


<p>Nama: (Isi Nama Anda)</p> <p>NIM: (Isi NIM Anda)</p>	 <p>Praktikum Statistika</p>	<p>MODUL 2</p> <p>Nama Dosen: Dedy Sugiarto</p>
<p>Hari/Tanggal: Hari, Tanggal Bulan 2023</p>		<p>Nama Asisten Labratorium:</p> <ol style="list-style-type: none"> Elen Fadilla Estri 064002000008 Rukhy Zaifa Aduhalim 064002000041

Tipe Data, Filter Data & Koneksi ke Database

1. Teori Singkat

R (juga dikenal sebagai GNU S) adalah bahasa pemrograman dan perangkat lunak untuk analisis statistika dan grafik. R dibuat oleh Ross Ihaka dan Robert Gentleman di Universitas Auckland, Selandia Baru, dan kini dikembangkan oleh R Development Core Team, di mana Chambers merupakan anggotanya. R dinamakan sebagian setelah nama dua pembuatnya (Robert Gentleman dan Ross Ihaka), dan sebagian sebagian dari permainan nama dari S.

Bahasa R kini menjadi standar de facto di antara statistikawan untuk pengembangan perangkat lunak statistika, serta digunakan secara luas untuk pengembangan perangkat lunak statistika dan analisis data. R merupakan bagian dari proyek GNU. Kode sumbernya tersedia secara bebas di bawah Lisensi Publik Umum GNU, dan versi biner prekompilasinya tersedia untuk berbagai sistem operasi. R menggunakan antarmuka baris perintah, meski beberapa antarmuka pengguna grafik juga tersedia.

R menyediakan berbagai teknik statistika (permodelan linier dan nonlinier, uji statistik klasik, analisis deret waktu, klasifikasi, klusterisasi, dan sebagainya) serta grafik. R, sebagaimana S, dirancang sebagai bahasa komputer sebenarnya, dan mengizinkan penggunaannya untuk menambah fungsi tambahan dengan mendefinisikan fungsi baru. Kekuatan besar dari R yang

lain adalah fasilitas grafiknya, yang menghasilkan grafik dengan kualitas publikasi yang dapat memuat simbol matematika. R memiliki format dokumentasi seperti LaTeX, yang digunakan untuk menyediakan dokumentasi yang lengkap, baik secara daring (dalam berbagai format) maupun secara cetakan.

RStudio merupakan integrated development environment (IDE) khusus bagi bahasa pemrograman R. Software ini menyediakan R console, code editor dengan syntax highlighting, code completion dan direct execution, environment, history, connections, dan fitur-fitur tambahan lainnya seperti file manager, packages manager, help, plot viewer, hingga project versioning menggunakan git. RStudio sebenarnya memiliki dua versi, yaitu open source (gratis) dan commercial edition (berbayar). RStudio juga tidak hanya terbatas dalam bentuk aplikasi dekstop, melainkan terdapat versi RStudio Server, yaitu RStudio yang dapat diakses melalui browser yang terhubung dengan suatu jaringan komputer. Untuk saat ini, versi RStudio yang akan dijelaskan hanyalah RStudio open source berbasis dekstop saja.

2. Alat dan Bahan

Hardware : Laptop/PC

Software : R Studio

3. Elemen Kompetensi

Terdapat beberapa tipe data di R antara lain vektor, matriks dan data frame. Cantumkan setiap output yang dihasilkan dari console Rstudio, ke kolom yang sudah disediakan.

