**INTRODUCTION TO DATA SCIENCE**

**R Programming Exercises**

**Member Names:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Year and Section: \_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**College Grades Data**

Remember the college student’s data? We'll compute their final grades.

The grade computation will be:

* **30%:** Average grade of long exams (2 long exams)
* **40%:** Final exam grade (1 final exam)
* **30%:** HW/Machine Problems (3 Homeworks)

**Class policy**

Before the course started, you also have the discussed to them the following policies:

1. The grading scale is:

|  |  |
| --- | --- |
| **Final Grade** | **Final Grade in Percentage** |
| 1.0 | [ 80 – 100 ] |
| 2.0 | [ 70 – 79 ] |
| 3.0 | [ 50 – 69 ] |
| 4.0 | [ 0 – 49 ] |

2. Max absences is 6. Students who exceeded the limit will get a grade of 5.0.

3. Missed exams or HW: 0

4. Cheating will get 0 points on that specific HW/Exam.

**Additional Premise**

1. You will see that almost everyone cheated in **HW 3**, except for **Chrisphee**. They all have the same answers and got **97**. Change the **HW 3** scores of students who cheated to **0**, and change Chrisphee's HW score to **100**.
2. You promised a bonus of **5** points to be added in their final percentage grades.

**1. Data Audit**

**1.1. Import library ‘dplyr’ and load College Students dataset in R**

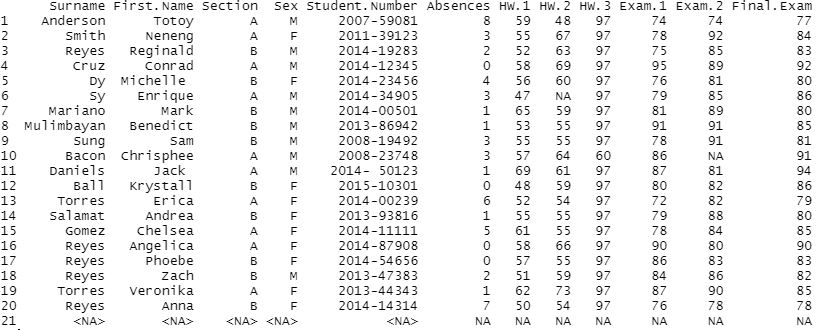
Load the **college\_students\_w\_header.csv** data here and store it as **college\_df.** Check the data frame then see the expected output below.

**library(dplyr)**

**college\_df <- read.csv(file.choose(), header=T)**

**college\_df**

**Expected Output:**

****

1098//8

//24

**1.2. Check how many entries do we have?**

**paste("Number of entries are : ", nrow(college\_df))**

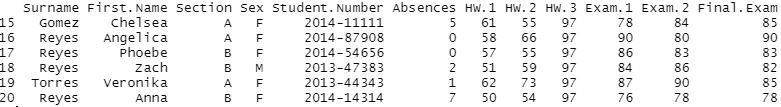
**Expected Output:**

****

**1.3. Remove the entries that have no “Surname” and “First Name”**

**college\_df <- college\_df[!(is.na(college\_df$Surname) & is.na(college\_df$First.Name)), ]**

**Expected Output:**

****

**1.4. How many students did not take the “HW 2”?**

**sum(is.na(college\_df$HW.2))**

**Expected Output:**

****

**1.5. Who are they?**

***college\_df[is.na(college\_df$HW.2),c("Surname","First.Name")]***

**Expected Output:**

****

**1.6. How many students did not take the “Exam 2”?**

**sum(is.na(college\_df$Exam.2))**

**Expected Output:**

****

**1.7. Who are they?**

***college\_df[is.na(college\_df$Exam.2),c("Surname","First.Name")]***

**Expected Output:**

****

**2. Data Cleaning and Processing**

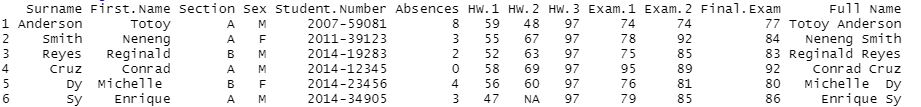
Checklist:

* **Add new columns**
  + “***Full Name***” from “**Surname**” and “**First Name**”, please follow the format <*first\_name*> <*surname*>, example: Chrisphee Bacon
* **Resolve the cheating incident**
  + HW 3 grades should be 0
  + Chrisphee's HW 3 grade should be 100 because of her honestly.
* **Resolve the missed HWs and Exams**
  + HW 2
  + Exam 2

