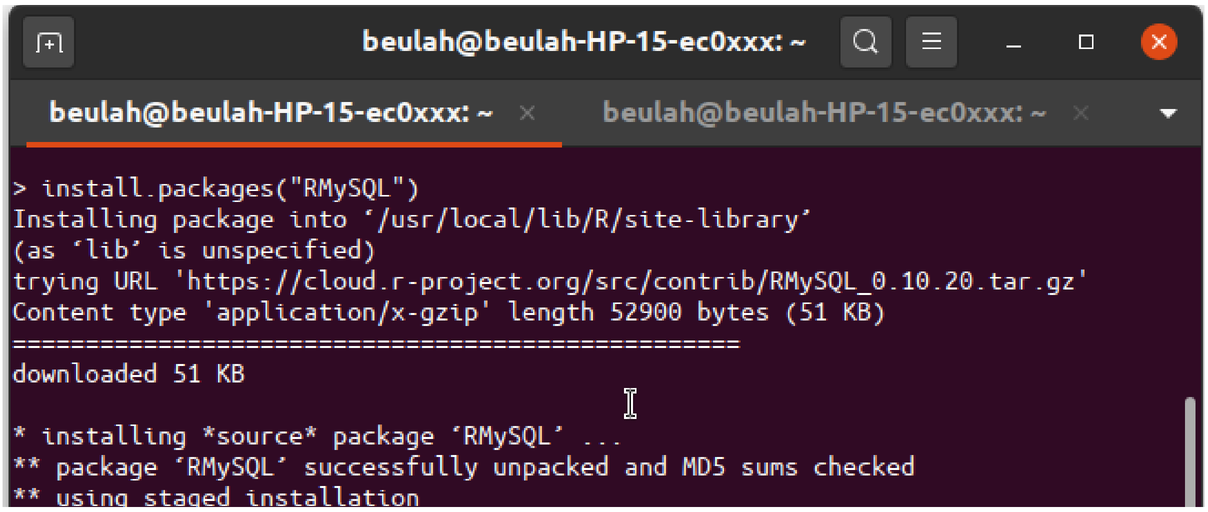
QUESTION 1:

Writing R code snippet that will enable R interaction with Mysql:

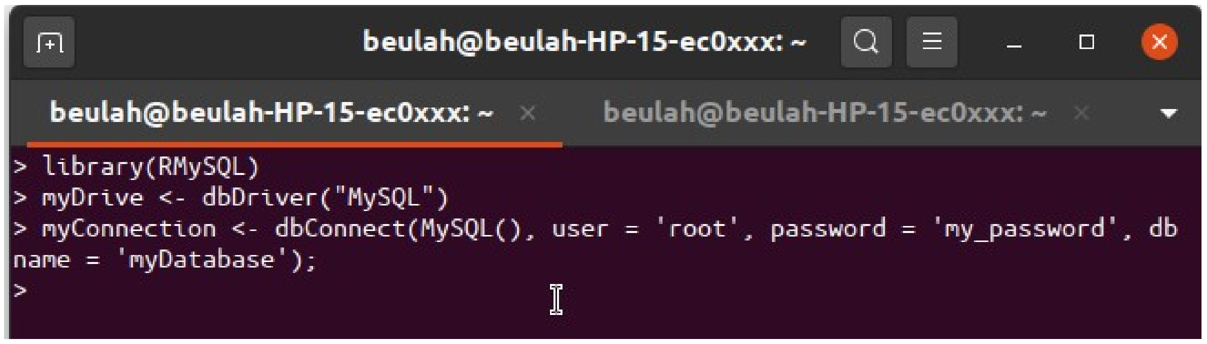
Firstly, I am installing the package “RMySQL” and loading its library in the R console.

> install.packages("RMySQL")


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

```
> install.packages("RMySQL")
Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)
trying URL 'https://cloud.r-project.org/src/contrib/RMySQL_0.10.20.tar.gz'
Content type 'application/x-gzip' length 52900 bytes (51 KB)
==================================================
downloaded 51 KB

* installing *source* package ‘RMySQL’ ...
** package ‘RMySQL’ successfully unpacked and MD5 sums checked
** using staged installation
```

> library(RMySQL)


