Introduction to the Tidyverse

Ken Mwai | Alice Kamau | Mark Otiende

The tidyverse help

- https://ggplot2.tidyverse.org/
- https://www.tidyverse.org/learn/

Using packages in R

```r
install.packages('name')
```

- Downloads the files to your computer
- Do this once per computer

```r
# How to use the package
library('name')
```

- Loads the package
- Do this once per seesion

coherent system of packages for data manipulation

The tidyverse

The tidyverse

Tidyverse?

- The tidyverse is an opinionated collection of R packages designed for data science.
- All packages share an underlying design philosophy, grammar, and data structures.
- The tidyverse makes data science faster, easier and more fun

"Tidyverse package"

Task

- The tidyverse package is a shortcut for installing and loading all the key tidyverse packages

Install the `tidyverse` package

Solution

```
install.packages('tidyverse')
```

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyr")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("stringr")
install.packages("forcats")
install.packages("lubridate")
install.packages("hms")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```

Data frames and tibbles

- Data frames are the most common kind of data objects; used for rectangular data (like spreadsheets)
- Data frames: R's native data object
- Tibbles (tbl): a fancier enhanced kind of data frame
- (You really won't notice a difference in this class)

Vectors

- Vectors are a list of values of the same time (all text, or all numbers, etc.)
- **Make them with c():**

```
c(1, 4, 2, 5, 7)
```

```
## [1] 1 4 2 5 7
```

- You'll usually want to assign them to something: :::{.task} Create a vector `c(1, 4, 2, 5, 7)` and assign it `neat_numbers` object name :::

Solution

```
neat_numbers <- c(1, 4, 2, 5, 7)
```

Packages for importing data

readr	Work with plain text data	`my_data <- read_csv("file.csv")`
readxl	Work with Excel files	`my_data <- read_excel("file.xlsx")`
haven	Work with Stata, SPSS, and SAS data	`my_data <- read_stata("file.dta")`