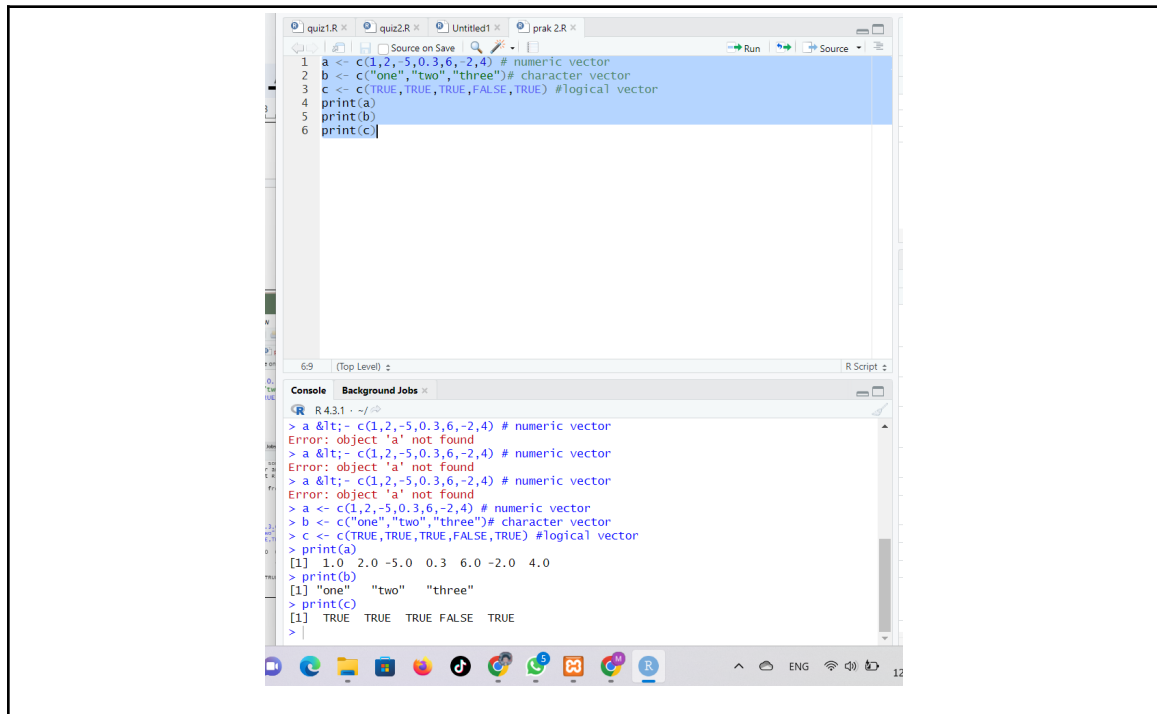
a. Latihan pertama – Vektor

1. Tuliskan Perintah berikut ini di window R Script:

```
a <- c(1,2,-5,0.3,6,-2,4) # numeric vector  
b <- c("one","two","three") # character vector  
c <- c(TRUE,TRUE,TRUE,FALSE,TRUE) #logical vector  
print(a)  
print(b)  
print(c)
```

Output:





```

1 a <- c(1,2,-5,0.3,6,-2,4) # numeric vector
2 b <- c("one","two","three") # character vector
3 c <- c(TRUE,TRUE,TRUE,FALSE,TRUE) #logical vector
4 print(a)
5 print(b)
6 print(c)

```

```

> a &lt;- c(1,2,-5,0.3,6,-2,4) # numeric vector
Error: object 'a' not found
> a &lt;- c(1,2,-5,0.3,6,-2,4) # numeric vector
Error: object 'a' not found
> a &lt;- c(1,2,-5,0.3,6,-2,4) # numeric vector
Error: object 'a' not found
> a <- c(1,2,-5,0.3,6,-2,4) # numeric vector
> b <- c("one","two","three") # character vector
> c <- c(TRUE,TRUE,TRUE,FALSE,TRUE) #logical vector
> print(a)
[1] 1.0 2.0 -5.0 0.3 6.0 -2.0 4.0
> print(b)
[1] "one" "two" "three"
> print(c)
[1] TRUE TRUE TRUE FALSE TRUE
>

```

b. Latihan Kedua – Matriks

1. Seluruh kolom dalam sebuah matriks harus memiliki tipe yang sama (numerik semua, karakter semua, dll) dan memiliki panjang yang sama.