**2.1. Add new columns – “Full Name”**

**college\_df$Full.Name <- paste(college\_df$First.Name, college\_df$Surname)**

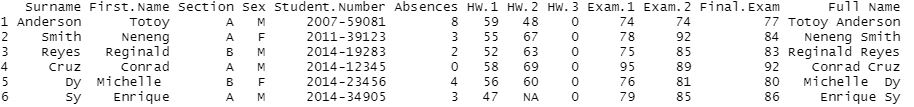
**Expected Output:**

****

**2.2. Perform changes based on the cheating incident.**

**college\_df$HW.3[college\_df$HW.3 == "97"] <- "0"**

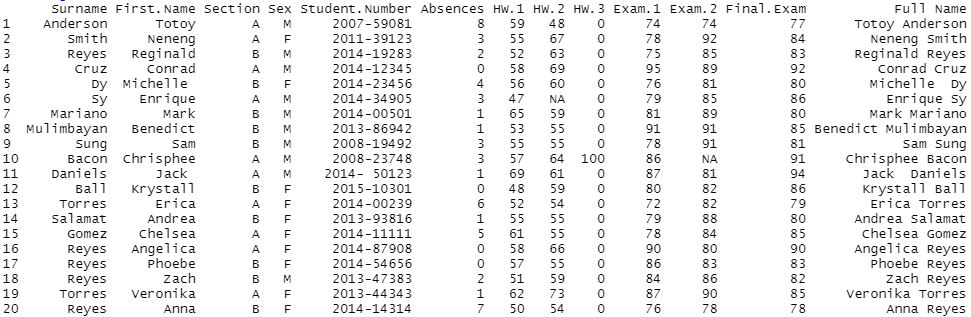
**Expected Output:**

****

**2.3. Chrisphee's HW 3 grade should be 100 because of her honesty.**

**college\_df$HW.3[college\_df$HW.3 == "60"] <- "100"**

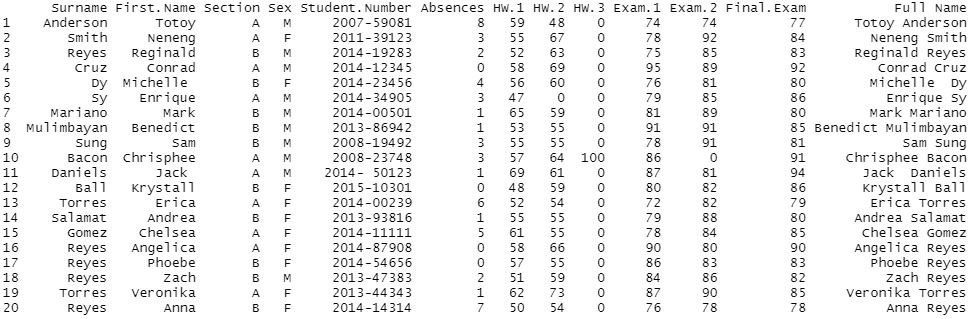
**Expected Output:**

****

**2.4. Resolve missed HW’s and Exams.**

**college\_df[is.na(college\_df)] <- "0"**

**Expected Output:**

****

**2.5. Computation of Grades**

**Checklist:**

* Compute for the final grades. Don't forget to add **5** points in their Final Grade (**%**), you kind-hearted person you.
* Convert their grades based on the scale.
* Set the grade to **5.0** for students with excessive absences.

**2.5.1. Get the average mean of the HWs and Exams.**

Before getting the **“HW mean”** and **“Exam mean”,** check first if these attributes: **‘HW.1’**, **‘HW.2’**, **‘HW.3’**, **‘Exam.1’**, and **‘Exam.2’** are in numeric or integer data type.

Text

Description automatically generated

***typeof(college\_df$HW.1)***

***typeof(college\_df$HW.2)***

***typeof(college\_df$HW.3)***

***typeof(college\_df$Exam.1)***

***typeof(college\_df$Exam.2)***

Just in case, if one of them are in a ***character data type***, this should be converted into **numeric data type**.

***college\_df$HW.2 = as.integer(college\_df$HW.2)***

***college\_df$HW.3 = as.integer(college\_df$HW.3)***

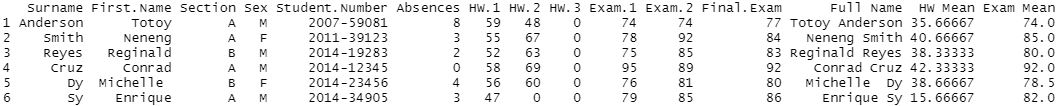
***college\_df$Exam.2 = as.integer(college\_df$Exam.2)***

Get the **mean** and store them in columns **“HW Mean”** and **“Exam Mean”**, respectively.

***college\_df$HW.Mean <- rowMeans(subset(college\_df, select = c(HW.1, HW.2, HW.3)), )***

***college\_df$Exam.Mean <- rowMeans(subset(college\_df, select = c(Exam.1, Exam.2)), )***

**Expected Output:**

****

**2.5.2. Compute Final Grade in Percentage (FG Per)**

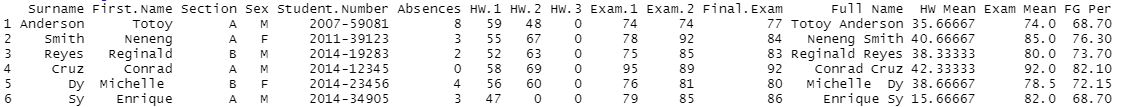
Use the columns **Final Exam**, **HW Mean**, and **Exam Mean**.

