminine       Clawdite
## 47        yellow          NA           male masculine       Besalisk
## 48         black          NA           male masculine       Kaminoan
## 49         black          NA           <NA>      <NA>           <NA>
## 50          blue          NA           <NA>      <NA>           <NA>
## 51       unknown          NA           <NA>      <NA>           <NA>
## 52     red, blue          NA           <NA>      <NA>           <NA>
## 53       unknown          NA           <NA>      <NA>           <NA>
## 54          gold          NA           <NA>      <NA>           <NA>
## 55         black          NA           <NA>      <NA>           <NA>
## 56 green, yellow          NA           <NA>      <NA>           <NA>
## 57          blue          NA           <NA>      <NA>           <NA>
## 58         brown          NA           <NA>      <NA>           <NA>
## 59         white          NA           <NA>      <NA>           <NA>
## 60         black          NA           <NA>      <NA>           <NA>
## 61          dark          NA           <NA>      <NA>           <NA>
## 62         hazel          NA           <NA>      <NA>           <NA>
## 63         brown          NA           <NA>      <NA>           <NA>
## 64         black          NA           <NA>      <NA>           <NA>
## 65       unknown          NA           <NA>      <NA>           <NA>
## 66         brown          NA           <NA>      <NA>           <NA>
## 67          <NA>        29.0           male masculine          Human
## 68          <NA>        44.0           male masculine         Rodian
## 69          <NA>       600.0 hermaphroditic masculine           Hutt
## 70          <NA>        21.0           male masculine          Human
## 71          <NA>          NA           male masculine          Human
## 72          <NA>       896.0           male masculine Yoda's species
## 73          <NA>        82.0           male masculine          Human
## 74          <NA>        31.5           male masculine          Human
## 75          <NA>        15.0           none masculine          Droid
## 76          <NA>        53.0           male masculine     Trandoshan
## 77          <NA>        31.0           male masculine          Human
## 78          <NA>        37.0           male masculine          Human
## 79          <NA>        41.0           male masculine   Mon Calamari
## 80          <NA>        48.0         female  feminine          Human
## 81          <NA>          NA           male masculine          Human
## 82          <NA>         8.0           male masculine           Ewok
## 83          <NA>          NA           male masculine      Sullustan
## 84          <NA>        92.0           male masculine          Human
## 85          <NA>          NA           male masculine      Neimodian
## 86          <NA>        91.0           male masculine          Human
## 87          <NA>        52.0           male masculine         Gungan
```

```
## using semi_join and anti_join
?semi_join ## return all rows from x with a match in y.
?anti_join ## return all rows from x without a match in y.

semi <- body %>%
  semi_join(colors, by = c("character_name" = "character_name"))
glimpse(semi) ## keep all the rows from x which match with y and return only x tabl
e
```

```
## Rows: 53
## Columns: 3
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
```

```
inner <- body %>%
  inner_join(colors, by = c("character_name" = "character_name"))
glimpse(inner) ## keep all the rows from x which match with y and return a joint x
and y table
```

```
## Rows: 53
## Columns: 6
## $ character_name <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Lei…
## $ height         <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 18…
## $ mass           <dbl> 77, 75, 32, 136, 49, 120, 75, 32, 84, 77, 84, NA, 112, …
## $ hair_color     <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown…
## $ skin_color     <chr> "fair", "gold", "white, blue", "white", "light", "light…
## $ eye_color      <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "bl…
```

```
anti <- body %>%
  anti_join(colors, by = c("character_name" = "character_name"))
glimpse(anti) ## keep all the rows from x which Do Not match with y and return only
x table
```

```
## Rows: 13
## Columns: 3
## $ character_name <chr> "Shmi Skywalker", "Darth Maul", "Bib Fortuna", "Ayla Se…
## $ height         <int> 163, 175, 180, 178, 94, 122, 163, 188, 198, 196, 171, 1…
## $ mass           <dbl> NA, 80, NA, 55, 45, NA, 65, 84, 82, 87, NA, 50, NA
```

```
## Question for students

## From the body dataset, generate a dataframe showing:
##  1. name of each character name
##  2. height of each character in descending order
##  3. join the species table to the body table
##  4. join the year table to the previous join and provide the appropriate suffix
##  5. replace the NAs in the gender and sex with "Not provided"
```

Data Visualisation

```
## learn the essential skills of data visualization using the ggplot2 package
```

Types of visualizations

```
## learn how to create basic line plots, bar plots, histograms, and boxplots.
```

Conclusion