***gunakan nama variable dengan nama anda masing-masing**

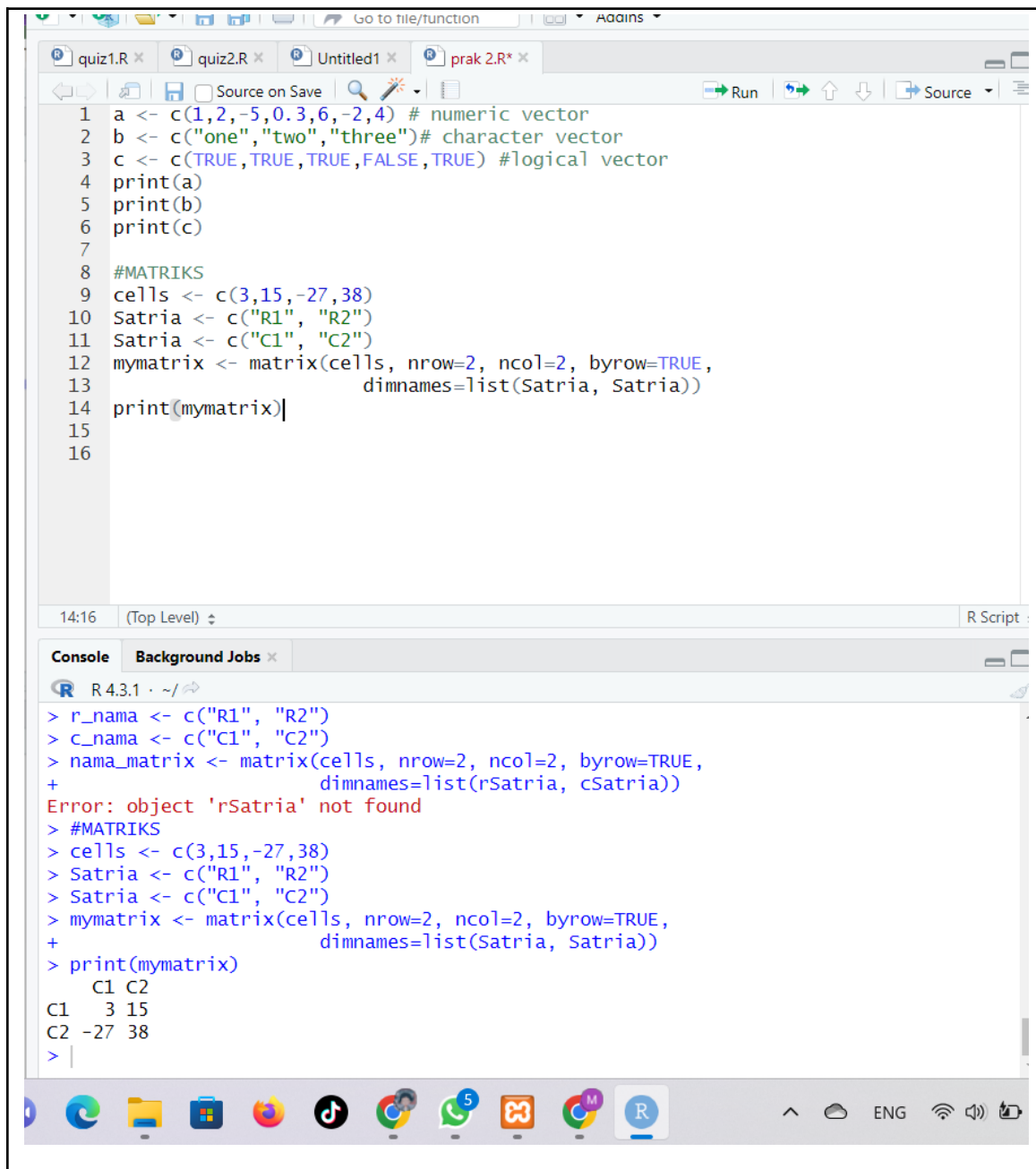
```

#MATRIKS
cells <- c(3,15,-27,38)
r_nama <- c("R1", "R2")
c_nama <- c("C1", "C2")
nama_matrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
                        dimnames=list(r_nama, c_nama))
print(nama_matrix)