For reference:

* **30%**: Mean grade of long exams (2 long exams)
* **40%**: Final exam grade (1 final exam)
* **30%**: HW/Machine Problems (3 homeworks)

***college\_df$FG.Per <- (college\_df$Exam.Mean\* .3 + college\_df$Final.Exam\* .4 + college\_df$HW.Mean\* .3) + 5***

**Expected Output:**

****

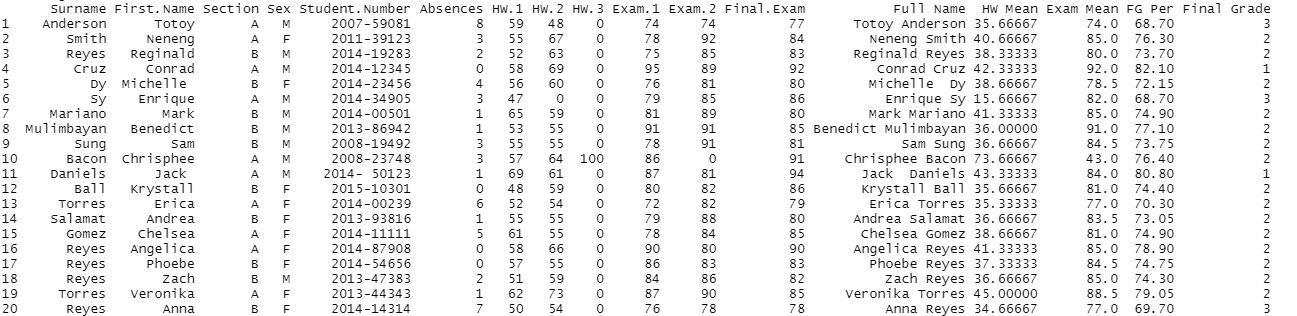
**2.5.3. Get the scaled grades**

This should be based from the **FG Per** column.

For reference:

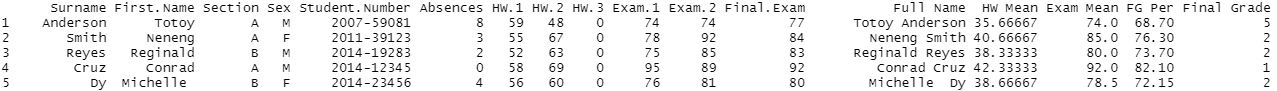
|  |  |
| --- | --- |
| **Final Grade** | **Final Grade in Percentage** |
| 1.0 | [ 80 – 100 ] |
| 2.0 | [ 70 – 79 ] |
| 3.0 | [ 50 – 69 ] |
| 4.0 | [ 0 – 49 ] |

***college\_df$Final.Grade <- ifelse((between(college\_df$FG.Per, 80 , 100)), 1, ifelse((between(college\_df$FG.Per, 70, 79.99)), 2, ifelse((between(college\_df$FG.Per, 50, 69.99)), 3, ifelse((between(college\_df$FG.Per, 0, 49.99)), 4, ""))))***

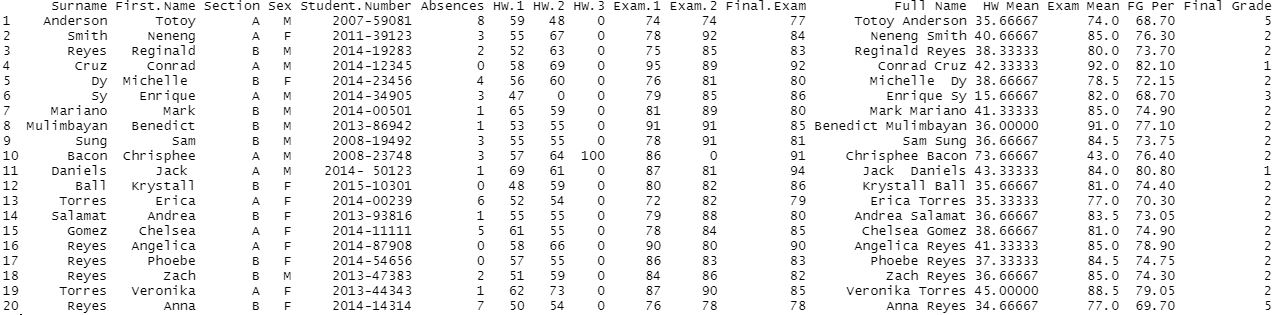
**Expected Output:**

**2.5.4. Resolve excessive absences**

***college\_df$Final.Grade[college\_df$Absences > 6] <- 5***

****

**Expected Output:**

**FINAL OUTPUT:**

**End of Exercise.**