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

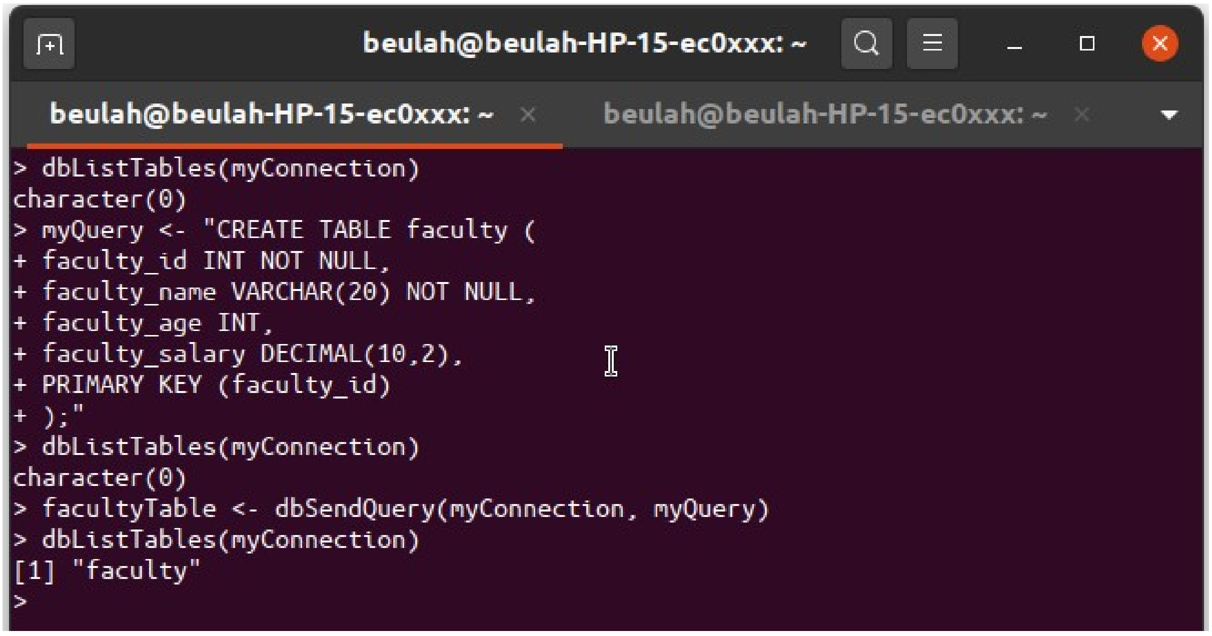
```
> library(RMySQL)
> myDrive <- dbDriver("MySQL")
> myConnection <- dbConnect(MySQL(), user = 'root', password = 'my_password', db
name = 'myDatabase');
>
```

And then I am creating an object for database connection to enable interaction with MySQL using the following command.

myConnection <- dbConnect(MySQL(), user = 'root', password = 'my_password', dbname = 'myDatabase');

Inserting rows in R and displaying the result in R console:

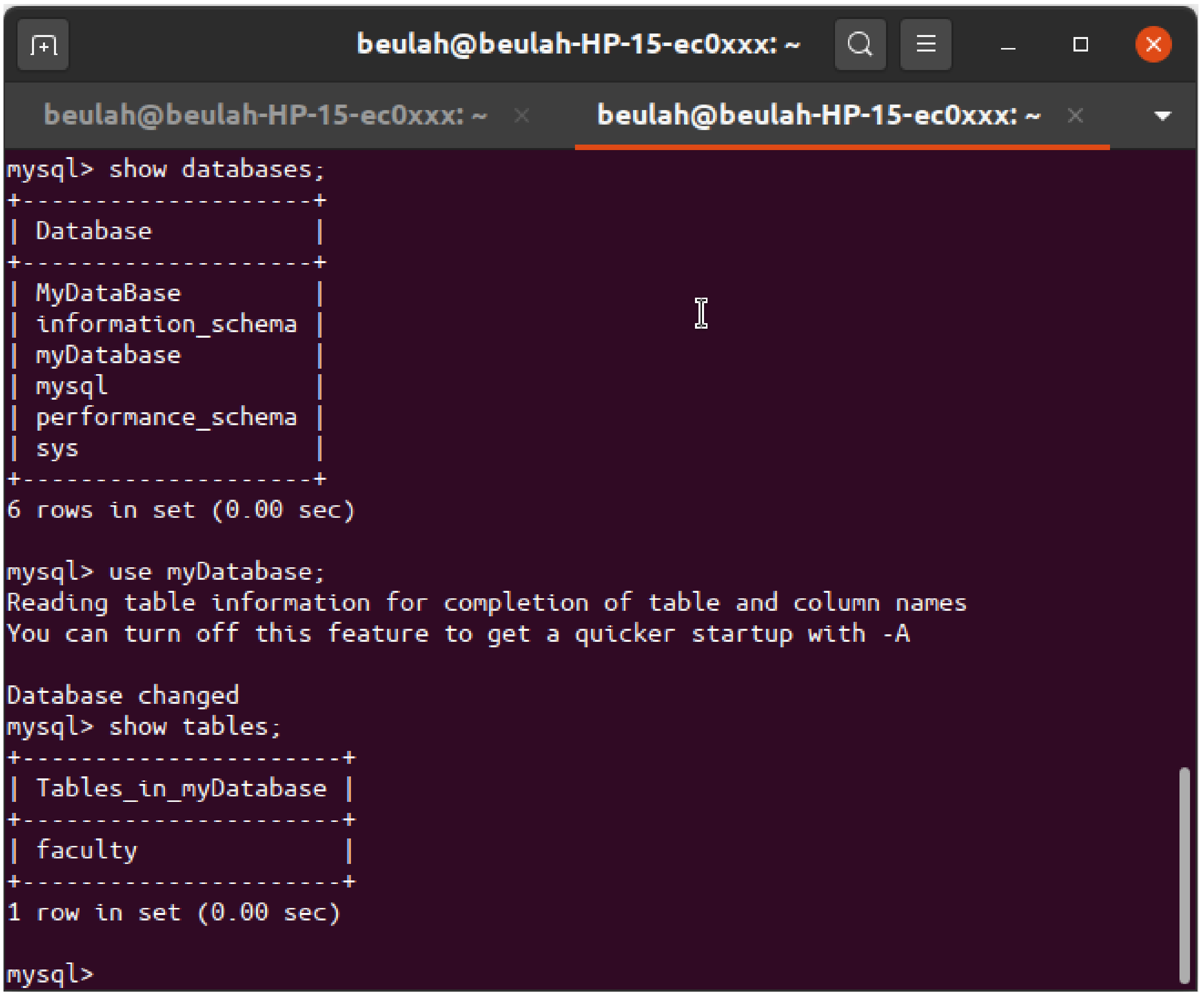
Creating a table called “faculty”:



beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

```
> dbListTables(myConnection)
character(0)
> myQuery <- "CREATE TABLE faculty (
+ faculty_id INT NOT NULL,
+ faculty_name VARCHAR(20) NOT NULL,
+ faculty_age INT,
+ faculty_salary DECIMAL(10,2),
+ PRIMARY KEY (faculty_id)
+ );"
> dbListTables(myConnection)
character(0)
> facultyTable <- dbSendQuery(myConnection, myQuery)
> dbListTables(myConnection)
[1] "faculty"
>
```

Successfully created the faculty table.
It even gets reflected in mysql console.

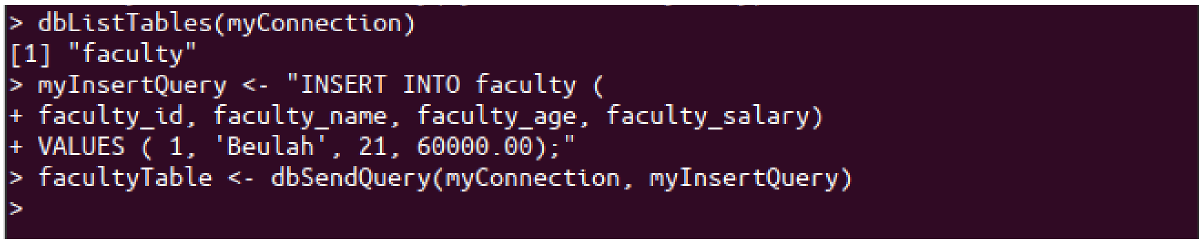


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| MyDataBase         |
| information_schema |
| myDatabase         |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0.00 sec)

mysql> use myDatabase;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+----------------------+
| Tables_in_myDatabase |
+----------------------+
| faculty              |
+----------------------+
1 row in set (0.00 sec)

mysql>
```

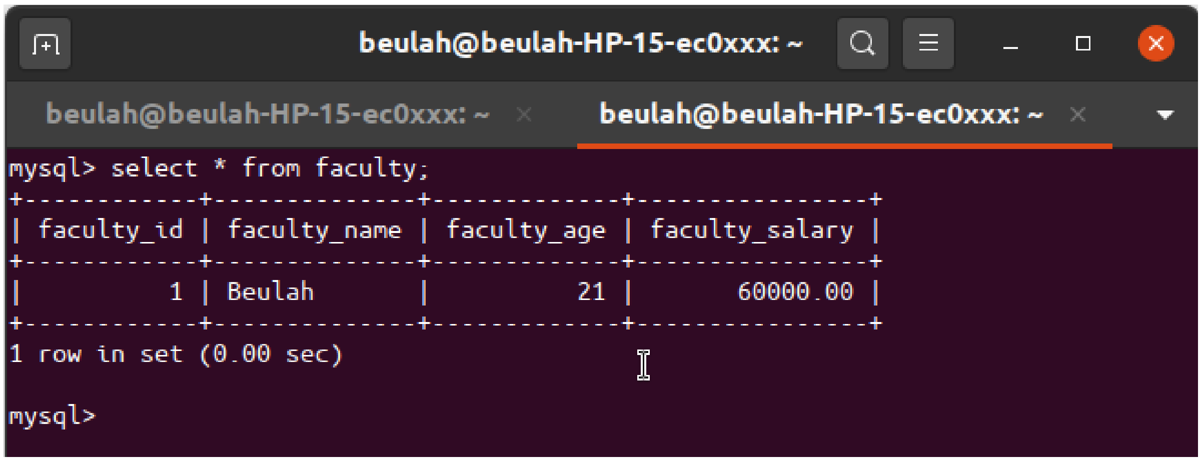
Inserting the data into faculty table:

```
> myInsertQuery <- "INSERT INTO faculty (
+ faculty_id, faculty_name, faculty_age, faculty_salary)
+ VALUES ( 1, 'Beulah', 21, 60000.00);"

