Name: Shama Shafique *Matrikelnummer: 1136669*

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 :

	X	exam	attempt	score
1	1	516647/Mathematics	1	75
2	2	193841/Data Bases	2	49
3	3	326761/Computer Networks	2	75
4	4	230275/Data Bases	1	73
5	5	911920/OOP	1	55
6	6	840873/Formal Languages	1	33
7	7	473291/Mathematics	2	56
8	8	412501/Mathematics	2	102
9	9	320446/Formal Languages	2	26
10	10	332749/Formal Languages	1	67
11	11	310006/Mathematics	1	25
12	12	702390/Mathematics	2	50
13	13	971258/Computer Networks	1	70
14	14	899219/Mathematics	2	50
15	15	212398/OOP	2	43
16	16	880476/Mathematics	1	76

Showing 1 to 17 of 1,137 entries, 4 total columns

Name: Shama Shafique *Matrikelnummer: 1136669*

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 :

```
> exam_data
      X                        exam attempt score
1     1           516647/Mathematics       1    75
2     2            193841/Data Bases       2    49
3     3     326761/Computer Networks       2    75
4     4            230275/Data Bases       1    73
5     5                  911920/OOP       1    55
6     6      840873/Formal Languages       1    33
7     7           473291/Mathematics       2    56
8     8           412501/Mathematics       2   102
9     9      320446/Formal Languages       2    26
10   10      332749/Formal Languages       1    67
11   11           310006/Mathematics       1    25
12   12           702390/Mathematics       2    50
```

Name: Shama Shafique *Matrikelnummer: 1136669*

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:

	X	matriculation	subject	attempt	score
1	1	516647	Mathematics	1	75
2	2	193841	Data Bases	2	49
3	3	326761	Computer Networks	2	75
4	4	230275	Data Bases	1	73
5	5	911920	OOP	1	55
6	6	840873	Formal Languages	1	33
7	7	473291	Mathematics	2	56
8	8	412501	Mathematics	2	102
9	9	320446	Formal Languages	2	26
10	10	332749	Formal Languages	1	67
11	11	310006	Mathematics	1	25
12	12	702390	Mathematics	2	50

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

```
> exam_data %>% count(subject) %>% mutate(x=n())
              subject   n x
1   Computer Networks 186 6
2          Data Bases 190 6
3    Formal Languages 190 6
4         Mathematics 185 6
5                 OOP 196 6
6 Software Engineering 190 6
> tibble(1:10)
```

#the number of students participating

Code: `exam_data %>% select(matriculation) %>% unique %>% summarise(anz.stud=n())`

Ausgabe:

```
> exam_data %>% select(matriculation) %>% unique %>% summarise(anz.stud=n())
  anz.stud
1      243
```

Name: Shama Shafique *Matrikelnummer: 1136669*

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:

```
table(exam_data$grade, exam_data$subject)

  Computer Networks Data Bases Formal Languages Mathematics OOP Software Engineering
1                 6         10                4           4   4                    1
2                 8         12                6          16  15                   19
3                58         46               57          43  62                   57
4                53         69               75          72  69                   76
5                61         53               48          50  46                   37
```

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:

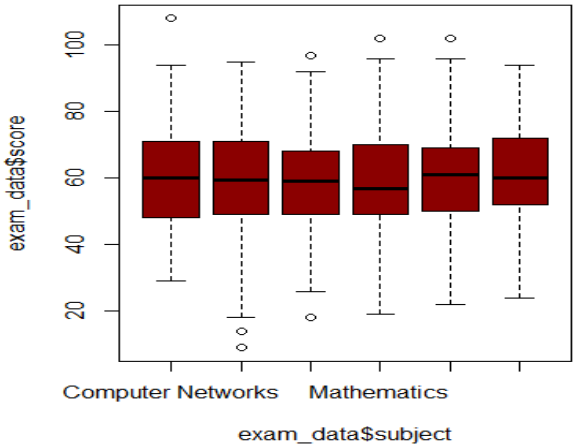
```
# A tibble: 6 x 9
  subject             minimum maximum    q1    q2    q3  mean no.participants dropout.rate
  <chr>                 <int>   <int> <int> <int> <int> <dbl>           <int>        <dbl>
1 Computer Networks        29     108    29    29    29  59.4             186        0.328
2 Data Bases                9      95     9     9     9  60.3             190        0.279
3 Formal Languages         18      97    18    18    18  58.9             190        0.253
4 Mathematics              19     102    19    19    19  58.6             185        0.270
5 OOP                      22     102    22    22    22  60.3             196        0.235
6 Software Engineering     24      94    24    24    24  61.2             190        0.195
```

Name: Shama Shafique *Matrikelnummer: 1136669*

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:

exam_data$attempt	exam_data$grade 1	2	3	4	5	Sum
1	14	49	206	226	184	679
2	14	21	84	142	84	345
3	1	6	33	46	27	113
Sum	29	76	323	414	295	1137

Warning message: