Module 2: Installing Packages and Reading Data

Functions

```
sqrt(100)
```

```
[1] 10
```

```
median(c(3, 4, 5, 6, 7 ))
```

```
[1] 5
```

Packages

- Packages are collections of functions and data sets developed by the community.
- Two steps to use a package
  - installed with the install.packages function (only once)
  - imported with the library function (once per session)

```
install.packages("package_name")
library(package_name)
```

- Now Try it yourself

```
install.packages("tidyverse")
```

```
library(tidyverse)
```

```
starwars
```

What is the Tidyverse?

- The tidyverse is an opinionated collection of R packages designed for data science with similar underlying philosophy and a common syntax.

Loading data from files

```
getwd()
```

```
setwd("/Users/jacob/Downloads/Module 2")
```

Loading Data from Files

1.

2.

3.

```
wealth_data=read.csv("wealth_data.csv")
```

Data Frame vs Vector vs List

Vector

Data Frame

List

	country	year	strike.volume	unemployment
1	Australia	1951	296	1.3
2	Australia	1952	397	2.2
3	Australia	1953	360	2.5
4	Australia	1954	3	1.7
5	Australia	1955	326	1.4
6	Australia	1956	352	1.8
7	Australia	1957	195	2.3
8	Australia	1958	133	2.7
9	Australia	1959	109	2.6
10	Australia	1960	208	2.5

Data Frame

Creating a Data Frame

file type	package	function
.csv	readr	`read_csv()`
.dta (stata)	haven	`read_dta()`
.xlsx	readxl	`read_xlsx()`

Obtaining Basic Information of Data Frame

- Overview of the data
- Attributes of the data

Overview of the Data

- `view()` or `View()` - look at the table
- `glimpse()` - structure of data frame - name, type and preview of data in each column
- `summary()` - displays min, 1st quartile, median, mean, 3rd quartile and max - values for numeric attributes.
- `head()` - shows first 6 rows

```
view(wealth_data)
glimpse(wealth_data)
summary(wealth_data)
head(wealth_data)
```

Attributes of the Data

- `names()` or `colnames()` - both show the names attribute for a data frame
- `dim()` - returns the dimensions of data frame (i.e. number of rows and number of columns)
- `nrow()` - number of rows
- `ncol()` - number of columns

```
names(wealth_data)
dim(wealth_data)
nrow(wealth_data)
ncol(wealth_data)
```

Accessing Data

- By index (slicing)
- By name (columns only)
- By logical vector (criteria)

Dataset: Starwars

```
starwars
view(starwars)
glimpse(starwars)
```

	name	height	mass	hair_color	skin_color
1	Luke Skywalker	172	77	blond	fair
2	C-3PO	167	75	NA	gold
3	R2-D2	96	32	NA	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	NA	white, red
9	Biggs Darklighter	183	84	black	light
10	Obi-Wan Kenobi	182	77	auburn, white	fair

Accessing by Index

What will be returned by starwars[1, 1]?


eye_color = blue
hair_color = blond
skin_color = fair
gender = male
species = Human
height = 172 cm
birth_year = 19 BBY (Before Battle of Yavin)
films = c("Revenge of the Sith",
"Return of the Jedi",
"The Empire Strikes Back",
"A New Hope",
"The Force Awakens")

What will be returned by starwars[, 2]? What will be returned by starwars[1,]?

```
4:6
```

Use the colon operator to index just the hair color, skin color, and eye color (columns 4 to 6).

```r
starwars[c(1, 5, 7, 9), 1:5]
```

Now try to return the name (column 1) and mass (column 3) values for the first 5 character.

Accessing by Name

```
names(starwars)
```

```
starwars$species
```

Accessing by Name

Best Practice

Best practice is to address columns by name, often you will create or delete columns and the column position will change.

Accessing by Logical Vector (Criteria)

Find all characters with species being Human (with missing values)?

```
criteria = starwars$species == "Human"
starwars[criteria,]$name
```

```
starwars[starwars$species == "Human",]$name
```

Exercise: Find all characters with height greater than 170 (with missing values)?

Functions for Missing Values

- `na.omit(dataframe)` - removes the missing values in data frame
- `is.na(dataframe$colname)` - indicates which elements are missing

```
na.omit(starwars)
starwars[!is.na(starwars$species),]
```

Recap: Installing Packages and Reading Data

- install packages using `install.packages()` and load them using `library()`
  - particularly, the package `tidyverse` using `library(tidyverse)`
- Index into a dataframe by:
  - index (slicing)
  - name (columns only)
  - logical vector (criteria)