> facultyTable <- dbSendQuery(myConnection, myInsertQuery)
```



```
> dbListTables(myConnection)
[1] "faculty"
> myInsertQuery <- "INSERT INTO faculty (
+ faculty_id, faculty_name, faculty_age, faculty_salary)
+ VALUES ( 1, 'Beulah', 21, 60000.00);"
> facultyTable <- dbSendQuery(myConnection, myInsertQuery)
>
```

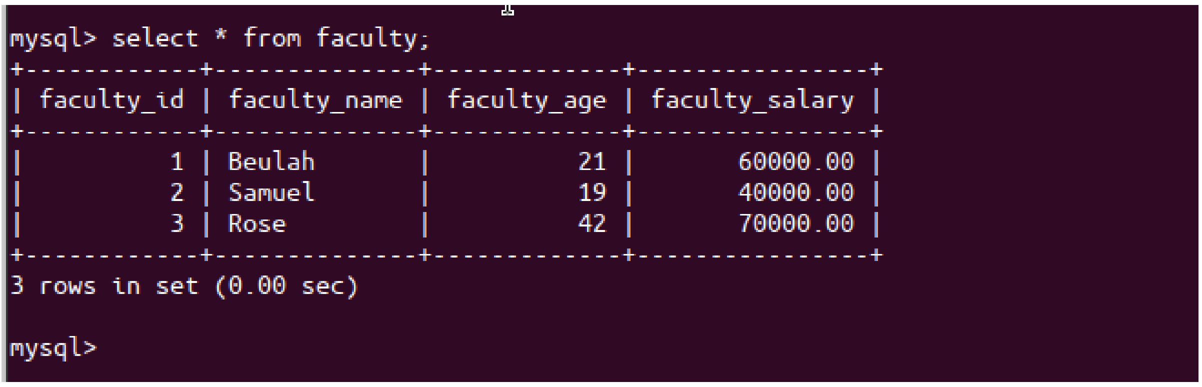
Data for "Beulah" Successfully added using the above R codes.


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

```
mysql> select * from faculty;
+------------+--------------+-------------+----------------+
| faculty_id | faculty_name | faculty_age | faculty_salary |
+------------+--------------+-------------+----------------+
|          1 | Beulah       |          21 |       60000.00 |
+------------+--------------+-------------+----------------+
1 row in set (0.00 sec)

mysql>
```

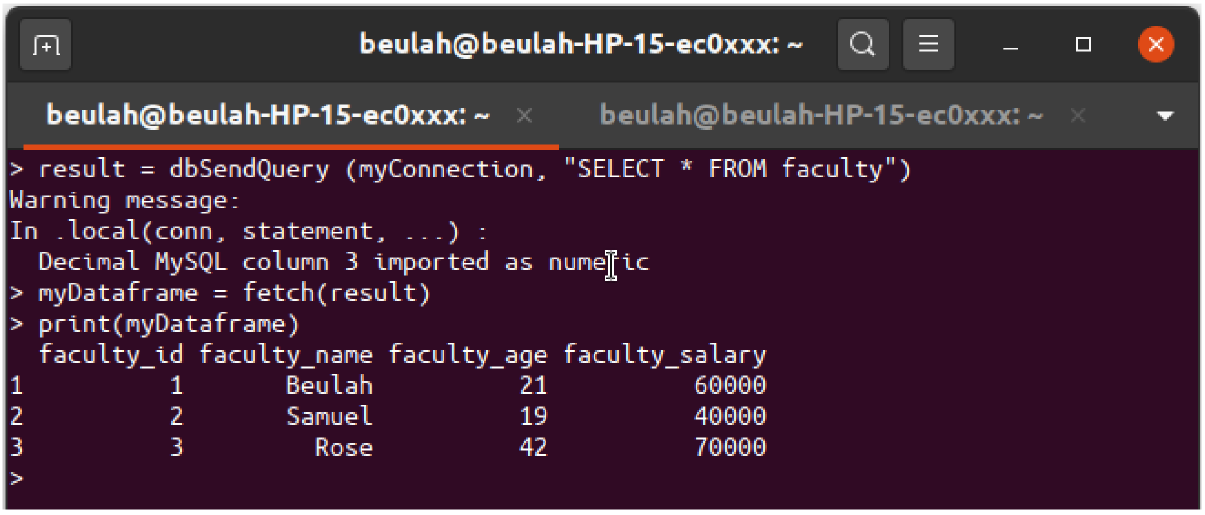
Adding two more rows:

```
> myInsertQuery1 <- "INSERT INTO faculty (
+ faculty_id, faculty_name, faculty_age, faculty_salary)
+ VALUES ( 2, 'Samuel', 19, 40000.00);"

> myInsertQuery2 <- "INSERT INTO faculty (
+ faculty_id, faculty_name, faculty_age, faculty_salary)
+ VALUES ( 3, 'Rose', 42, 70000.00);"

> facultyTable <- dbSendQuery(myConnection, myInsertQuery1)