- Hint use `read_csv` after loading `tidyverse`

Read in the `birthweight.csv file` and assign it to `bw_df`

Solution

```r
library(tidyverse)
bw_df <- read_csv('data/birthweight.csv')
```

The tidyverse: dplyr

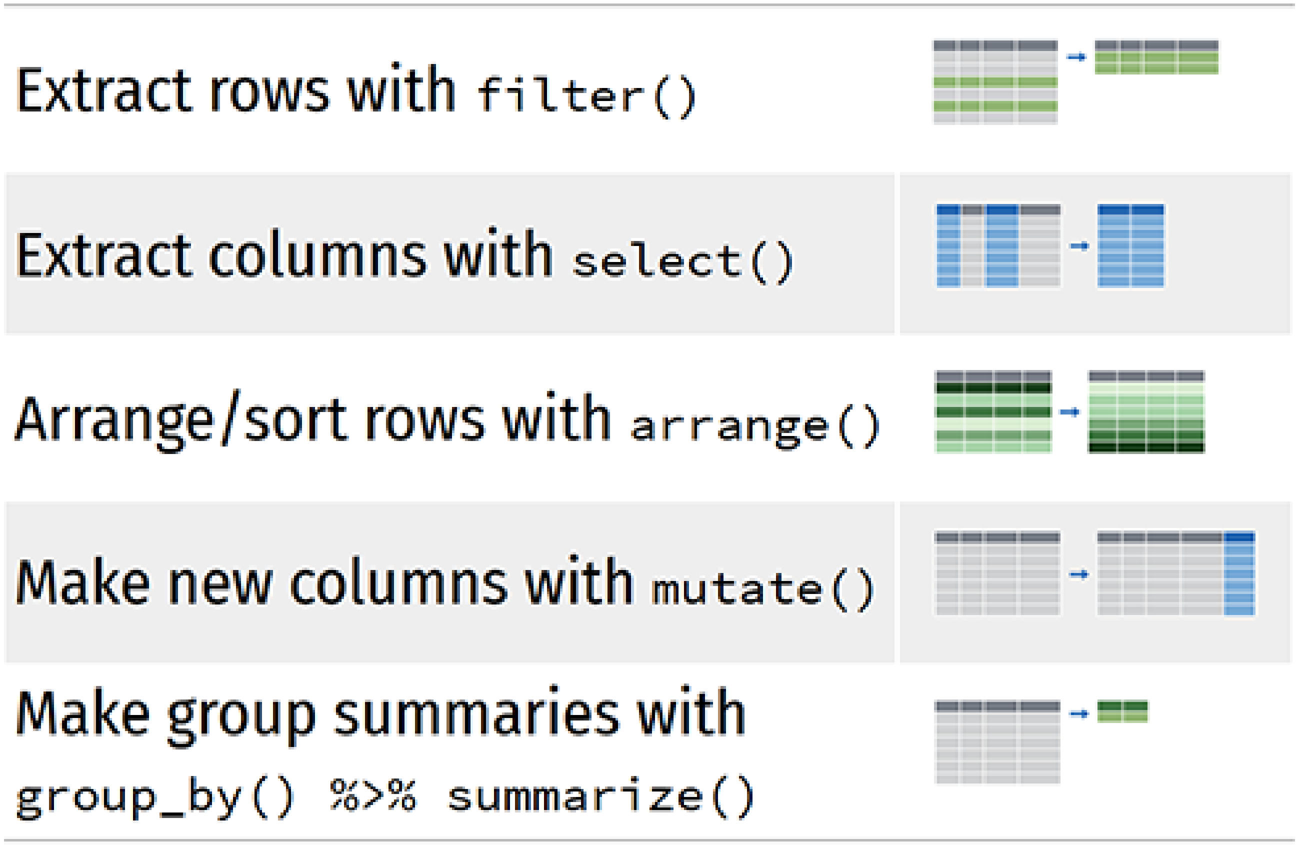
Dataset to use

- Excerpt of the `Gapminder` data on life expectancy, GDP per capita, and population by country.
- The data frame `gapminder` has 1704 rows and 6 variables
    - Country -factor with 142 levels
    - Continent - factor with 5 levels
    - Year - ranges from 1952 to 2007 in increments of 5 years
    - lifeExp - life expectancy at birth, in years
    - Pop - population
    - gdpPercap - GDP per capita (US$, inflation-adjusted)
- Task: Install and load the gapminder package

```
library(gapminder)
glimpse(gapminder)
```

```
## Rows: 1,704
## Columns: 7
## $ country   □[3m□[38;5;246m<fct>□[39m□[23m "Afghanistan", "Afghanistan", "Afghanistan", "Afghanistan", "Afghanist
## $ continent □[3m□[38;5;246m<fct>□[39m□[23m Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia
## $ year      □[3m□[38;5;246m<int>□[39m□[23m 1952, 1957, 1962, 1967, 1972, 1977, 1982, 1987, 1992, 1997, 2002, 2007
## $ lifeExp   □[3m□[38;5;246m<dbl>□[39m□[23m 28.801, 30.332, 31.997, 34.020, 36.088, 38.438, 39.854, 40.822, 41.674
## $ pop       □[3m□[38;5;246m<int>□[39m□[23m 8425333, 9240934, 10267083, 11537966, 13079460, 14880372, 12881816, 13
## $ gdpPercap □[3m□[38;5;246m<dbl>□[39m□[23m 779.4453, 820.8530, 853.1007, 836.1971, 739.9811, 786.1134, 978.0114,
## $ year_cat  □[3m□[38;5;246m<chr>□[39m□[23m "Before 1980", "Before 1980", "Before 1980", "Before 1980", "Before 19
```

dplyr: verbs for manipulating data



Extract rows with `filter()`
Extract columns with `select()`
Arrange/sort rows with `arrange()`
Make new columns with `mutate()`
Make group summaries with `group_by() %>% summarize()`

Select a subset of variables

```
select(.data = DATA, ...)
```

- `DATA` = Data frame to transform
- `...` = variables to select

dplyr: select

Our data

```
head(gapminder)
```

```
## # A tibble: 6 x 7
##   country     continent  year lifeExp      pop gdpPercap year_cat
##   <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>
## 1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980
## 2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980
## 3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980
## 4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980
## 5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980
## 6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980
```

Subset country and life expectancy and year variables only

```
select(.data = gapminder, c(country,year , lifeExp))
```

```
## # A tibble: 1,704 x 3
##    country      year lifeExp
##    <fct>       <int>   <dbl>
##  1 Afghanistan  1952    28.8
##  2 Afghanistan  1957    30.3
##  3 Afghanistan  1962    32.0
##  4 Afghanistan  1967    34.0
##  5 Afghanistan  1972    36.1
##  6 Afghanistan  1977    38.4
##  7 Afghanistan  1982    39.9
##  8 Afghanistan  1987    40.8
##  9 Afghanistan  1992    41.7
## 10 Afghanistan  1997    41.8
## # ... with 1,694 more rows
```

Task #1: Select

Use select()

- subset `country`,`year`,`gdpPercap`, and `pop` variables only
- create a new object called `population_gdp` assinged to subset data

Select() solution

```
population_gdp <- select(gapminder, c(country,year,gdpPercap,pop))
population_gdp
```

```
## # A tibble: 1,704 x 4
##    country      year gdpPercap      pop
##    <fct>       <int>     <dbl>    <int>
##  1 Afghanistan  1952      779.  8425333
##  2 Afghanistan  1957      821.  9240934
##  3 Afghanistan  1962      853. 10267083
##  4 Afghanistan  1967      836. 11537966
##  5 Afghanistan  1972      740. 13079460
##  6 Afghanistan  1977      786. 14880372
##  7 Afghanistan  1982      978. 12881816
##  8 Afghanistan  1987      852. 13867957
##  9 Afghanistan  1992      649. 16317921
## 10 Afghanistan  1997      635. 22227415
## # ... with 1,694 more rows
```

Pick rows: filter()

dplyr: filter

Extract rows that meet some sort of test

```
filter(.data = DATA, ...)
```

- `DATA` = Data frame to transform
- `...` = One or more tests
- `filter()` returns each row for which the test is TRUE

Our data

```
head(gapminder)
```

```
## # A tibble: 6 x 7
##   country     continent  year lifeExp      pop gdpPercap year_cat
##   <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>
## 1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980
## 2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980
## 3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980
## 4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980
## 5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980
## 6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980
```

Filtering only Tanzania data

- NB: We use == which tests if equal
    - One = sets an argument.

```
filter(.data = gapminder, country == "Tanzania")
```

```
## # A tibble: 12 x 7
##    country  continent  year lifeExp      pop gdpPercap year_cat
##    <fct>    <fct>     <int>   <dbl>    <int>     <dbl> <chr>
##  1 Tanzania Africa     1952    41.2  8322925      717. Before 1980
##  2 Tanzania Africa     1957    43.0  9452826      699. Before 1980
##  3 Tanzania Africa     1962    44.2 10863958      722. Before 1980
##  4 Tanzania Africa     1967    45.8 12607312      848. Before 1980
##  5 Tanzania Africa     1972    47.6 14706593      916. Before 1980
##  6 Tanzania Africa     1977    49.9 17129565      962. Before 1980
##  7 Tanzania Africa     1982    50.6 19844382      874. 1980-2000
##  8 Tanzania Africa     1987    51.5 23040630      832. 1980-2000
##  9 Tanzania Africa     1992    50.4 26605473      826. 1980-2000
## 10 Tanzania Africa     1997    48.5 30686889      789. 1980-2000
## 11 Tanzania Africa     2002    49.7 34593779      899. After 200
## 12 Tanzania Africa     2007    52.5 38139640     1107. After 200
```