```

Output:





The screenshot shows an R Studio interface. The top pane contains a script with the following code:

```
1 a <- c(1,2,-5,0.3,6,-2,4) # numeric vector
2 b <- c("one","two","three") # character vector
3 c <- c(TRUE,TRUE,TRUE,FALSE,TRUE) #logical vector
4 print(a)
5 print(b)
6 print(c)
7
8 #MATRIKS
9 cells <- c(3,15,-27,38)
10 Satria <- c("R1", "R2")
11 Satria <- c("C1", "C2")
12 mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
13                     dimnames=list(Satria, Satria))
14 print(mymatrix)
15
16
```

The bottom pane shows the console output for the same script:

```
> r_nama <- c("R1", "R2")
> c_nama <- c("C1", "C2")
> nama_matrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
+                       dimnames=list(rSatria, cSatria))
Error: object 'rSatria' not found
> #MATRIKS
> cells <- c(3,15,-27,38)
> Satria <- c("R1", "R2")
> Satria <- c("C1", "C2")
> mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
+                   dimnames=list(Satria, Satria))
> print(mymatrix)
      C1 C2
C1     3 15
C2    -27 38
> |
```

c. Latihan Ketiga – Data Frame

1. Mengubah data input menjadi data frame

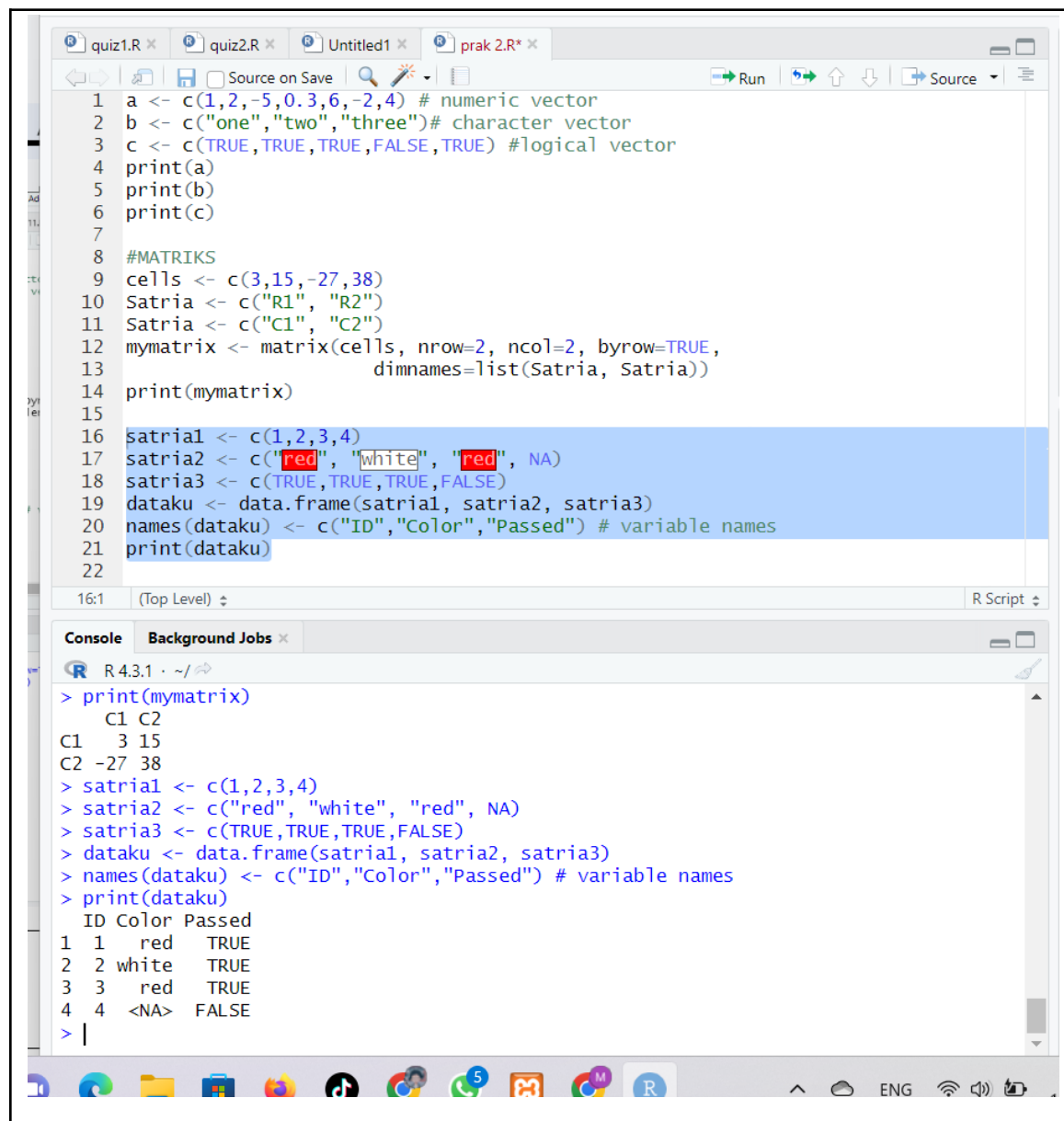
**gunakan nama variable dengan nama anda masing-masing*

```
nama1 <- c(1,2,3,4)
```



```
nama2 <- c("red", "white", "red", NA)
nama3 <- c(TRUE,TRUE,TRUE,FALSE)
dataku <- data.frame(nama1, nama2, nama3)
names(dataku) <- c("ID","Color","Passed") # variable names
print(dataku)
```

