

# PHW251 Problem Set 4

Nilson Palma

2025-09-29

For this problem set you will tidy up a dataset of 500 individuals. We also want to calculate each individual's BMI and appropriately categorize them.

Load your data (“../data/500\_Person\_Gender\_Height\_Weight.csv”):

## Question 1

Clean the column headers to be all lower case, have no spaces, and rename “Location information” to location.

```
persons <- clean_names(persons)
persons <- persons %>%
  rename(location = location_information)
colnames(persons)
```

```
[1] "location" "gender"   "height"   "weight"
```

## Question 2

Create a new variable that calculates BMI for each individual.

You will need to navigate the different system of measurements (metric vs imperial). Only the United States is using imperial.

- BMI calculation and conversions:
  - metric:  $BMI = weight(kg) / [height(m)]^2$
  - imperial:  $BMI = 703 * weight(lbs) / [height(in)]^2$
  - 1 foot = 12 inches
  - 1 cm = 0.01 meter

Although there's many ways you can accomplish this task, we want you to use an `if_else()` to calculate BMI with the appropriate formula based on each person's location.

```
unique(persons$location)
```

```
[1] "New York"          "United Kingdom" "Taiwan"          "Colorado"
[5] "Hawaii"
```

```
persons1 <- persons %>%
  mutate(bmi = if_else(
    location %in% c("United Kingdom", "Taiwan"),
    weight / ((height/100)^2),
    703 * weight / ((height*12)^2)
  ))
head(persons1)
```

```
# A tibble: 6 x 5
  location      gender height weight  bmi
  <chr>         <chr>   <dbl> <dbl> <dbl>
1 New York     Male     5.71  212.  31.7
2 United Kingdom Male    189    87  24.4
3 New York     Female   6.07  243.  32.1
4 Taiwan       Female  195   104  27.4
5 Taiwan       Male    149    61  27.5
6 Taiwan       Male    189   104  29.1
```

### Question 3

Create a new variable that categorizes BMI with `case_when()`:

- Underweight: BMI below 18.5
- Normal: 18.5-24.9
- Overweight: 25.0-29.9
- Obese: 30.0 and Above

```
persons2 <- persons1 %>%  
  mutate(bmi_cat = case_when(  
    bmi < 18.5 ~ "Underweight",  
    bmi >= 18.5 & bmi <= 24.9 ~ "Normal",  
    bmi >= 25 & bmi <= 29.9 ~ "Overweight",  
    bmi >= 30.0 ~ "Obese"  
  ))  
head(persons2)
```

# A tibble: 6 x 6

	location	gender	height	weight	bmi	bmi_cat
	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<chr>
1	New York	Male	5.71	212.	31.7	Obese
2	United Kingdom	Male	189	87	24.4	Normal
3	New York	Female	6.07	243.	32.1	Obese
4	Taiwan	Female	195	104	27.4	Overweight
5	Taiwan	Male	149	61	27.5	Overweight
6	Taiwan	Male	189	104	29.1	Overweight

Could we have used `if_else()`?

Yes `if_else()` could do the same categorization.

#### Question 4

Arrange your data first by location and then by descending order of BMI.

```
arrange(persons2, location, desc(bmi))
```

```
# A tibble: 500 x 6
  location gender height weight   bmi bmi_cat
  <chr>    <chr>   <dbl>  <dbl> <dbl> <chr>
1 Colorado Female   4.66   351.  78.8 Obese
2 Colorado Female   4.59   322.  74.6 Obese
3 Colorado Male     4.72   320.  70.1 Obese
4 Colorado Female   4.95   348.  69.4 Obese
5 Colorado Female   4.66   302.  67.9 Obese
6 Colorado Male     4.95   340.  67.7 Obese
7 Colorado Male     4.76   313.  67.5 Obese
8 Colorado Male     4.59   284.  65.9 Obese
9 Colorado Male     4.66   289.  64.9 Obese
10 Colorado Female  4.63   278.  63.3 Obese
# i 490 more rows
```

#### Question 5

Use a dplyr method to remove the height, weight, and BMI columns from your data.

```
persons3 <- persons2 %>%
  select(-c(height, weight, bmi))

head(persons3)
```

```
# A tibble: 6 x 3
  location gender bmi_cat
  <chr>    <chr> <chr>
1 New York Male   Obese
2 United Kingdom Male   Normal
3 New York Female Obese
4 Taiwan   Female Overweight
5 Taiwan   Male   Overweight
6 Taiwan   Male   Overweight
```

## Optional Challenge

Perform all the actions in this problem set with one dplyr call.

```
final <- mega %>%
  #Clean column names and rename location information
  clean_names() %>%
  rename(location = location_information) %>%

  #Calculate BMI with metric vs imperial
  mutate(bmi = if_else(
    location %in% c("United Kingdom", "Taiwan"),
    weight / ((height / 100)^2),          # metric height from cm to m
    703 * weight / ((height * 12)^2)     # imperial height from ft to in
  )) %>%

  #Categorize BMI
  mutate(bmi_cat = case_when(
    bmi < 18.5 ~ "Underweight",
    bmi >= 18.5 & bmi <= 24.9 ~ "Normal",
    bmi >= 25.0 & bmi <= 29.9 ~ "Overweight",
    bmi >= 30.0 ~ "Obese",
  )) %>%

  #Arrange by location then bmi descending
  arrange(location, desc(bmi)) %>%

  #Drop height, weight, and bmi
  select(-c(height, weight, bmi))

head(final)
```

```
# A tibble: 6 x 3
  location gender bmi_cat
  <chr>      <chr> <chr>
1 Colorado Female Obese
2 Colorado Female Obese
3 Colorado Male   Obese
4 Colorado Female Obese
5 Colorado Female Obese
6 Colorado Male   Obese
```