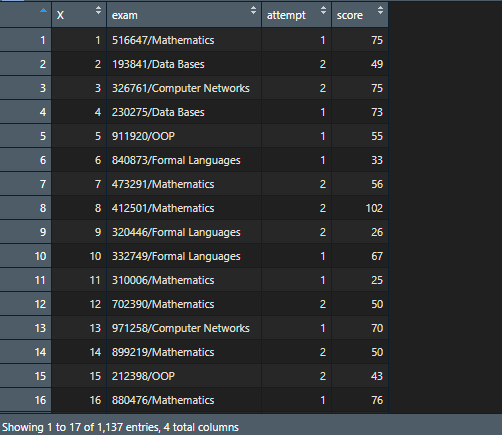
**Aufgabe 1a): Import the files stud.data, exam.data as tibbles.**

**Code:**

**exam\_data <- read.csv("C:/Users/madin/Dropbox/Studium/unterricht folien/3 Semester/Statistic/Statistik2022/exam\_data.csv")**

view (exam\_data)

Ausgabe :



**Aufgabe 1b): Determine the scale and type of all variables.**

mat.nr matriculation number **🡪 Qual. Nominal**

gender sex **🡪 Qual. Nominal**

semester current semester **🡪** **Quant. Ratio**

course course of study **🡪** **Qual. Nominal**

exam matriculation number/exams **🡪 Qual. Nominal**

attempt number of attempt **🡪** **Quant. Ratio**

score achieved score **🡪Quant. Ordinal**

**Aufgabe 1c): Add a variable grade that indicates the grade of the exam. The grade is derived from the score as follows: • Grade 5 if score < 50**

**• Grade 4, if 50 \_ score < 65**

**• Grade 3, if 65 \_ score < 80**

**• Grade 2, if 80 \_ score < 90**

**• Grade 1, if 90 \_ score**

**Code:**

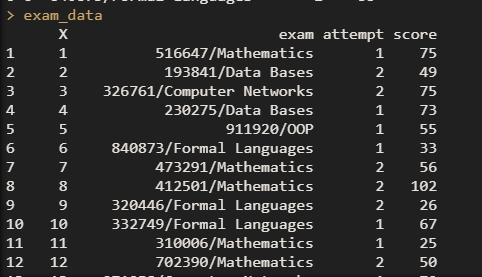
**exam\_data %>% mutate(grade = case\_when( score < 50 ~ 5,**

**score >= 50 & score < 65 ~ 4,**

**score >= 65 & score < 80 ~ 3,**

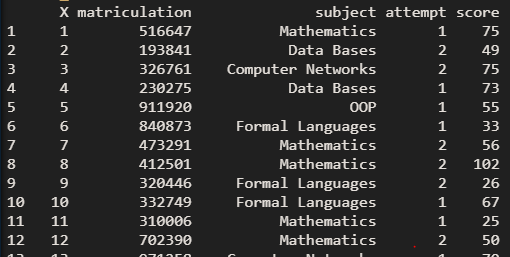
**score >= 80 & score < 90 ~ 2,**

**score >= 90 ~ 1))**

Ausgabe :

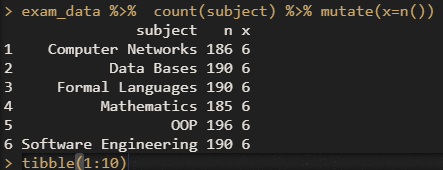
**Aufgabe 1d): Split the variable exam into 2 columns containing the matricula-tion number and the exam subject.**

**Code: exam\_data<- exam\_data %>% separate(exam, into = c("matriculation", "subject"), sep = "/")**

**exam\_data**  
Ausgabe:   
  
  
  
  
\_\_\_\_\_\_\_\_\_

**Aufgabe 1e): Determine the total number of tests in each exam and the number of students articipating**

#total number of tests in each exam

**Code:  exam\_data %>% count(subject) %>% mutate(x=n())**

Ausgabe

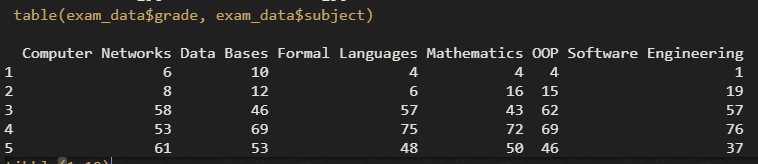
#the number of students participating

**Code: exam\_data %>% select(matriculation) %>% unique %>% summarise(anz.stud=n())**

Ausgabe:

**Aufgabe 1f): For each subject, determine the absolute frequencies of the grades and store the result in a tibble with the variables grade, Computer Networks, Data Bases, Formal Languages, Mathematics, OOP and Software Engineering.**

**Code: table(exam\_data$grade, exam\_data$subject)**

 Ausgabe:

**Aufgabe 1g): For each subject, determine the minimum, maximum, the three quartiles, the mean of the variable score, the number of participants and the dropout rates.**

**Code: exam\_data %>% mutate(fail = if\_else (grade <= 4,0,1)) %>%**

**group\_by(subject) %>% summarise( minimum = min (score),**

**maximum = max(score),**

**q1 = quantile(score, 0,25, type = 1),**

**q2 = quantile(score, 0,5, type = 1),**

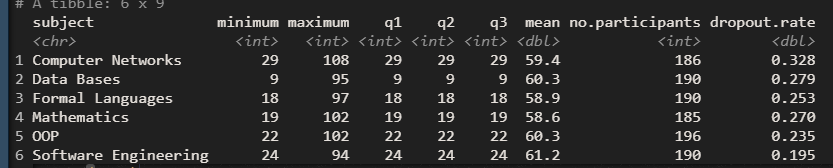
**q3 = quantile(score, 0,75, type = 1),**

**mean = mean(score),**

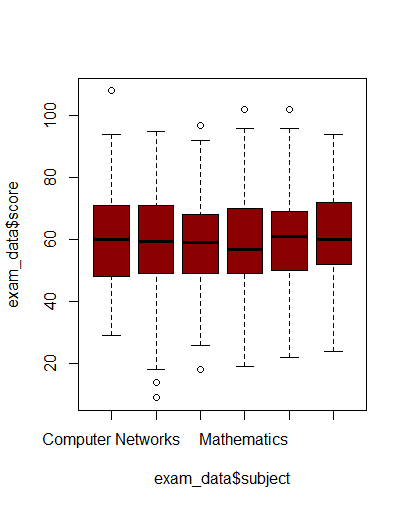
**no.participants = n(),**

**dropout.rate = sum(fail)/no.participants)**

**Ausgabe:**



**Aufgabe 1h): Create side by side boxplots of the score for each subject and interpret the results.**

**Code: boxplot(exam\_data$score~exam\_data$subject,col=c("dark red"))**

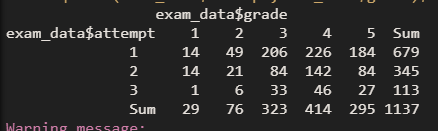
**Ausgabe:**

**Aufgabe 1h): Determine the contingency table of the variables attempt and grade and determine the indifference table and chi-square value..**

**Code: chisq.test(exam\_data$attempt,exam\_data$grade)$observed %>% addmargins()**

**chisq.test(exam\_data$attempt,exam\_data$grade)$expectetd %>% addmargins()**

**chisq.test(exam\_data$attempt,exam\_data$grade)$statistic**

**Ausgabe:**