Logical tests

Test	Meaning	Test	Meaning
`x < y`	Less than	`x %in% y`	In (group membership)
`x > y`	Greater than	`is.na(x)`	Is missing
`==`	Equal to	`!is.na(x)`	Is not missing
`x <= y`	Less than or equal to		
`x >= y`	Greater than or equal to		
`x != y`	Not equal to		

Task #1: Filtering

Use filter() and logical tests to show…

- The data for Kenya
- All data for countries in Oceania **Hint: Oceania is not a country**
- Rows where the life expectancy is less than 30

filter the data for Kenya

```
filter(gapminder, country == "Kenya")
```

```
## # A tibble: 12 x 7
##    country continent  year lifeExp      pop gdpPercap year_cat
##    <fct>   <fct>     <int>   <dbl>    <int>     <dbl> <chr>
##  1 Kenya   Africa     1952    42.3  6464046      854. Before 1980
##  2 Kenya   Africa     1957    44.7  7454779      944. Before 1980
##  3 Kenya   Africa     1962    47.9  8678557      897. Before 1980
##  4 Kenya   Africa     1967    50.7 10191512     1057. Before 1980
##  5 Kenya   Africa     1972    53.6 12044785     1222. Before 1980
##  6 Kenya   Africa     1977    56.2 14500404     1268. Before 1980
##  7 Kenya   Africa     1982    58.8 17661452     1348. 1980-2000
##  8 Kenya   Africa     1987    59.3 21198082     1362. 1980-2000
##  9 Kenya   Africa     1992    59.3 25020539     1342. 1980-2000
## 10 Kenya   Africa     1997    54.4 28263827     1360. 1980-2000
## 11 Kenya   Africa     2002    51.0 31386842     1288. After 200
## 12 Kenya   Africa     2007    54.1 35610177     1463. After 200
```

filter all data for countries in Oceania

```
filter(gapminder, continent == "Oceania")
```

```
## # A tibble: 24 x 7
##    country   continent  year lifeExp      pop gdpPercap year_cat
##    <fct>     <fct>     <int>   <dbl>    <int>     <dbl> <chr>
##  1 Australia Oceania    1952    69.1  8691212    10040. Before 1980
##  2 Australia Oceania    1957    70.3  9712569    10950. Before 1980
##  3 Australia Oceania    1962    70.9 10794968    12217. Before 1980
##  4 Australia Oceania    1967    71.1 11872264    14526. Before 1980
##  5 Australia Oceania    1972    71.9 13177000    16789. Before 1980
##  6 Australia Oceania    1977    73.5 14074100    18334. Before 1980
##  7 Australia Oceania    1982    74.7 15184200    19477. 1980-2000
##  8 Australia Oceania    1987    76.3 16257249    21889. 1980-2000
##  9 Australia Oceania    1992    77.6 17481977    23425. 1980-2000
## 10 Australia Oceania    1997    78.8 18565243    26998. 1980-2000
## # ... with 14 more rows
```

filter rows where the life expectancy is less than 30

```
filter(gapminder, lifeExp <30)
```

```
## # A tibble: 2 x 7
##   country     continent  year lifeExp     pop gdpPercap year_cat
##   <fct>       <fct>     <int>   <dbl>   <int>     <dbl> <chr>
## 1 Afghanistan Asia       1952    28.8 8425333      779. Before 1980
## 2 Rwanda      Africa     1992    23.6 7290203      737. 1980-2000
```

Common mistakes

- Using = instead of ==
- Forgeting quote

```
## Wrong
filter(gapminder,       country = "Kenya")
filter(gapminder,       country = Kenya)


## Correct
filter(gapminder,       country == "Kenya")
```

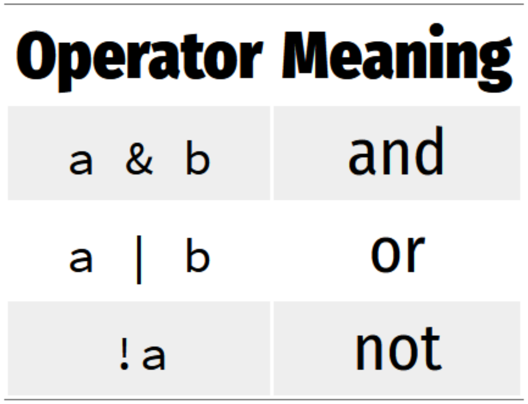
filter() with multiple conditions

- Extract rows that meet every test
- Extract for `Tanzania` before year `2000`

```
filter(.data = gapminder, country == "Tanzania" , year<2000)
```

```
## # A tibble: 10 x 7
##    country  continent  year lifeExp      pop gdpPercap year_cat
##    <fct>    <fct>     <int>   <dbl>    <int>     <dbl> <chr>
##  1 Tanzania Africa     1952    41.2  8322925      717. Before 1980
##  2 Tanzania Africa     1957    43.0  9452826      699. Before 1980
##  3 Tanzania Africa     1962    44.2 10863958      722. Before 1980
##  4 Tanzania Africa     1967    45.8 12607312      848. Before 1980
##  5 Tanzania Africa     1972    47.6 14706593      916. Before 1980
##  6 Tanzania Africa     1977    49.9 17129565      962. Before 1980
##  7 Tanzania Africa     1982    50.6 19844382      874. 1980-2000
##  8 Tanzania Africa     1987    51.5 23040630      832. 1980-2000
##  9 Tanzania Africa     1992    50.4 26605473      826. 1980-2000
## 10 Tanzania Africa     1997    48.5 30686889      789. 1980-2000
```

Boolean operators



Operator	Meaning
`a & b`	and
`a | b`	or
`!a`	not

These do the same thing:

```
filter(.data = gapminder, country == "Tanzania" , year<2000)
```

```
filter(.data = gapminder, country == "Tanzania" &  year<2000)
```