Output:



The screenshot shows an R Studio interface with a script editor and a console. The script editor contains R code for creating vectors, a matrix, and a data frame. The console shows the output of the code, including the matrix and the data frame.

```
1 a <- c(1,2,-5,0.3,6,-2,4) # numeric vector
2 b <- c("one","two","three")# character vector
3 c <- c(TRUE,TRUE,TRUE,FALSE,TRUE) #logical vector
4 print(a)
5 print(b)
6 print(c)
7
8 #MATRIKS
9 cells <- c(3,15,-27,38)
10 Satria <- c("R1", "R2")
11 Satria <- c("C1", "C2")
12 mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
13                     dimnames=list(Satria, Satria))
14 print(mymatrix)
15
16 satrial <- c(1,2,3,4)
17 satria2 <- c("red", "white", "red", NA)
18 satria3 <- c(TRUE,TRUE,TRUE,FALSE)
19 dataku <- data.frame(satrial, satria2, satria3)
20 names(dataku) <- c("ID","Color","Passed") # variable names
21 print(dataku)
22
```

Console Output:

```
> print(mymatrix)
  C1 C2
C1  3 15
C2 -27 38

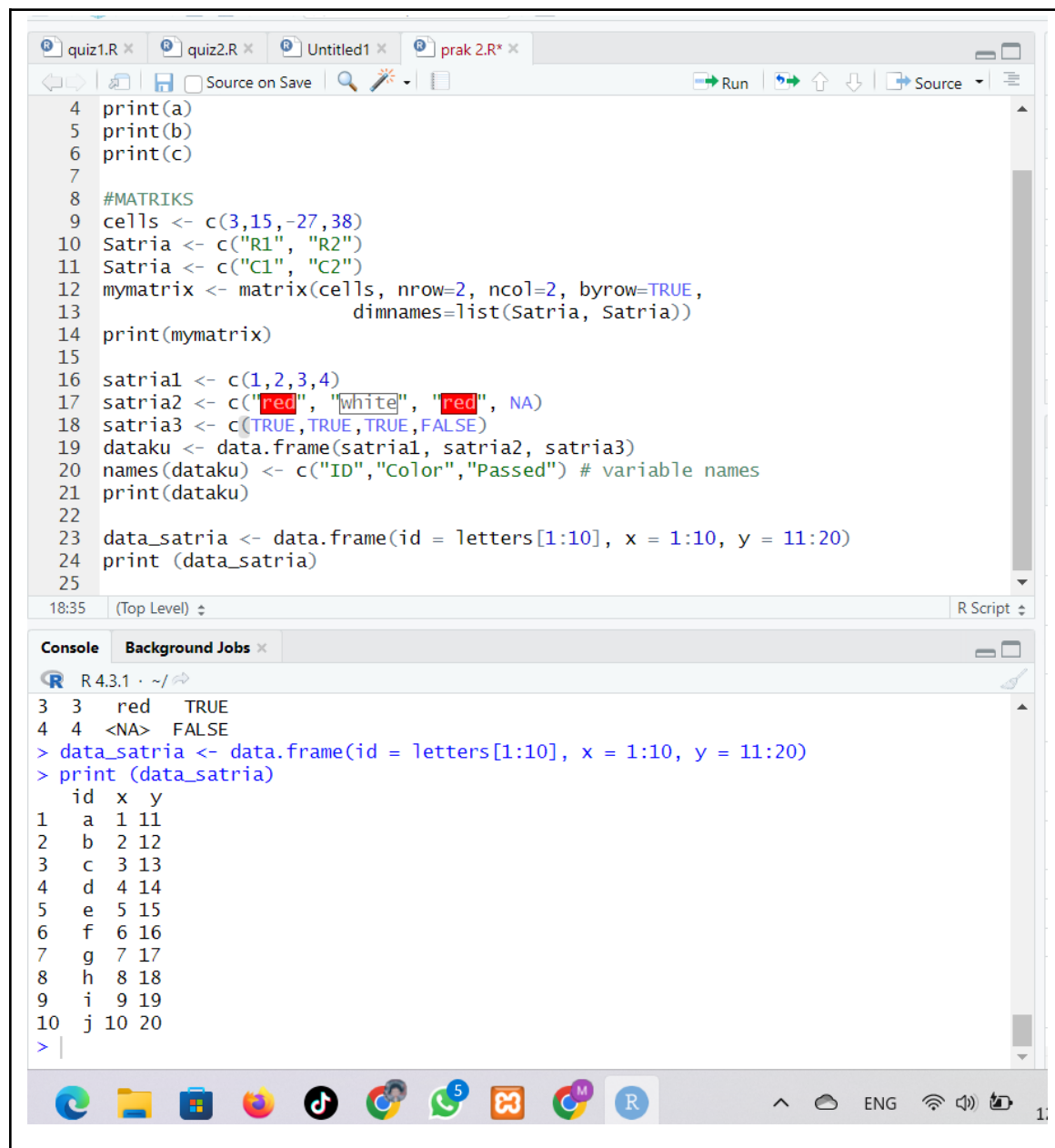
> satrial <- c(1,2,3,4)
> satria2 <- c("red", "white", "red", NA)
> satria3 <- c(TRUE,TRUE,TRUE,FALSE)
> dataku <- data.frame(satrial, satria2, satria3)
> names(dataku) <- c("ID","Color","Passed") # variable names
> print(dataku)
  ID Color Passed
1  1  red   TRUE
2  2 white  TRUE
3  3  red   TRUE
4  4 <NA> FALSE
```



2. Selanjutnya ketikkan perintah dibawah ini

```
data_nama <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
print (data_nama)
```

Output:



The screenshot shows an R Studio interface with a script editor and a console. The script editor contains R code for creating a data frame and a matrix. The console shows the output of the code, including the creation of a data frame with 10 rows and 3 columns.

```
4 print(a)
5 print(b)
6 print(c)
7
8 #MATRIKS
9 cells <- c(3,15,-27,38)
10 Satria <- c("R1", "R2")
11 Satria <- c("C1", "C2")
12 mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
13                     dimnames=list(Satria, Satria))
14 print(mymatrix)
15
16 satria1 <- c(1,2,3,4)
17 satria2 <- c("red", "white", "red", NA)
18 satria3 <- c(TRUE,TRUE,TRUE,FALSE)
19 dataku <- data.frame(satria1, satria2, satria3)
20 names(dataku) <- c("ID","Color","Passed") # variable names
21 print(dataku)
22
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
24 print (data_satria)
25
```

