Lesson 07-02: In-class assignment

NAME

3/18/2021

Set-up

This block will install the gapminder package and load it. If this is your first time, you will need to "uncomment" the `install.packages` line in order to install it. You will want to comment it back out or delete it once it's installed, or it can create problems when knitting

```
# install.packages("gapminder")
library(tidyverse)
library(gapminder)
```

Now that we've loaded the library, we have access to a *data frame*. Take a look at the data set by running this code. You can also type `View(gapminder)` into the console to see *all* the data, which will show up in a new window.

Note that you can use `head()` and `tail()` to show the first and last six rows of the data as well. What happens when you run `head(gapminder,20)`?

```
# Show a tibble
gapminder
```

```
## # A tibble: 1,704 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 1,694 more rows
```

```
# Show the first six rows (6 is default)
head(gapminder)
```

```
## # A tibble: 6 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.
```

```
# Show the last six rows (6 is default)

tail(gapminder)
```

```
## # A tibble: 6 x 6
##   country  continent  year lifeExp      pop gdpPercap
##   <fct>    <fct>     <int>   <dbl>    <int>     <dbl>
## 1 Zimbabwe Africa     1982    60.4  7636524      789.
## 2 Zimbabwe Africa     1987    62.4  9216418      706.
## 3 Zimbabwe Africa     1992    60.4 10704340      693.
## 4 Zimbabwe Africa     1997    46.8 11404948      792.
## 5 Zimbabwe Africa     2002    40.0 11926563      672.
## 6 Zimbabwe Africa     2007    43.5 12311143      470.
```

```
## What happens here?

head(gapminder,20)
```

```
## # A tibble: 20 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.
## 11 Afghanistan Asia       2002    42.1 25268405      727.
## 12 Afghanistan Asia       2007    43.8 31889923      975.
## 13 Albania     Europe     1952    55.2  1282697     1601.
## 14 Albania     Europe     1957    59.3  1476505     1942.
## 15 Albania     Europe     1962    64.8  1728137     2313.
## 16 Albania     Europe     1967    66.2  1984060     2760.
## 17 Albania     Europe     1972    67.7  2263554     3313.
## 18 Albania     Europe     1977    68.9  2509048     3533.
## 19 Albania     Europe     1982    70.4  2780097     3631.
## 20 Albania     Europe     1987    72    3075321     3739.
```

Selecting with `select()`

We can select specific columns or a range of columns using `select()`

What does this do?

Write two different ways to select country, continent, year, and gdpPercap

```
# First way to select country, continent, year, and gdpPercal

# Second way to select country, continent, year, and gdpPercal
```

Filtering with `filter()`

Recall that we can use `filter()` restrict our data according to criteria. We can display it as is, or we can assign it to an object if we want to use that restricted data later.

You can also use multiple conditions, and these will extract rows that meet every test. By default, if you separate the tests with a comma, R will consider this an "and" test and find rows that are both Denmark and greater than 2000.

What does this do?

```
filter(gapminder,gdpPercap < 1000)
```

```
## # A tibble: 351 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 341 more rows
```

```
# Note that putting parentheses around the entire line below will assign the object /and/ display it

(verypoor <- filter(gapminder, year == 1992, gdpPercap<500))
```

```
## # A tibble: 4 x 6
##   country          continent  year lifeExp      pop gdpPercap
##   <fct>            <fct>     <int>   <dbl>    <int>     <dbl>
## 1 Congo, Dem. Rep. Africa     1992    45.5 41672143      458.
## 2 Ethiopia         Africa     1992    48.1 52088559      421.
## 3 Mozambique       Africa     1992    44.3 13160731      411.
## 4 Myanmar          Asia       1992    59.3 40546538      347
```