Task #2: Filtering

- **Use filter() and Boolean logical tests to show...**
  - Kenya after 2000
  - Countries where life expectancy in 2002 is over 80
  - Countries where life expectancy in 2007 is below 50 and are not in Africa

filter the data for Kenya after 200

```
filter(gapminder, country == "Kenya" & year>2000)
```

```
## # A tibble: 2 x 7
##   country continent  year lifeExp      pop gdpPercap year_cat
##   <fct>   <fct>     <int>   <dbl>    <int>     <dbl> <chr>
## 1 Kenya   Africa     2002    51.0 31386842     1288. After 200
## 2 Kenya   Africa     2007    54.1 35610177     1463. After 200
```

filter countries where life expectancy in 2002 is over 80

```
filter(gapminder, lifeExp >80 & year==2002)
```

```
## # A tibble: 7 x 7
##   country          continent  year lifeExp       pop gdpPercap year_cat
##   <fct>            <fct>     <int>   <dbl>     <int>     <dbl> <chr>
## 1 Australia        Oceania    2002    80.4  19546792    30688. After 200
## 2 Hong Kong, China Asia       2002    81.5   6762476    30209. After 200
## 3 Iceland          Europe     2002    80.5    288030    31163. After 200
## 4 Italy            Europe     2002    80.2  57926999    27968. After 200
## 5 Japan            Asia       2002    82   127065841    28605. After 200
## 6 Sweden           Europe     2002    80.0   8954175    29342. After 200
## 7 Switzerland      Europe     2002    80.6   7361757    34481. After 200
```

Countries where life expectancy in 2007 is below 50 and are not in Africa

```r
filter(gapminder, lifeExp <50 & continent!='Africa' & year==2007)
```

```
## # A tibble: 1 x 7
##   country     continent  year lifeExp      pop gdpPercap year_cat
##   <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>
## 1 Afghanistan Asia       2007    43.8 31889923      975. After 200
```

Common mistakes

Collapsing multiple tests into one

```
filter(gapminder, 1960 < year < 1980)
```

```
filter(gapminder,
       year > 1960, year < 1980)
```

Using multiple tests instead of `%in%`

```
filter(gapminder,
       country == "Mexico",
       country == "Canada",
       country == "United States")
```

```
filter(gapminder,
       country %in% c("Mexico", "Canada",
                      "United States"))
```

Create new columns: mutate()

dplyr: mutate

```
mutate(.data = DATA, ...)
```

- `DATA` = Data frame to transform
- `...` = Columns to make

mutate the gdp variable

```
mutate(gapminder, gdp = gdpPercap * pop)
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat             gdp
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>              <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980  6567086330.
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980  7585448670.
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980  8758855797.
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980  9648014150.
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980  9678553274.
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 11697659231.
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   12598563401.
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   11820990309.
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   10595901589.
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   14121995875.
## # ... with 1,694 more rows
```

mutate 2 variables

```
mutate(gapminder, gdp = gdpPercap * pop,
                  pop_mil = round(pop / 1000000))
```

```
## # A tibble: 1,704 x 9
##    country     continent  year lifeExp      pop gdpPercap year_cat                gdp pop_mil
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>                 <dbl>   <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980  6567086330.       8
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980  7585448670.       9
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980  8758855797.      10
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980  9648014150.      12
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980  9678553274.      13
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 11697659231.      15
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   12598563401.      13
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   11820990309.      14
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   10595901589.      16
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   14121995875.      22
## # ... with 1,694 more rows
```

Do conditional tests within mutate()

dplyr: ifelse()

```
ifelse(TEST,
       VALUE_IF_TRUE,
       VALUE_IF_FALSE)
```

- TEST = A logical test
- VALUE_IF_TRUE = What happens if test is true
- VALUE_IF_FALSE = What happens if test is false

Create a variable to show before and after 2000

```
mutate(gapminder,
       after_1960 = ifelse(year > 1960, TRUE, FALSE))
```

```
mutate(gapminder,
       after_2000 = ifelse(year > 2000,
                           "After 2000",
                           "Before 2000"))
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat    after_2000
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>       <chr>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980 Before 2000
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980 Before 2000
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980 Before 2000
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980 Before 2000
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980 Before 2000
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 Before 2000
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   Before 2000
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   Before 2000
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   Before 2000
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   Before 2000
## # ... with 1,694 more rows
```

Task #1: Mutate

- **Use mutate() and if_else() to…**
  - Add an `africa` column that is TRUE if the country is on the African continent
  - Add a column for logged GDP per capita **(hint: use log())**
  - Add a column `life_exp_asia` for life expectancy that is TRUE if the country is in Asia and life expectancy id greater than 80
  - Add an `africa_asia` column that says “Africa or Asia” if the country is in Africa or Asia, and “Not Africa or Asia” if it’s not

Add an `africa` column

```
mutate(gapminder, africa = ifelse(continent == "Africa",
                                  TRUE, FALSE))
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat    africa
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>       <lgl> 
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980 FALSE 
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980 FALSE 
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980 FALSE 
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980 FALSE 
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980 FALSE 
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 FALSE 
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   FALSE 
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   FALSE 
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   FALSE 
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   FALSE 
## # ... with 1,694 more rows
```

Add a column for logged GDP per capita

```
mutate(gapminder, log_gdpPercap = log(gdpPercap))
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat    log_gdpPercap
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>               <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980          6.66
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980          6.71
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980          6.75
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980          6.73
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980          6.61
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980          6.67
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000            6.89
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000            6.75
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000            6.48
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000            6.45
## # ... with 1,694 more rows
```

