Data Wrangling

Thiyanga Talagala

Load the `gapminder` dataset and the `tidyverse` and `magrittr` packages.

Exercises

1. Filter all rows for "Sweden".

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

```
# A tibble: 12 x 6
   country continent  year lifeExp     pop gdpPercap
   <fct>   <fct>     <int>   <dbl>   <int>     <dbl>
 1 Sweden  Europe     1952    71.9 7124673     8528.
 2 Sweden  Europe     1957    72.5 7363802     9912.
 3 Sweden  Europe     1962    73.4 7561588    12329.
 4 Sweden  Europe     1967    74.2 7867931    15258.
 5 Sweden  Europe     1972    74.7 8122293    17832.
 6 Sweden  Europe     1977    75.4 8251648    18856.
 7 Sweden  Europe     1982    76.4 8325260    20667.
 8 Sweden  Europe     1987    77.2 8421403    23587.
 9 Sweden  Europe     1992    78.2 8718867    23880.
10 Sweden  Europe     1997    79.4 8897619    25267.
11 Sweden  Europe     2002    80.0 8954175    29342.
12 Sweden  Europe     2007    80.9 9031088    33860.
```

2. Filter all rows where `lifeExp` is less than or equal to 30.

```
gapminder %>% filter(lifeExp <= 50)
```

```
# A tibble: 491 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.
 2 Afghanistan Asia       1957    30.3  9240934      821.
 3 Afghanistan Asia       1962    32.0 10267083      853.
 4 Afghanistan Asia       1967    34.0 11537966      836.
 5 Afghanistan Asia       1972    36.1 13079460      740.
 6 Afghanistan Asia       1977    38.4 14880372      786.
 7 Afghanistan Asia       1982    39.9 12881816      978.
 8 Afghanistan Asia       1987    40.8 13867957      852.
 9 Afghanistan Asia       1992    41.7 16317921      649.
10 Afghanistan Asia       1997    41.8 22227415      635.
# ... with 481 more rows
```

3. Filter all rows that have a missing value for `year`.

```
filter(gapminder, is.na(year))
```

```
# A tibble: 0 x 6
# ... with 6 variables: country <fct>, continent <fct>, year <int>,
#   lifeExp <dbl>, pop <int>, gdpPercap <dbl>
```

4. Filter all countries that had population over 100000 in 1960 or earlier.

```
filter(gapminder, pop>100000 & year <=1960)
```

```
# A tibble: 280 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.
 2 Afghanistan Asia       1957    30.3  9240934      821.
 3 Albania     Europe     1952    55.2  1282697     1601.
 4 Albania     Europe     1957    59.3  1476505     1942.
 5 Algeria     Africa     1952    43.1  9279525     2449.
 6 Algeria     Africa     1957    45.7 10270856     3014.
 7 Angola      Africa     1952    30.0  4232095     3521.
 8 Angola      Africa     1957    32.0  4561361     3828.
 9 Argentina   Americas   1952    62.5 17876956     5911.
10 Argentina   Americas   1957    64.4 19610538     6857.
# ... with 270 more rows
```

5. Count the number of countries with life expectancy greater than 30 in 1952.

```
df <- gapminder %>%
  filter(year==1952 & lifeExp < 30)
df
```

```
# A tibble: 1 x 6
  country     continent  year lifeExp     pop gdpPercap
  <fct>       <fct>     <int>   <dbl>   <int>     <dbl>
1 Afghanistan Asia       1952    28.8 8425333      779.
```

```
dim(df)
```

```
[1] 1 6
```

6. Calculate the mean life expectancy for each year and continent.

```
gapminder %>%
  group_by(continent, year) %>%
  summarise(mean.lifeExp = mean(lifeExp))
```

```
`summarise()` regrouping output by 'continent' (override with `.groups` argument)
```

```
# A tibble: 60 x 3
# Groups:   continent [5]
   continent  year mean.lifeExp
   <fct>     <int>        <dbl>
 1 Africa     1952         39.1
 2 Africa     1957         41.3
 3 Africa     1962         43.3
 4 Africa     1967         45.3
 5 Africa     1972         47.5
 6 Africa     1977         49.6
 7 Africa     1982         51.6
 8 Africa     1987         53.3
 9 Africa     1992         53.6
10 Africa     1997         53.6
# ... with 50 more rows
```