Use `filter()` to do the following:

```
# Show all data for Canada
filter(gapminder,country == "Canada")
```

```
## # A tibble: 12 x 6
##    country continent  year lifeExp      pop gdpPercap
##    <fct>   <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Canada  Americas   1952    68.8 14785584    11367.
##  2 Canada  Americas   1957    70.0 17010154    12490.
##  3 Canada  Americas   1962    71.3 18985849    13462.
##  4 Canada  Americas   1967    72.1 20819767    16077.
##  5 Canada  Americas   1972    72.9 22284500    18971.
##  6 Canada  Americas   1977    74.2 23796400    22091.
##  7 Canada  Americas   1982    75.8 25201900    22899.
##  8 Canada  Americas   1987    76.9 26549700    26627.
##  9 Canada  Americas   1992    78.0 28523502    26343.
## 10 Canada  Americas   1997    78.6 30305843    28955.
## 11 Canada  Americas   2002    79.8 31902268    33329.
## 12 Canada  Americas   2007    80.7 33390141    36319.
```

```
## Show all data for Oceania
filter(gapminder,continent == "Oceania")
```

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

```
## Show all rows where the life expectancy is greater than 82
filter(gapminder,lifeExp > 70)
```

```
## # A tibble: 493 x 6
##    country   continent  year lifeExp      pop gdpPercap
##    <fct>     <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Albania   Europe     1982    70.4  2780097     3631.
##  2 Albania   Europe     1987    72    3075321     3739.
##  3 Albania   Europe     1992    71.6  3326498     2497.
##  4 Albania   Europe     1997    73.0  3428038     3193.
##  5 Albania   Europe     2002    75.7  3508512     4604.
##  6 Albania   Europe     2007    76.4  3600523     5937.
##  7 Algeria   Africa     2002    71.0 31287142     5288.
##  8 Algeria   Africa     2007    72.3 33333216     6223.
##  9 Argentina Americas   1987    70.8 31620918     9140.
## 10 Argentina Americas   1992    71.9 33958947     9308.
## # ... with 483 more rows
```

```
## Show all countries where the life expetancy is greater than 82 and the country is in Africa (you can
filter(gapminder,lifeExp>70,continent == "Africa")
```

```
## # A tibble: 19 x 6
##    country   continent  year lifeExp      pop gdpPercap
##    <fct>     <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Algeria   Africa     2002    71.0 31287142     5288.
##  2 Algeria   Africa     2007    72.3 33333216     6223.
##  3 Egypt     Africa     2007    71.3 80264543     5581.
##  4 Libya     Africa     1997    71.6  4759670     9467.
##  5 Libya     Africa     2002    72.7  5368585     9535.
##  6 Libya     Africa     2007    74.0  6036914    12057.
##  7 Mauritius Africa     1997    70.7  1149818     7426.
##  8 Mauritius Africa     2002    72.0  1200206     9022.
##  9 Mauritius Africa     2007    72.8  1250882    10957.
## 10 Morocco   Africa     2007    71.2 33757175     3820.
## 11 Reunion   Africa     1987    71.9   562035     5303.
## 12 Reunion   Africa     1992    73.6   622191     6101.
## 13 Reunion   Africa     1997    74.8   684810     6072.
## 14 Reunion   Africa     2002    75.7   743981     6316.
## 15 Reunion   Africa     2007    76.4   798094     7670.
## 16 Tunisia   Africa     1992    70.0  8523077     4333.
## 17 Tunisia   Africa     1997    72.0  9231669     4877.
## 18 Tunisia   Africa     2002    73.0  9770575     5723.
## 19 Tunisia   Africa     2007    73.9 10276158     7093.
```

Arrange

We sort our data using `arrange()`

The first line sorts by year, then by country. What does the second line do? What does the third line do?

```
arrange(gapminder,year, country)
```

```
## # A tibble: 1,704 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779.
##  2 Albania     Europe     1952    55.2  1282697     1601.
##  3 Algeria     Africa     1952    43.1  9279525     2449.
##  4 Angola      Africa     1952    30.0  4232095     3521.
##  5 Argentina   Americas   1952    62.5 17876956     5911.
##  6 Australia   Oceania    1952    69.1  8691212    10040.
##  7 Austria     Europe     1952    66.8  6927772     6137.
##  8 Bahrain     Asia       1952    50.9   120447     9867.
##  9 Bangladesh  Asia       1952    37.5 46886859      684.
## 10 Belgium     Europe     1952    68    8730405     8343.
## # ... with 1,694 more rows
```

```
arrange(gapminder,desc(year), continent, country)
```

```
## # A tibble: 1,704 x 6
##    country                  continent  year lifeExp      pop gdpPercap
##    <fct>                    <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Algeria                  Africa     2007    72.3 33333216     6223.
##  2 Angola                   Africa     2007    42.7 12420476     4797.
##  3 Benin                    Africa     2007    56.7  8078314     1441.
##  4 Botswana                 Africa     2007    50.7  1639131    12570.
##  5 Burkina Faso             Africa     2007    52.3 14326203     1217.
##  6 Burundi                  Africa     2007    49.6  8390505      430.
##  7 Cameroon                 Africa     2007    50.4 17696293     2042.
##  8 Central African Republic Africa     2007    44.7  4369038      706.
##  9 Chad                     Africa     2007    50.7 10238807     1704.
## 10 Comoros                  Africa     2007    65.2   710960      986.
## # ... with 1,694 more rows
```