Add a column `life_exp_asia` for Asian countries with lifeExp>80

```r
mutate(gapminder, life_exp_asia=ifelse(continent=="Asia" & lifeExp>80,
                                       TRUE,FALSE))
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat    life_exp_asia
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>       <lgl>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980 FALSE
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980 FALSE
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980 FALSE
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980 FALSE
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980 FALSE
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 FALSE
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   FALSE
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   FALSE
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   FALSE
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   FALSE
## # ... with 1,694 more rows
```

Add an `africa_asia` column

```r
mutate(gapminder,
       africa_asia = ifelse(continent %in% c("Africa", "Asia"),
                            "Africa or Asia",
                            "Not Africa or Asia"))
```

```
## # A tibble: 1,704 x 8
##    country     continent  year lifeExp      pop gdpPercap year_cat    africa_asia
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl> <chr>       <chr>
##  1 Afghanistan Asia       1952    28.8  8425333      779. Before 1980 Africa or Asia
##  2 Afghanistan Asia       1957    30.3  9240934      821. Before 1980 Africa or Asia
##  3 Afghanistan Asia       1962    32.0 10267083      853. Before 1980 Africa or Asia
##  4 Afghanistan Asia       1967    34.0 11537966      836. Before 1980 Africa or Asia
##  5 Afghanistan Asia       1972    36.1 13079460      740. Before 1980 Africa or Asia
##  6 Afghanistan Asia       1977    38.4 14880372      786. Before 1980 Africa or Asia
##  7 Afghanistan Asia       1982    39.9 12881816      978. 1980-2000   Africa or Asia
##  8 Afghanistan Asia       1987    40.8 13867957      852. 1980-2000   Africa or Asia
##  9 Afghanistan Asia       1992    41.7 16317921      649. 1980-2000   Africa or Asia
## 10 Afghanistan Asia       1997    41.8 22227415      635. 1980-2000   Africa or Asia
## # ... with 1,694 more rows
```

What if you have multiple conditions?

- Make a dataset for just 2002 and calculate logged GDP per capita

Solution 1

```
gapminder_2002 <- filter(gapminder, year == 2002)

gapminder_2002_log <- mutate(gapminder_2002,
                             log_gdpPercap = log(gdpPercap))
```

Solution 2: Pipes

- The %>% operator (pipe) takes an object on the left
- Then passes it as the first argument of the function on the right

```
gapminder %>% filter(_, country == "Kenya")
```

- These two commands do the same thing

```
filter(gapminder, country == "Kenya")

gapminder %>% filter(country == "Kenya")
```

- Make a dataset for just 2002 and calculate logged GDP per capita

```
gapminder_2002_log <- gapminder %>%
  filter(year == 2002) %>%
  mutate(log_gdpPercap = log(gdpPercap))
gapminder_2002_log
```

%>%

```
leave_house(
  take_breakfast(
    get_dressed(
      wake_up(
        me, ## start here
        time = "8:00"),
      trouser = TRUE, shirt = TRUE , socks=FALSE),
    mayai = TRUE, viazi = TRUE , chai=TRUE),
  nduthi = TRUE, car = FALSE)
```

```
me %>%
  wake_up(time = "8:00") %>%
  get_dressed(trouser = TRUE, shirt = TRUE , socks=FALSE) %>%
  take_breakfast( mayai = TRUE, viazi = TRUE , chai=TRUE, ukwanju=FALSE) %>%
  leave_house( nduthi = TRUE, car = FALSE)
```

Questions()

Data wrangling with R 1 - Done()

dplyr: summarize()

Summarize

pivot_longer() and pivot_wider()

tidyr: reshape data

pivot_wider()

- `pivot_wider()` “widens” data, increasing the number of columns and decreasing the number of rows.
- `pivot_wider()` is an updated approach to `spread()`

```
DATA %>%
  pivot_wider(names_from,
       values_from ,
       ....)
```

- DATA = A data frame to pivot
- names_from = Column(s) to pivot into wider format.
- values_from = Column(s) to to get the cell values from to be into wider format.
- `...` = other specifications (check help)

- Filter data after 1992
- Select the `continent, country, year and gdpPercap` and pivot_wider the values of `gdpPercap` by `country`

```
gapminder_sub <- gapminder %>%
  select(continent, country, year, gdpPercap) %>%
  filter(year>1992 )
gapminder_sub %>%
 pivot_wider(names_from =country , values_from =gdpPercap )
```

```
## # A tibble: 15 x 144
##    continent  year Afghanistan Albania Algeria Angola Argentina Australia Austria Bahrain Bangladesh Belgium Beni
##    <fct>     <int>       <dbl>   <dbl>   <dbl>  <dbl>     <dbl>     <dbl>   <dbl>   <dbl>      <dbl>   <dbl> <dbl
##  1 Asia       1997        635.      NA      NA     NA        NA        NA      NA  20292.       973.      NA    NA
##  2 Asia       2002        727.      NA      NA     NA        NA        NA      NA  23404.      1136.      NA    NA
##  3 Asia       2007        975.      NA      NA     NA        NA        NA      NA  29796.      1391.      NA    NA
##  4 Europe     1997          NA   3193.      NA     NA        NA        NA  29096.      NA         NA  27561.    NA
##  5 Europe     2002          NA   4604.      NA     NA        NA        NA  32418.      NA         NA  30486.    NA
##  6 Europe     2007          NA   5937.      NA     NA        NA        NA  36126.      NA         NA  33693.    NA
##  7 Africa     1997          NA      NA   4797.  2277.        NA        NA      NA      NA         NA      NA  1233
##  8 Africa     2002          NA      NA   5288.  2773.        NA        NA      NA      NA         NA      NA  1373
##  9 Africa     2007          NA      NA   6223.  4797.        NA        NA      NA      NA         NA      NA  1441
## 10 Americas   1997          NA      NA      NA     NA    10967.        NA      NA      NA         NA      NA    NA
## 11 Americas   2002          NA      NA      NA     NA     8798.        NA      NA      NA         NA      NA    NA
## 12 Americas   2007          NA      NA      NA     NA    12779.        NA      NA      NA         NA      NA    NA
## 13 Oceania    1997          NA      NA      NA     NA        NA    26998.      NA      NA         NA      NA    NA
## 14 Oceania    2002          NA      NA      NA     NA        NA    30688.      NA      NA         NA      NA    NA
## 15 Oceania    2007          NA      NA      NA     NA        NA    34435.      NA      NA         NA      NA    NA
## # ... with 131 more variables: Bolivia <dbl>, Bosnia and Herzegovina <dbl>, Botswana <dbl>, Brazil <dbl>,
## #   Bulgaria <dbl>, Burkina Faso <dbl>, Burundi <dbl>, Cambodia <dbl>, Cameroon <dbl>, Canada <dbl>,
## #   Central African Republic <dbl>, Chad <dbl>, Chile <dbl>, China <dbl>, Colombia <dbl>, Comoros <dbl>,
## #   Congo, Dem. Rep. <dbl>, Congo, Rep. <dbl>, Costa Rica <dbl>, Cote d'Ivoire <dbl>, Croatia <dbl>, Cuba <dbl>,
## #   Czech Republic <dbl>, Denmark <dbl>, Djibouti <dbl>, Dominican Republic <dbl>, Ecuador <dbl>, Egypt <dbl>,
```