7. Get the maximum and minimum of GDP per capita for all continents in a "wide" format.

```
gapminder %>%
  group_by(continent) %>%
  summarize(maxGdpPercap=max(gdpPercap),
            minGdpPercap=min(gdpPercap))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 5 x 3
  continent maxGdpPercap minGdpPercap
  <fct>            <dbl>        <dbl>
1 Africa          21951.         241.
2 Americas        42952.        1202.
3 Asia           113523.         331
4 Europe          49357.         974.
5 Oceania         34435.       10040.
```

8. Get the maximum and minimum of GDP per capita for all continents in a "long" format.

```
gapminder %>%
  group_by(continent) %>%
  summarize(maxGdpPercap=max(gdpPercap),
            minGdpPercap=min(gdpPercap)) %>%
  pivot_longer(2:3, "summary", "value")
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 10 x 3
   continent summary        value
   <fct>     <chr>          <dbl>
 1 Africa    maxGdpPercap  21951.
 2 Africa    minGdpPercap    241.
 3 Americas  maxGdpPercap  42952.
 4 Americas  minGdpPercap   1202.
```

```
 5 Asia      maxGdpPercap 113523.
 6 Asia      minGdpPercap    331
 7 Europe    maxGdpPercap  49357.
 8 Europe    minGdpPercap    974.
 9 Oceania   maxGdpPercap  34435.
10 Oceania   minGdpPercap  10040.
```

9. What was the population of the United States in 1952 and 2007.

```
gapminder %>%
  filter(country=="United States", year %in% c(1952, 2007))
```

```
# A tibble: 2 x 6
  country       continent  year lifeExp       pop gdpPercap
  <fct>         <fct>     <int>   <dbl>     <int>     <dbl>
1 United States Americas   1952    68.4 157553000    13990.
2 United States Americas   2007    78.2 301139947    42952.
```

10. Subset the gapminder data to extract rows where `lifeExp` is greater than or equal 80. Retain only the columns `country`, `year`, and `lifeExp`. Sort the results from largest to smallest based on `lifeExp`.

```
gapminder %>%
filter(lifeExp >= 80) %>%
select(country, year, lifeExp) %>%
arrange(desc(lifeExp))
```

```
# A tibble: 22 x 3
   country           year lifeExp
   <fct>            <int>   <dbl>
 1 Japan             2007    82.6
 2 Hong Kong, China  2007    82.2
 3 Japan             2002    82
 4 Iceland           2007    81.8
 5 Switzerland       2007    81.7
 6 Hong Kong, China  2002    81.5
 7 Australia         2007    81.2
 8 Spain             2007    80.9
 9 Sweden            2007    80.9
10 Israel            2007    80.7
# ... with 12 more rows
```

11. Calculate the total GDP in billions of dollars, extract the results for the year 2002, and sort the rows so that the total GDP is in decreasing order.

Help: `gpd = gdpPercap * pop`

```
gapminder %>%
mutate(gdp = gdpPercap * pop) %>%
  filter(year==2002) %>%
  arrange(desc(gdp))
```

```
# A tibble: 142 x 7
   country        continent  year lifeExp        pop gdpPercap     gdp
   <fct>          <fct>     <int>   <dbl>      <int>     <dbl>   <dbl>
 1 United States  Americas   2002    77.3  287675526    39097. 1.12e13
 2 China          Asia       2002    72.0 1280400000     3119. 3.99e12
 3 Japan          Asia       2002    82    127065841    28605. 3.63e12
 4 Germany        Europe     2002    78.7   82350671    30036. 2.47e12
 5 India          Asia       2002    62.9 1034172547     1747. 1.81e12
 6 United Kingdom Europe     2002    78.5   59912431    29479. 1.77e12
 7 France         Europe     2002    79.6   59925035    28926. 1.73e12
 8 Italy          Europe     2002    80.2   57926999    27968. 1.62e12
 9 Brazil         Americas   2002    71.0  179914212     8131. 1.46e12
10 Mexico         Americas   2002    74.9  102479927    10742. 1.10e12
# ... with 132 more rows
```

12. Calculate the average life expectancy by continent in 2002.

```
gapminder %>%
  filter(year==2002) %>%
group_by(continent) %>%
summarize(mean_lifeExp=mean(lifeExp))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 5 x 2
  continent mean_lifeExp
  <fct>            <dbl>
1 Africa            53.3
2 Americas          72.4
3 Asia              69.2
4 Europe            76.7
5 Oceania           79.7
```

