Anatomy of dplyr Functions

filter

- Filter the data frame to only show a particular set of values

Means “and then”

Must be a double equals

NewDataFrame <- DataFrame %>% filter(column == value)

Argument Name

Number – no quotes
Word - quotes

filter %in%

- Filter the data frame to only show a particular set of values in a vector

Create a vector of the values

Means “and then”

Double parentheses

```
NewDataFrame <- DataFrame %>% filter(column %in%
c(value, value))
```

Number – no quotes
Word - quotes

Argument Name

arrange

- Order the data frame by a variable's values

Means "and then"

desc: Largest first
asc: Smallest first

```
NewDataFrame <- DataFrame %>%
arrange(desc(column))
```

Argument Name

Double parentheses

select

- Choose variables (columns)

Means “and then”

Name of the columns to keep

NewDataFrame <- DataFrame %>% select(column, column)

Argument Name

summarise

- Get an aggregate value for a column

Means "and then"

Name of your new column

NewDataFrame <- DataFrame %>% summarise(NewColumnName = function(column))

Argument Name

Options include:

- median
- mean
- max
- min
- And More!

group_by

- Rolling things up by a variable
- Must be used in conjunction with summarise to get the aggregation

```
NewDataFrame <- DataFrame %>%
        group_by(column) %>%
        summarise(NewColumnName =
function(column))
```

The AMAZING power of dplyr

- Can mix and match and combine with %>% all in one new dataset!