- What if I remove the continent?

```
country_gdp_wider <- gapminder_sub %>%
  select(-continent) %>%
 pivot_wider(names_from =country , values_from =gdpPercap )
country_gdp_wider
```

```
## # A tibble: 3 x 143
##    year Afghanistan Albania Algeria Angola Argentina Australia Austria Bahrain Bangladesh Belgium Benin Bolivia
##   <int>       <dbl>   <dbl>   <dbl>  <dbl>     <dbl>     <dbl>   <dbl>   <dbl>      <dbl>   <dbl> <dbl>   <dbl>
## 1  1997        635.   3193.   4797.  2277.    10967.    26998.  29096.  20292.       973.  27561.  1233.   3326.
## 2  2002        727.   4604.   5288.  2773.     8798.    30688.  32418.  23404.      1136.  30486.  1373.   3413.
## 3  2007        975.   5937.   6223.  4797.    12779.    34435.  36126.  29796.      1391.  33693.  1441.   3822.
## # ... with 130 more variables: Bosnia and Herzegovina <dbl>, Botswana <dbl>, Brazil <dbl>, Bulgaria <dbl>,
## #   Burkina Faso <dbl>, Burundi <dbl>, Cambodia <dbl>, Cameroon <dbl>, Canada <dbl>,
## #   Central African Republic <dbl>, Chad <dbl>, Chile <dbl>, China <dbl>, Colombia <dbl>, Comoros <dbl>,
## #   Congo, Dem. Rep. <dbl>, Congo, Rep. <dbl>, Costa Rica <dbl>, Cote d'Ivoire <dbl>, Croatia <dbl>, Cuba <dbl>,
## #   Czech Republic <dbl>, Denmark <dbl>, Djibouti <dbl>, Dominican Republic <dbl>, Ecuador <dbl>, Egypt <dbl>,
## #   El Salvador <dbl>, Equatorial Guinea <dbl>, Eritrea <dbl>, Ethiopia <dbl>, Finland <dbl>, France <dbl>,
## #   Gabon <dbl>, Gambia <dbl>, Germany <dbl>, Ghana <dbl>, Greece <dbl>, Guatemala <dbl>, Guinea <dbl>,
## #   Guinea-Bissau <dbl>, Haiti <dbl>, Honduras <dbl>, Hong Kong, China <dbl>, Hungary <dbl>, Iceland <dbl>,
## #   India <dbl>, Indonesia <dbl>, Iran <dbl>, Iraq <dbl>, Ireland <dbl>, Israel <dbl>, Italy <dbl>,
## #   Jamaica <dbl>, Japan <dbl>, Jordan <dbl>, Kenya <dbl>, Korea, Dem. Rep. <dbl>, Korea, Rep. <dbl>,
## #   Kuwait <dbl>, Lebanon <dbl>, Lesotho <dbl>, Liberia <dbl>, Libya <dbl>, Madagascar <dbl>, Malawi <dbl>,
## #   Malaysia <dbl>, Mali <dbl>, Mauritania <dbl>, Mauritius <dbl>, Mexico <dbl>, Mongolia <dbl>,
## #   Montenegro <dbl>, Morocco <dbl>, Mozambique <dbl>, Myanmar <dbl>, Namibia <dbl>, Nepal <dbl>,
## #   Netherlands <dbl>, New Zealand <dbl>, Nicaragua <dbl>, Niger <dbl>, Nigeria <dbl>, Norway <dbl>, Oman <dbl>,
## #   Pakistan <dbl>, Panama <dbl>, Paraguay <dbl>, Peru <dbl>, Philippines <dbl>, Poland <dbl>, Portugal <dbl>,
## #   Puerto Rico <dbl>, Reunion <dbl>, Romania <dbl>, Rwanda <dbl>, Sao Tome and Principe <dbl>,
## #   Saudi Arabia <dbl>, Senegal <dbl>, Serbia <dbl>, ...
```

Task #1: pivot_wider

- **Use pivot_wider() to…**
  - Show the population data only for African countries before 1992. **(hint: pivot_wider() population values from countries)**
  - Create a `year_cat` variable that is `"Before 1980"` if year in `1952, 1957 ,1962 ,1967, 1972 ,1977`, `"1980-2000"` if year in `1982, 1987, 1992, 1997` and `"After 2000"` if year in `2002 and 2007`
  - Summarize the median `gdpPerCap` for each `year_cat` by country
  - Pivot_wider the median values from the countries