13. Which countries and which years had the worst five GDP per capita measurements?

```
gapminder %>%
  arrange(desc(gdpPercap)) %>%
  tail(5)
```

```
# A tibble: 5 x 6
  country          continent  year lifeExp      pop gdpPercap
  <fct>            <fct>     <int>   <dbl>    <int>     <dbl>
1 Congo, Dem. Rep. Africa     1997    42.6 47798986      312.
2 Guinea-Bissau    Africa     1952    32.5   580653      300.
3 Lesotho          Africa     1952    42.1   748747      299.
4 Congo, Dem. Rep. Africa     2007    46.5 64606759      278.
5 Congo, Dem. Rep. Africa     2002    45.0 55379852      241.
```

14. What was the mean life expectancy across all countries for each year in the dataset?

```r
gapminder %>%
  group_by(year) %>%
  summarize(mean(lifeExp))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 12 x 2
    year `mean(lifeExp)`
   <int>           <dbl>
 1  1952            49.1
 2  1957            51.5
 3  1962            53.6
 4  1967            55.7
 5  1972            57.6
 6  1977            59.6
 7  1982            61.5
 8  1987            63.2
 9  1992            64.2
10  1997            65.0
11  2002            65.7
12  2007            67.0
```

15. Which five Asian countries had the highest life expectancy in 2007?

```r
gapminder %>%
  filter(continent=="Asia") %>%
  arrange(desc(lifeExp)) %>%
  head(5)
```

```
# A tibble: 5 x 6
  country          continent  year lifeExp       pop gdpPercap
  <fct>            <fct>     <int>   <dbl>     <int>     <dbl>
1 Japan            Asia       2007    82.6 127467972    31656.
2 Hong Kong, China Asia       2007    82.2   6980412    39725.
3 Japan            Asia       2002    82   127065841    28605.
4 Hong Kong, China Asia       2002    81.5   6762476    30209.
5 Israel           Asia       2007    80.7   6426679    25523.
```

16. Calculate the total number of observations for each country in Europe. Help: use `n()` function.

```r
gapminder %>%
filter(continent == "Europe") %>%
group_by(country) %>%
summarize(n = n())
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 30 x 2
   country                   n
   <fct>                 <int>
 1 Albania                  12
```

```
 2 Austria                  12
 3 Belgium                  12
 4 Bosnia and Herzegovina   12
 5 Bulgaria                 12
 6 Croatia                  12
 7 Czech Republic           12
 8 Denmark                  12
 9 Finland                  12
10 France                   12
# ... with 20 more rows
```

17. How many observations do we have per continent?

```
gapminder %>%
group_by(continent) %>%
summarize(n = n())
```

```
`summarise()` ungrouping output (override with `.groups` argument)

# A tibble: 5 x 2
  continent     n
  <fct>     <int>
1 Africa      624
2 Americas    300
3 Asia        396
4 Europe      360
5 Oceania      24
```

18. Compute the average life expectancy by continent.

```
gapminder %>%
group_by(continent) %>%
summarize(avg_lifeExp = mean(lifeExp))
```

```
`summarise()` ungrouping output (override with `.groups` argument)

# A tibble: 5 x 2
  continent avg_lifeExp
  <fct>           <dbl>
1 Africa           48.9
2 Americas         64.7
3 Asia             60.1
4 Europe           71.9
5 Oceania          74.3
```

19. Rank countries according to their life expectancy and store it in a new column called rank. Rearrange the rows according to the ascending order of ranks (1, 2, 3. . .).

```
gapminder %>%
  filter(year == 2007) %>%
  select(country, lifeExp) %>%
  mutate(rank = min_rank(desc(lifeExp))) %>%
  arrange(rank)
```

```
# A tibble: 142 x 3
   country          lifeExp  rank
   <fct>              <dbl> <int>
 1 Japan               82.6     1
 2 Hong Kong, China    82.2     2
 3 Iceland             81.8     3
 4 Switzerland         81.7     4
 5 Australia           81.2     5
 6 Spain               80.9     6
 7 Sweden              80.9     7
 8 Israel              80.7     8
 9 France              80.7     9
10 Canada              80.7    10
# ... with 132 more rows
```

20. Calculate the mean and the standard error of the life expectancy for Belgium, Netherlands and France.

```
gapminder %>%
  filter(country %in% c("Belgium", "Netherlands", "France")) %>%
  group_by(country) %>%
  summarize(mean = mean(lifeExp), se = sd(lifeExp)/sqrt(n()))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 3 x 3
  country      mean    se
  <fct>       <dbl> <dbl>
1 Belgium      73.6 1.09
2 France       74.3 1.24
3 Netherlands  75.6 0.718
```

21. Categorize countries as "low" (`lifeExp` < 50) and "high" (`lifeExp` > 50) and store the values in a new column named "category".

```
gapminder %>%
   mutate(category = ifelse(lifeExp > 50, "high", "low"))
```

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