Use `arrange()` to do the followin:

```
# Sort the data by year, then by descending life expectancy
```

Piping

You can nest commands inside other commands to combine them, but your life will be 100% better with... piping! (The keyboard shortcut is `Shift-Command-M` on a Mac, or `Ctrl-Shift-M` on a PC)

```
# Sort by descending life expectancy in 2007

gapminder %>% filter(year == "2007") %>% arrange(desc(lifeExp))
```

```
## # A tibble: 142 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 Iceland          Europe     2007    81.8    301931    36181.
##  4 Switzerland      Europe     2007    81.7   7554661    37506.
##  5 Australia        Oceania    2007    81.2  20434176    34435.
##  6 Spain            Europe     2007    80.9  40448191    28821.
##  7 Sweden           Europe     2007    80.9   9031088    33860.
##  8 Israel           Asia       2007    80.7   6426679    25523.
##  9 France           Europe     2007    80.7  61083916    30470.
## 10 Canada           Americas   2007    80.7  33390141    36319.
## # ... with 132 more rows
```

```
## Show just population data, year and country names when looking at the most populous countries in 197


## Does not work
#gapminder %>% select(country,pop) %>% filter(year == "1977") %>% arrange(desc(pop))

gapminder %>% select(country,year,pop) %>% filter(year == "1977") %>% arrange(desc(pop))
```

```
## # A tibble: 142 x 3
##    country        year       pop
##    <fct>         <int>     <int>
##  1 China          1977 943455000
##  2 India          1977 634000000
##  3 United States  1977 220239000
##  4 Indonesia      1977 136725000
##  5 Brazil         1977 114313951
##  6 Japan          1977 113872473
##  7 Bangladesh     1977  80428306
##  8 Germany        1977  78160773
##  9 Pakistan       1977  78152686
## 10 Mexico         1977  63759976
## # ... with 132 more rows
```

Use `arrange()` and `filter()` to do the following:

```
# Examine the countries with the lowest life expectancy in 2002
```

In general, you can also use help files, Google, and other resources to find new commands. I wanted to list countries that ever had life expectancies below 35.

```
gapminder %>%  filter(lifeExp < 35)
```

```
## # A tibble: 33 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 Angola       Africa     1952    30.0  4232095     3521.
##  6 Angola       Africa     1957    32.0  4561361     3828.
##  7 Angola       Africa     1962    34    4826015     4269.
##  8 Burkina Faso Africa     1952    32.0  4469979      543.
##  9 Burkina Faso Africa     1957    34.9  4713416      617.
## 10 Cambodia     Asia       1977    31.2  6978607      525.
## # ... with 23 more rows
```

This is okay, but there are lots of duplicates! I also don't want to pick just one year, because life expectancies aren't always increasing. What's one way to solve this if you don't know where to start?