Show the population data only for African countries before 1992

```
africa_before_1992 <- gapminder %>%
  filter(continent=="Africa") %>%
  select(country, year, pop) %>%
  filter(year<1992 )
africa_before_1992_wide <- africa_before_1992 %>%
 pivot_wider(names_from =country , values_from =pop )
africa_before_1992_wide
```

```
## # A tibble: 8 x 53
##    year  Algeria  Angola   Benin Botswana `Burkina Faso` Burundi Cameroon `Central African Republi~   Chad Comoro
##   <int>    <int>   <int>   <int>    <int>          <int>   <int>    <int>                     <int>  <int>   <int
## 1  1952  9279525 4232095 1738315   442308        4469979 2445618  5009067                   1291695 2.68e6  15393
## 2  1957 10270856 4561361 1925173   474639        4713416 2667518  5359923                   1392284 2.89e6  17092
## 3  1962 11000948 4826015 2151895   512764        4919632 2961915  5793633                   1523478 3.15e6  19168
## 4  1967 12760499 5247469 2427334   553541        5127935 3330989  6335506                   1733638 3.50e6  21737
## 5  1972 14760787 5894858 2761407   619351        5433886 3529983  7021028                   1927260 3.90e6  25002
## 6  1977 17152804 6162675 3168267   781472        5889574 3834415  7959865                   2167533 4.39e6  30473
## 7  1982 20033753 7016384 3641603   970347        6634596 4580410  9250831                   2476971 4.88e6  34864
## 8  1987 23254956 7874230 4243788  1151184        7586551 5126023 10780667                   2840009 5.50e6  39511
## # ... with 42 more variables: Congo, Dem. Rep. <int>, Congo, Rep. <int>, Cote d'Ivoire <int>, Djibouti <int>,
## #   Egypt <int>, Equatorial Guinea <int>, Eritrea <int>, Ethiopia <int>, Gabon <int>, Gambia <int>, Ghana <int>,
## #   Guinea <int>, Guinea-Bissau <int>, Kenya <int>, Lesotho <int>, Liberia <int>, Libya <int>, Madagascar <int>,
## #   Malawi <int>, Mali <int>, Mauritania <int>, Mauritius <int>, Morocco <int>, Mozambique <int>, Namibia <int>,
## #   Niger <int>, Nigeria <int>, Reunion <int>, Rwanda <int>, Sao Tome and Principe <int>, Senegal <int>,
## #   Sierra Leone <int>, Somalia <int>, South Africa <int>, Sudan <int>, Swaziland <int>, Tanzania <int>,
## #   Togo <int>, Tunisia <int>, Uganda <int>, Zambia <int>, Zimbabwe <int>
```

Add a column `year_cat`

```r
gapminder <- gapminder %>%
  mutate(year_cat=ifelse(year %in% c(1952, 1957 ,1962 ,1967, 1972 ,1977),
                         "Before 1980",
                         ifelse(year %in% c(1982, 1987, 1992, 1997),
                                "1980-2000","After 200")))
glimpse(gapminder)
```

```
## Rows: 1,704
## Columns: 7
## $ country   <fct> "Afghanistan", "Afghanistan", "Afghanistan", "Afghanistan", "Afghanistan", "Afghanistan", "Afg~
## $ continent <fct> Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Europe, Europe, Europe~
## $ year      <int> 1952, 1957, 1962, 1967, 1972, 1977, 1982, 1987, 1992, 1997, 2002, 2007, 1952, 1957, 1962, 1967~
## $ lifeExp   <dbl> 28.801, 30.332, 31.997, 34.020, 36.088, 38.438, 39.854, 40.822, 41.674, 41.763, 42.129, 43.828~
## $ pop       <int> 8425333, 9240934, 10267083, 11537966, 13079460, 14880372, 12881816, 13867957, 16317921, 222274~
## $ gdpPercap <dbl> 779.4453, 820.8530, 853.1007, 836.1971, 739.9811, 786.1134, 978.0114, 852.3959, 649.3414, 635.~
## $ year_cat  <chr> "Before 1980", "Before 1980", "Before 1980", "Before 1980", "Before 1980", "Before 1980", "198~
```

```r
table(gapminder$year_cat)
```

```
## 
##   1980-2000   After 200 Before 1980 
##         568         284         852
```

Summarize the median of each country by `year_cat`

```r
gapminder_yearcat <- gapminder %>%
 group_by(year_cat, country) %>%
  summarise(med_gdpPercap=median(gdpPercap))
```

```
## `summarise()` has grouped output by 'year_cat'. You can override using the `.groups` argument.
```

```r
head(gapminder_yearcat)
```

```
## # A tibble: 6 x 3
## # Groups:   year_cat [1]
##   year_cat  country     med_gdpPercap
##   <chr>     <fct>               <dbl>
## 1 1980-2000 Afghanistan          751.
## 2 1980-2000 Albania             3412.
## 3 1980-2000 Algeria             5352.
## 4 1980-2000 Angola              2529.
## 5 1980-2000 Argentina           9224.
## 6 1980-2000 Australia          22657.
```

pivot_wider `year_cat` values over country

```
median_gdpPercap <- gapminder_yearcat %>%
   pivot_wider(names_from =country , values_from =med_gdpPercap )
median_gdpPercap
```

```
## # A tibble: 3 x 143
## # Groups:   year_cat [3]
##   year_cat Afghanistan Albania Algeria Angola Argentina Australia Austria Bahrain Bangladesh Belgium Benin Bolivi
##   <chr>          <dbl>   <dbl>   <dbl>  <dbl>     <dbl>     <dbl>   <dbl>   <dbl>      <dbl>   <dbl> <dbl>   <dbl
## 1 1980-20~        751.   3412.   5352.  2529.     9224.    22657.  25365.  19123.       795.  24051. 1229.   3059
## 2 After 2~        851.   5271.   5756.  3785.    10789.    32562.  34272.  26600.      1264.  32089. 1407.   3618
## 3 Before ~        803.   2537.   3130.  4049.     7593.    13372.  11793.  13779.       673.  12070. 1032.   2632
## # ... with 130 more variables: Bosnia and Herzegovina <dbl>, Botswana <dbl>, Brazil <dbl>, Bulgaria <dbl>,
## #   Burkina Faso <dbl>, Burundi <dbl>, Cambodia <dbl>, Cameroon <dbl>, Canada <dbl>,
## #   Central African Republic <dbl>, Chad <dbl>, Chile <dbl>, China <dbl>, Colombia <dbl>, Comoros <dbl>,
## #   Congo, Dem. Rep. <dbl>, Congo, Rep. <dbl>, Costa Rica <dbl>, Cote d'Ivoire <dbl>, Croatia <dbl>, Cuba <dbl>,
## #   Czech Republic <dbl>, Denmark <dbl>, Djibouti <dbl>, Dominican Republic <dbl>, Ecuador <dbl>, Egypt <dbl>,
## #   El Salvador <dbl>, Equatorial Guinea <dbl>, Eritrea <dbl>, Ethiopia <dbl>, Finland <dbl>, France <dbl>,
## #   Gabon <dbl>, Gambia <dbl>, Germany <dbl>, Ghana <dbl>, Greece <dbl>, Guatemala <dbl>, Guinea <dbl>,
## #   Guinea-Bissau <dbl>, Haiti <dbl>, Honduras <dbl>, Hong Kong, China <dbl>, Hungary <dbl>, Iceland <dbl>,
## #   India <dbl>, Indonesia <dbl>, Iran <dbl>, Iraq <dbl>, Ireland <dbl>, Israel <dbl>, Italy <dbl>,
## #   Jamaica <dbl>, Japan <dbl>, Jordan <dbl>, Kenya <dbl>, Korea, Dem. Rep. <dbl>, Korea, Rep. <dbl>,
## #   Kuwait <dbl>, Lebanon <dbl>, Lesotho <dbl>, Liberia <dbl>, Libya <dbl>, Madagascar <dbl>, Malawi <dbl>,
## #   Malaysia <dbl>, Mali <dbl>, Mauritania <dbl>, Mauritius <dbl>, Mexico <dbl>, Mongolia <dbl>,
## #   Montenegro <dbl>, Morocco <dbl>, Mozambique <dbl>, Myanmar <dbl>, Namibia <dbl>, Nepal <dbl>,
## #   Netherlands <dbl>, New Zealand <dbl>, Nicaragua <dbl>, Niger <dbl>, Nigeria <dbl>, Norway <dbl>, Oman <dbl>,
## #   Pakistan <dbl>, Panama <dbl>, Paraguay <dbl>, Peru <dbl>, Philippines <dbl>, Poland <dbl>, Portugal <dbl>,
## #   Puerto Rico <dbl>, Reunion <dbl>, Romania <dbl>, Rwanda <dbl>, Sao Tome and Principe <dbl>,
## #   Saudi Arabia <dbl>, Senegal <dbl>, Serbia <dbl>, ...
```

pivot_longer()

- `pivot_longer()` “lengthens” data, increasing the number of rows and decreasing the number of columns.
- `pivot_longer()` is an updated approach to `gather()`

```
DATA %>%
  pivot_longer(cols,
               names_to = ,
               values_to = ,
       ....)
```

- DATA = A data frame to pivot
- cols = Columns to pivot into longer format.
- names_to = name of the column to create from the data stored in the column names
- values_to = string specifying the name of the column to create from the data stored in cell values
- `...` = other specifications (check help)

- We use the `country_gdp` pivot_longer the values of `gdpPercap`

```
country_gdp_longer <- country_gdp_wider %>%
 pivot_longer(cols =!year, names_to="country",values_to = "gdpPercap")
country_gdp_longer
```

```
## # A tibble: 426 x 3
##     year country     gdpPercap
##    <int> <chr>           <dbl>
##  1  1997 Afghanistan      635.
##  2  1997 Albania         3193.
##  3  1997 Algeria         4797.
##  4  1997 Angola          2277.
##  5  1997 Argentina      10967.
##  6  1997 Australia      26998.
##  7  1997 Austria        29096.
##  8  1997 Bahrain        20292.
##  9  1997 Bangladesh       973.
## 10  1997 Belgium        27561.
## # ... with 416 more rows
```

Task #1: pivot_longer

- Use pivot_longer() to...
  - pivot_longer the values of `median_gdpPercap` into `country` and `med_gdpPercap`

Solution

```
median_gdpPercap_longer <- median_gdpPercap %>%
 pivot_longer(cols =!year_cat , names_to="country",values_to = "med_gdpPercap")
median_gdpPercap_longer
```

```
## # A tibble: 426 x 3
## # Groups:   year_cat [3]
##    year_cat  country     med_gdpPercap
##    <chr>     <chr>               <dbl>
##  1 1980-2000 Afghanistan          751.
##  2 1980-2000 Albania             3412.
##  3 1980-2000 Algeria             5352.
##  4 1980-2000 Angola              2529.
##  5 1980-2000 Argentina           9224.
##  6 1980-2000 Australia          22657.
##  7 1980-2000 Austria            25365.
##  8 1980-2000 Bahrain            19123.
##  9 1980-2000 Bangladesh           795.
## 10 1980-2000 Belgium            24051.
## # ... with 416 more rows
```

How to export data to CSV

```
write_csv(x = median_gdpPercap_longer, file = "median_gdpPercap_longer.csv")
```

How to export data to Stata

```
## bad code
library(haven)
haven::write_dta(x = median_gdpPercap_longer, file = "median_gdpPercap_longer.dta")
```

Questions()

Data wrangling with R - Done()