Console Output:

```
R 4.3.1 ~ /
3 3 red TRUE
4 4 <NA> FALSE
> data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
> print (data_satria)
  id x y
1  a 1 11
2  b 2 12
3  c 3 13
4  d 4 14
5  e 5 15
6  f 6 16
7  g 7 17
8  h 8 18
9  i 9 19
10 j 10 20
>
```



Beberapa fungsi penting terkait data frame :

head() - shows first 6 rows

tail() - shows last 6 rows

dim() - returns the dimensions of data frame (i.e. number of rows and number of columns)

nrow() - number of rows

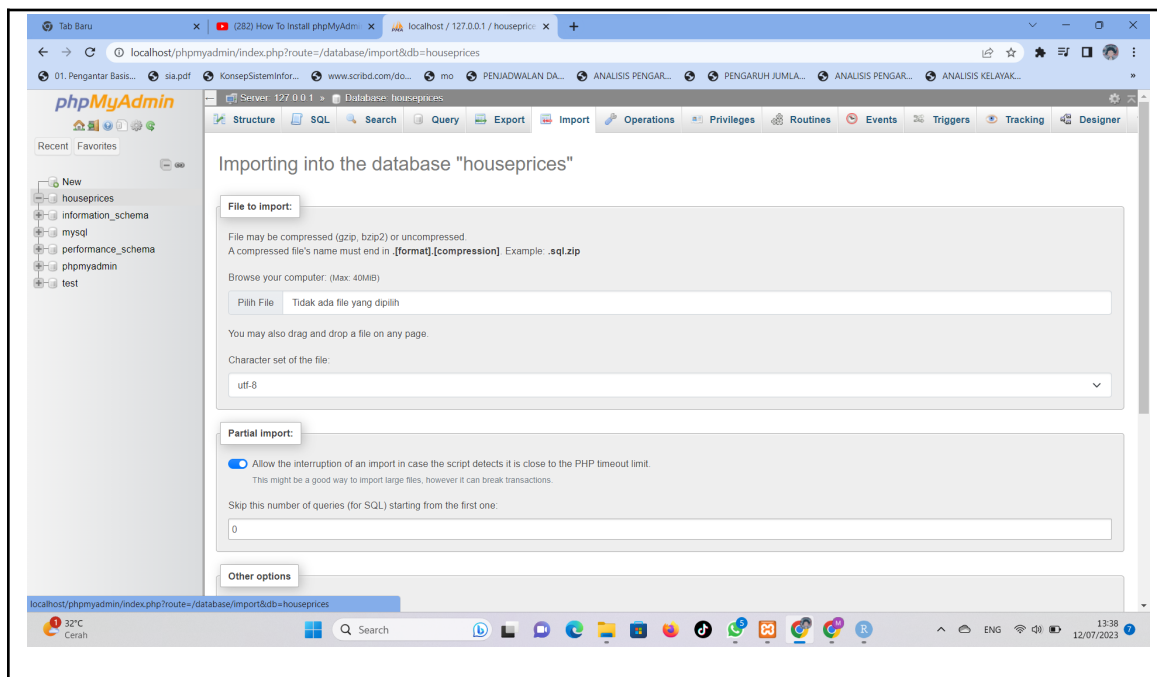
ncol() - number of columns

str() - structure of data frame - name, type and preview of data in each column

names() - shows the names attribute for a data frame, which gives the column names.