1. Googled what I wanted to do "list distinct observations" (it may take a few tries)
2. Looked for a command that worked with frames.
3. Ended up in the `dplyr` documentation (surprise!) with the distinct command.
4. Reviewed briefly, but then jumped down to examples to make sure I knew how to apply it.

```
gapminder %>%  filter(lifeExp < 35) %>% distinct(country)
```

```
## # A tibble: 16 x 1
##    country
```

```
##    <fct>
##  1 Afghanistan
##  2 Angola
##  3 Burkina Faso
##  4 Cambodia
##  5 Djibouti
##  6 Equatorial Guinea
##  7 Ethiopia
##  8 Gambia
##  9 Guinea
## 10 Guinea-Bissau
## 11 Mali
## 12 Mozambique
## 13 Rwanda
## 14 Sierra Leone
## 15 Somalia
## 16 Yemen, Rep.
```

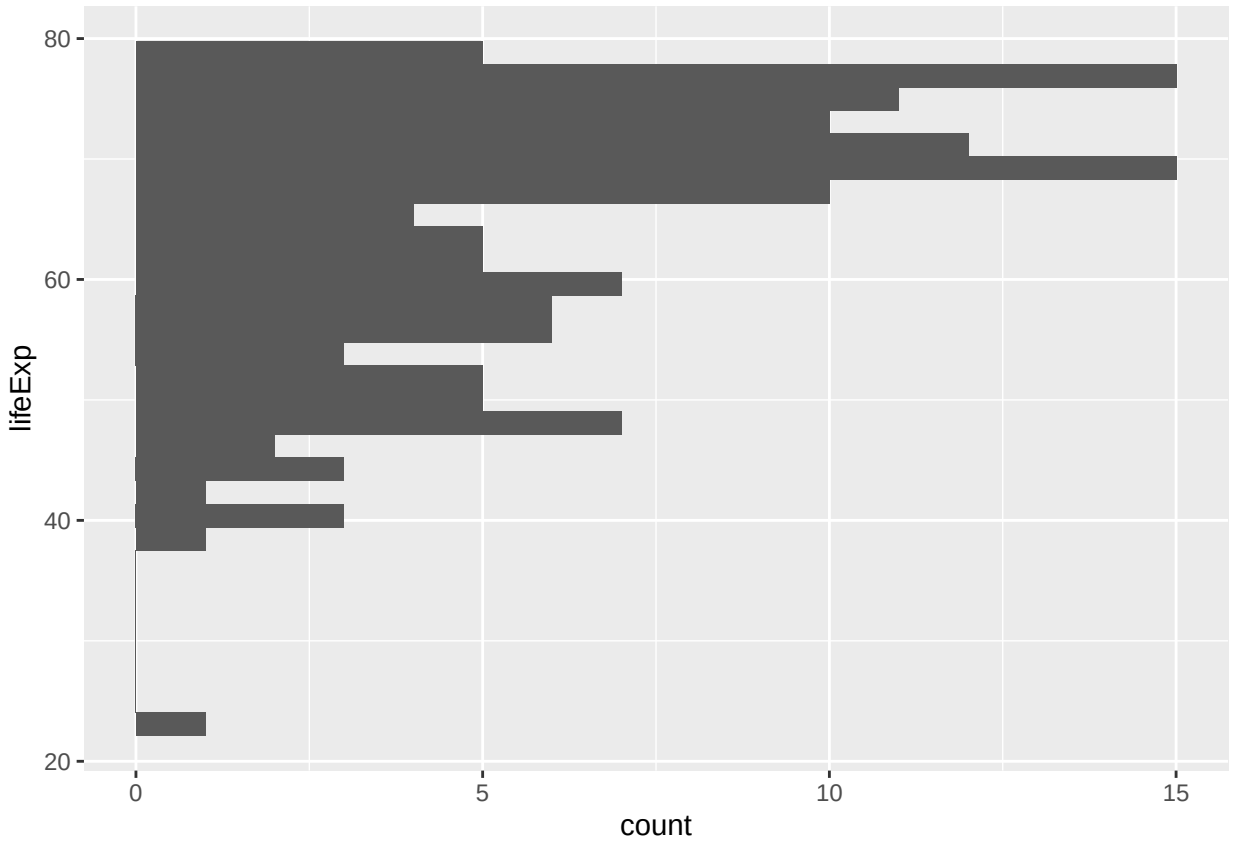
Basic plotting

Next week we'll do more plotting. For now, use what we know about filters to revise this code to plot only for 2007.

```
## This takes the data from gapminder and plots a historgram

gapminder %>%  filter(year == "1992") %>% ggplot(mapping=aes(y=lifeExp)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


80
60
40
20
lifeExp
0
5
10
15
count