> facultyTable <- dbSendQuery(myConnection, myInsertQuery2)
```



```
mysql> select * from faculty;
+------------+--------------+-------------+----------------+
| faculty_id | faculty_name | faculty_age | faculty_salary |
+------------+--------------+-------------+----------------+
|          1 | Beulah       |          21 |       60000.00 |
|          2 | Samuel       |          19 |       40000.00 |
|          3 | Rose         |          42 |       70000.00 |
+------------+--------------+-------------+----------------+
3 rows in set (0.00 sec)

mysql>
```

Displaying the result (created rows) in R console:


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~

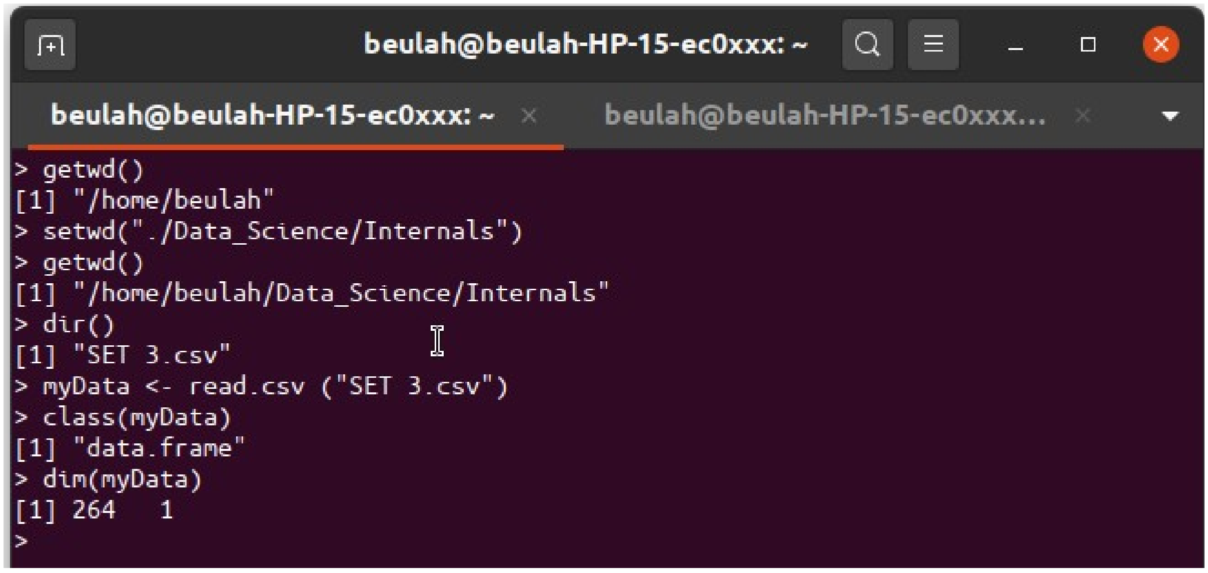
```
> result = dbSendQuery (myConnection, "SELECT * FROM faculty")
Warning message:
In .local(conn, statement, ...) :
  Decimal MySQL column 3 imported as numeric
> myDataframe = fetch(result)
> print(myDataframe)
  faculty_id faculty_name faculty_age faculty_salary
1          1       Beulah          21          60000
2          2       Samuel          19          40000
3          3         Rose          42          70000
>
```

Successfully retrieved the data in “myDataframe” using the following commands.

```
> result = dbSendQuery (myConnection, "SELECT * FROM faculty")
> myDataframe = fetch(result)
> print(myDataframe)
      faculty_id    faculty_name  faculty_age    faculty_salary
1            1            Beulah         21            60000
2            2            Samuel         19            40000
3            3            Rose           42            70000
```

QUESTION 2:

Data Cleaning:


beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx...

