Exam R lectures
August 23, 2016, 9.00; Prof. An Carbonez
At the end of the exam: upload the R script file (with extension .R) on Toledo > Assignment > upload R part.
BE SURE THAT THE FIRST LINE OF YOUR R SCRIPT CONTAINS YOUR NAME

We use the dataframe *diamonds* from the package *ggplot2*.

Variable name	**Description**	
price	Price in $	
carat	Weight of diamond	
cut	Quality of cut (fair, good, very good, premium, ideal)	
color	Diamond colour,from J (worst) to D (best)	
clarity	How clear the diamond is	

1. Use *diamonds* data frame
   - **a.** Create a subset diamonds_small with only the first 5000 observations. **For all later questions, we only work with this diamonds_small data frame (unless specified otherwise).**
   - b. Create a new variable **cut_new** by using the ifelse function in R(see table below)

cut	**Cut_new**
Ideal	Above average
Premium	Above average
Very Good	Very Good
Good	Below average
Fair	Below average

   - c. Create a data frame **diamonds_2** with only those variables specified in the view below.

```
> head(diamonds_2)
  carat color clarity price       cut_new
1  0.23     E     SI2   326 Above average
2  0.21     E     SI1   326 Above average
3  0.23     E     VS1   327 Below average
```

   - d. Write these data from *diamonds_2* to an .txt file with names *diamonds.txt*.

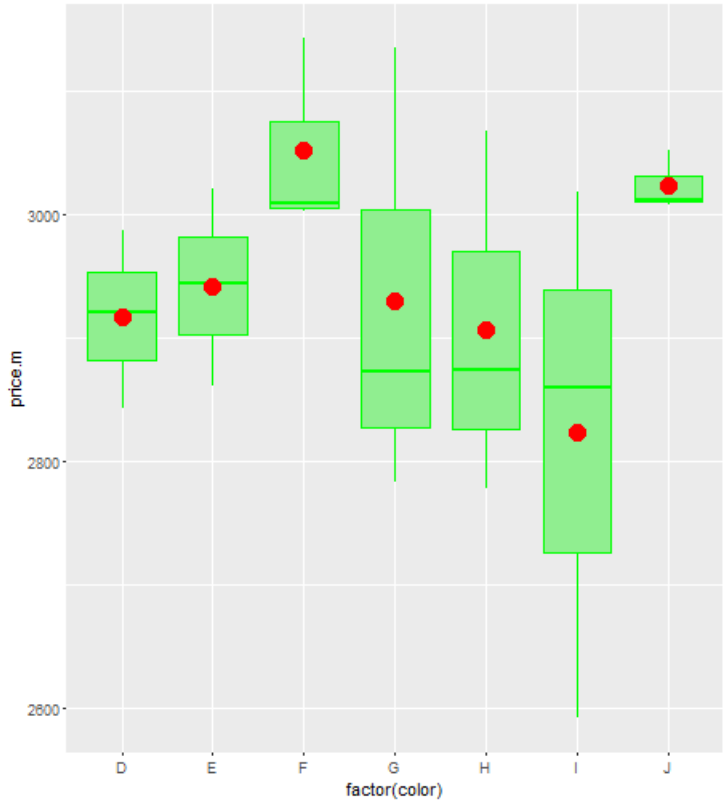
2. Use the data frame **diamonds_2** (in case you were not able to make diamonds_2, then start with diamonds_small data frame)
   - a. Compute the average price for every combination of color and cut_new. Save these averages in a data frame **result**.

b. Create a data frame **combine** by combining the original data frame *diamonds_2* with the *result* data frame so that for every diamond, you see the price and next to it also the average price (based on combination of color and cut_new).

```
> head(combine)
  color       cut_new price  price.m
1     D Above average  3217 2986.286
2     D Above average  2991 2986.286
3     D Above average  3509 2986.286
4     D Above average  3160 2986.286
```

c. Based on the information from ***result*** data frame. Create a green grouped box plot for *price.m* as below with *color* as grouping factor.

d. Add the average of price.m per color as a red dot on each box plot. See below.



3. Write a function *fun1* which produces the text 'Non-negative number' if you apply fun1 to a positive number and 'negative number' if you apply fun1 to a negative number. Apply this function to the values 9 and -13 so that the result looks as :

```
> x <- 9
>  fun1(x)
[1] "Non-negative number"
>  x <- -13
>  fun1(x)
[1] "Negative number"
```