```
> getwd()
[1] "/home/beulah"
> setwd("./Data_Science/Internals")
> getwd()
[1] "/home/beulah/Data_Science/Internals"
> dir()
[1] "SET 3.csv"
> myData <- read.csv ("SET 3.csv")
> class(myData)
[1] "data.frame"
> dim(myData)
[1] 264   1
>
```

Reading the dataset "SET 3.csv":

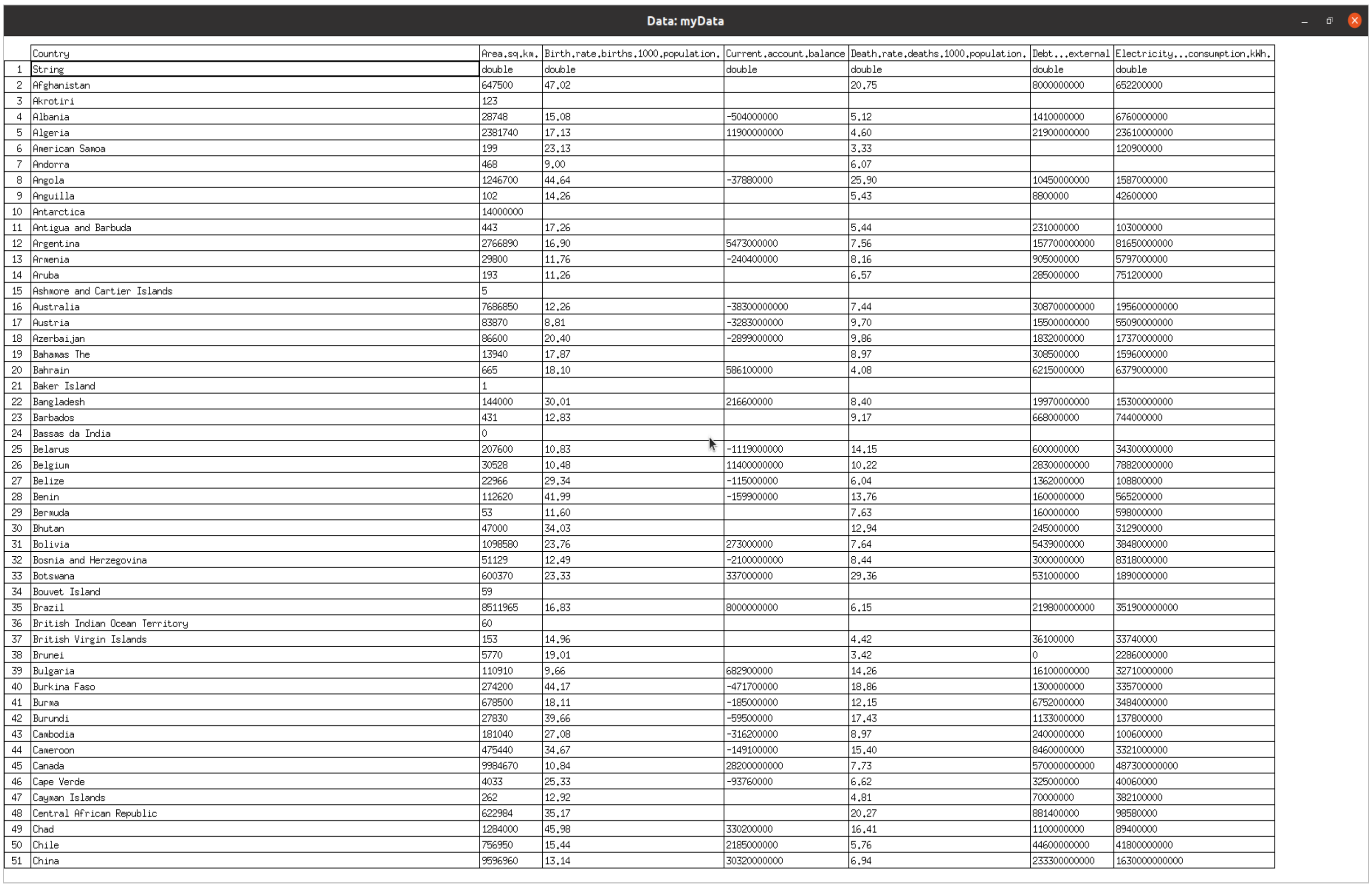
```
> myData <- read.table("SET 3.csv", header = TRUE, sep = ";")

> dim(myData)
[1] 61 45

> summary(myData)
 Country    Area.sq.km. Birth.rate.births.1000.population.
 Afghanistan   : 1         : 1         :12
 Akrotiri     : 1   0      : 1   10.48  : 1
 Albania      : 1   1      : 1   10.83  : 1
 Algeria      : 1   102    : 1   10.84  : 1
 American Samoa: 1   1098580: 1   11.26  : 1
 Andorra      : 1   110910 : 1   11.60  : 1
 (Other)      :55   (Other):55   (Other):44
 Current.account.balance Death.rate.deaths.1000.population.    Debt...external
           :28             :12                             :13
 -1119000000: 1          8.40   : 2                    0          : 1
 -115000000 : 1          8.97   : 2                   10450000000: 1
 -149100000 : 1          10.22  : 1                    1100000000 : 1
 -159900000 : 1          12.15  : 1                    1133000000 : 1
 -1706000000: 1          12.94  : 1                     11600000000: 1
 (Other)    :28          (Other):42                   (Other)    :43
 Electricity...consumption.kWh. Electricity...production.kWh.        Exports
         :12                        :13                        :11
 100600000: 1               10040000000: 1            102700000000: 1
 103000000: 1               106000000  : 1           11470000000 : 1
```

```
108800000: 1           110800000  : 1          1200000     : 1
120900000: 1           117000000  : 1          12760000000 : 1
137800000: 1           122000000  : 1          128000000   : 1
(Other)  :44          (Other)    :43          (Other)     :45
      GDP      GDP...per.capita GDP...real.growth.rate...
        :11         :11              :12
1023000000000: 1   1200   : 2      2.00   : 3
105000000    : 1   1400   : 2      3.00   : 3
112000000    : 1   2000   : 2      3.50   : 3
13010000000  : 1   6500   : 2      5.00   : 3
13650000000  : 1   6600   : 2      3.70   : 2
(Other)      :45   (Other):40      (Other):35
HIV.AIDS...adult.prevalence.rate... HIV.AIDS...deaths
       :21                                 :27
0.10   :11                          100    : 5
0.30   : 4                          200    : 5
0.70   : 3                          1500   : 2
0.20   : 2                          15000  : 2
4.20   : 2                          1000   : 1
(Other):18                          (Other):19
HIV.AIDS...people.living.with.HIV.AIDS  Highways.km.         Imports
       :23                              : 9            :11
10000  : 2                              10217  : 1    5200000000  : 2
100    : 1                              104000 : 1    10030000000 : 1
1100000: 1                              105    : 1    101200000000: 1
12000  : 1                              1100   : 1    1039000000  : 1
13000  : 1                              112998 : 1    1077000000  : 1
(Other):32                              (Other):47    (Other)     :44
Industrial.production.growth.rate...
       :22
3.10   : 3
4.00   : 3
6.00   : 3
1.00   : 2
2.00   : 2
(Other):26
Infant.mortality.rate.deaths.1000.live.births.
       :12
100.44 : 1
12.50  : 1
12.61  : 1
13.37  : 1
15.18  : 1
(Other):44
Inflation.rate..consumer.prices.... Internet.hosts Internet.users
       :12                                 :17            :12
```

```
3.20   : 3                1      : 1    15000  : 3
1.80   : 2                10826  : 1    30000  : 3
1.90   : 2                11     : 1    100000 : 2
2.30   : 2                115158 : 1    5000   : 2
2.40   : 2                118    : 1    60000  : 2
(Other):38                (Other):39    (Other):37
Investment..gross.fixed....of.GDP.   Labor.force
       :29                           :17
19.20  : 2                1026000 : 1
19.80  : 2                10350000: 1
10.20  : 1                1090000 : 1
10.40  : 1                11800000: 1
10.70  : 1                12770   : 1
(Other):25                (Other) :39
Life.expectancy.at.birth.years. Military.expenditures...dollar.figure
       :12                           :21
33.87  : 1              64200000   : 2
36.61  : 1              101300000  : 1
41.01  : 1              11000000000: 1
42.90  : 1              112000000  : 1
43.50  : 1              11600000   : 1
(Other):44              (Other)    :34
Military.expenditures...percent.of.GDP... Natural.gas...consumption.cu.m.
       :21                                 :34
1.30   : 3                    0          : 2
1.80   : 3                    1150000000 : 1
2.60   : 3                    1350000000 : 1
1.50   : 2                    1400000000 : 1
1.60   : 2                    15500000000: 1
(Other):27                    (Other)    :21
Natural.gas...exports.cu.m. Natural.gas...imports.cu.m.
           :34                        :34
0          :18             0          :16
2900000000 : 1             1000000000 : 1
403000000  : 1             1400000000 : 1
57980000000: 1             15400000000: 1
6050000000 : 1             18500000000: 1
(Other)    : 5             (Other)    : 7
Natural.gas...production.cu.m. Natural.gas...proved.reserves.cu.m.
            :34                            :36
0           : 5              104800000000 : 1
10350000000 : 1              132000000000 : 1
1180000000  : 1              150300000000 : 1
165800000000: 1              1691000000000: 1
1731000000  : 1              221700000000 : 1
(Other)     :18              (Other)      :20
```

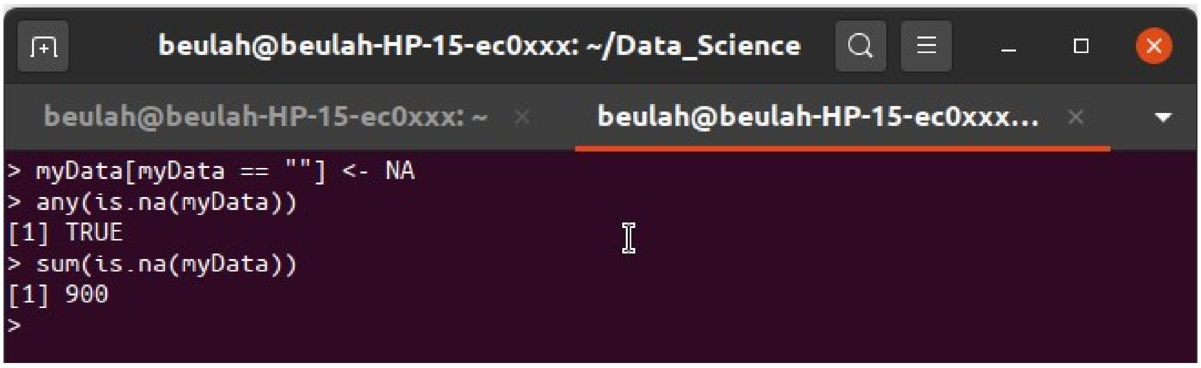
```
Oil...consumption.bbl.day. Oil...exports.bbl.day. Oil...imports.bbl.day.
       :13                       :49                   :51
2400   : 2                0      : 2            1042000: 1
5000   : 2                1370000: 1             221500 : 1
1020   : 1                14500  : 1            2414000: 1
10900  : 1                 199000 : 1            262000 : 1
11500  : 1                 29000  : 1            360000 : 1
(Other):41                 (Other): 6            (Other): 5
Oil...production.bbl.day. Oil...proved.reserves.bbl.      Population
0      :22                          :36                        : 9
       :13                0         : 1            10300483  : 1
1200000: 1                 11870000000: 1            10364388  : 1
1271   : 1                1254000    : 1            11190786  : 1
17550  : 1                1255000000 : 1             1306313812: 1
1788000: 1                 126000000  : 1             13254     : 1
(Other):22                (Other)    :20             (Other)    :47
Public.debt...of.GDP.  Railways.km. Reserves.of.foreign.exchange...gold
       :44                  :32              :28
118.00 : 1           1008   : 1    111100000  : 1
12.80  : 1           1021   : 1    112700000  : 1
17.40  : 1           2706   : 1    11940000000: 1
18.90  : 1           2761   : 1    1206000000 : 1
31.40  : 1           29412  : 1    1214000000 : 1
(Other):12           (Other):24    (Other)    :28
Telephones...main.lines.in.use Telephones...mobile.cellular
        : 9                         :11
38000   : 2                   1000000: 1
6200    : 2                   1050000: 1
0       : 1                   1077000: 1
10000   : 1                   1100000: 1
10815000: 1                    1118000: 1
(Other) :45                   (Other):45
Total.fertility.rate.children.born.woman. Unemployment.rate...
       :12                                   :19
1.72   : 2                              14.80  : 2
1.29   : 1                              30.00  : 2
1.32   : 1                              8.00   : 2
1.36   : 1                              0.00   : 1
1.38   : 1                              0.60   : 1
(Other):43                               (Other):34
```

View(myData)


Data: myData

	Country	Area.sq.km.	Birth.rate.births.1000.population.	Current.account.balance	Death.rate.deaths.1000.population.	Debt...external	Electricity...consumption.kWh.
1	String	double	double	double	double	double	double
2	Afghanistan	647500	47.02		20.75	8000000000	652200000
3	Akrotiri	123					
4	Albania	28748	15.08	-504000000	5.12	1410000000	6760000000
5	Algeria	2381740	17.13	11900000000	4.60	21900000000	23610000000
6	American Samoa	199	23.13		3.33		120900000
7	Andorra	468	9.00		6.07		
8	Angola	1246700	44.64	-37880000	25.90	10450000000	1587000000
9	Anguilla	102	14.26		5.43	8800000	42600000
10	Antarctica	14000000					
11	Antigua and Barbuda	443	17.26		5.44	231000000	103000000
12	Argentina	2766890	16.90	5473000000	7.56	157700000000	81650000000
13	Armenia	29800	11.76	-240400000	8.16	905000000	5797000000
14	Aruba	193	11.26		6.57	285000000	751200000
15	Ashmore and Cartier Islands	5					
16	Australia	7686850	12.26	-38300000000	7.44	308700000000	195600000000
17	Austria	83870	8.81	-3283000000	9.70	15500000000	55090000000
18	Azerbaijan	86600	20.40	-2899000000	9.86	1832000000	17370000000
19	Bahamas The	13940	17.87		8.97	308500000	1596000000
20	Bahrain	665	18.10	586100000	4.08	6215000000	6379000000
21	Baker Island	1					
22	Bangladesh	144000	30.01	216600000	8.40	19970000000	15300000000
23	Barbados	431	12.83		9.17	668000000	744000000
24	Bassas da India	0					
25	Belarus	207600	10.83	-1119000000	14.15	600000000	34300000000
26	Belgium	30528	10.48	11400000000	10.22	28300000000	78820000000
27	Belize	22966	29.34	-115000000	6.04	1362000000	108800000
28	Benin	112620	41.99	-159900000	13.76	1600000000	565200000
29	Bermuda	53	11.60		7.63	160000000	598000000
30	Bhutan	47000	34.03		12.94	245000000	312900000
31	Bolivia	1098580	23.76	273000000	7.64	5439000000	3848000000
32	Bosnia and Herzegovina	51129	12.49	-2100000000	8.44	3000000000	8318000000
33	Botswana	600370	23.33	337000000	29.36	531000000	1890000000
34	Bouvet Island	59					
35	Brazil	8511965	16.83	8000000000	6.15	219800000000	351900000000
36	British Indian Ocean Territory	60					
37	British Virgin Islands	153	14.96		4.42	36100000	33740000
38	Brunei	5770	19.01		3.42	0	2286000000
39	Bulgaria	110910	9.66	682900000	14.26	16100000000	32710000000
40	Burkina Faso	274200	44.17	-471700000	18.86	1300000000	335700000
41	Burma	678500	18.11	-185000000	12.15	6752000000	3484000000
42	Burundi	27830	39.66	-59500000	17.43	1133000000	137800000
43	Cambodia	181040	27.08	-316200000	8.97	2400000000	100600000
44	Cameroon	475440	34.67	-149100000	15.40	8460000000	3321000000
45	Canada	9984670	10.84	28200000000	7.73	570000000000	487300000000
46	Cape Verde	4033	25.33	-93760000	6.62	325000000	40060000
47	Cayman Islands	262	12.92		4.81	70000000	382100000
48	Central African Republic	622984	35.17		20.27	881400000	98580000
49	Chad	1284000	45.98	330200000	16.41	1100000000	89400000
50	Chile	756950	15.44	2185000000	5.76	44600000000	41800000000
51	China	9596960	13.14	30320000000	6.94	233300000000	1630000000000

Checking for missing values in the entire dataframe:


beulah@beulah-HP-15-ec0xxx: ~/Data_Science
beulah@beulah-HP-15-ec0xxx: ~
beulah@beulah-HP-15-ec0xxx...

```
> myData[myData == ""] <- NA
> any(is.na(myData))
[1] TRUE
> sum(is.na(myData))
[1] 900
>
```