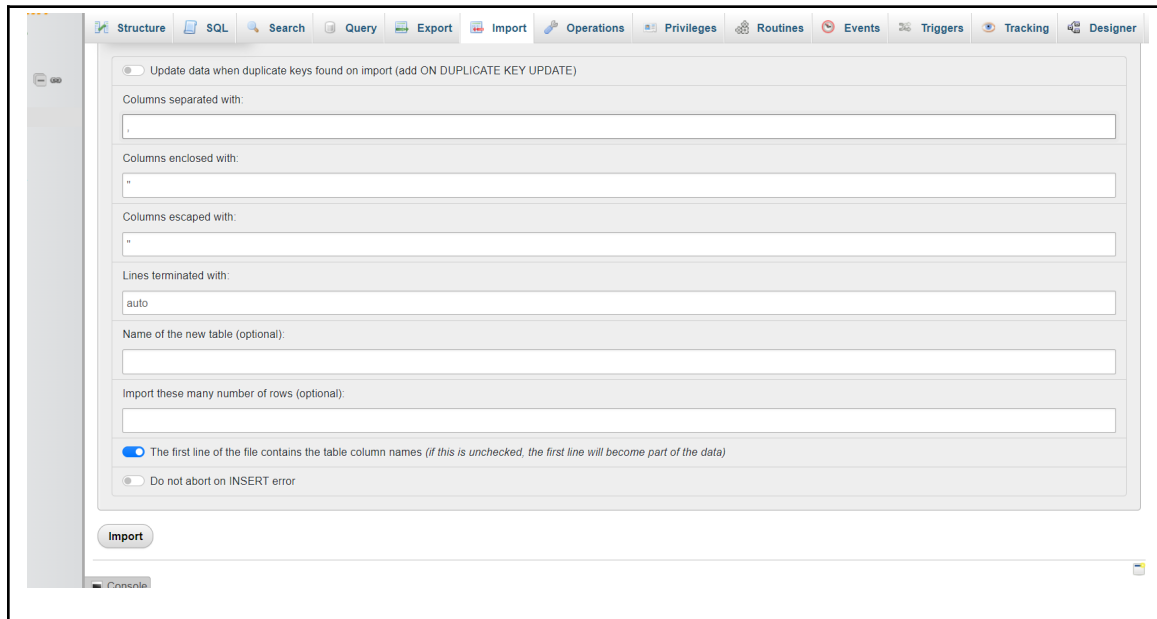
d. Latihan Keempat – Koneksi ke Database

1. Buat sebuah nama database terlebih dahulu dengan nama houseprices di phpmyadmin, Lalu klik menu import



2. Pilih file yang ingin di import ke database (untuk file nama_excel.csv), Ceklis the first line of the file contains the table column name untuk membuat baris pertama pada file excel tersebut menjadi nama atributnya atau nama kolomnya pada database.





The screenshot shows the MySQL Import Wizard dialog box. The 'Import' tab is selected in the top menu. The dialog contains the following fields and options:

- ☐ Update data when duplicate keys found on import (add ON DUPLICATE KEY UPDATE)
- Columns separated with:
- Columns enclosed with:
- Columns escaped with:
- Lines terminated with:
- Name of the new table (optional):
- Import these many number of rows (optional):
- ☒ The first line of the file contains the table column names (if this is unchecked, the first line will become part of the data)
- ☐ Do not abort on INSERT error

At the bottom, there is an 'Import' button and a 'Cancel' button.

3. Klik go, Rename nama tabel sesuai dengan nama anda



The screenshot shows the phpMyAdmin interface. The top panel displays the 'Move table to (database.table)' dialog, where the table 'houseprices' is being moved to the database 'satriapratama'. The bottom panel shows the table view for 'satriapratama', displaying 128 rows of data. The table has columns: Price, SqFt, Bedrooms, Bathrooms, Offers, Brick, and Neighborhood.

Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
114300	1790	2	2	2	No	East
114200	2030	4	2	3	No	East
114800	1740	3	2	1	No	East
94700	1980	3	2	3	No	East
119800	2130	3	3	3	No	East
114600	1780	3	2	2	No	North
151600	1830	3	3	3	Yes	West
150700	2160	4	2	2	No	West
119200	2110	4	2	3	No	East
104000	1730	3	3	3	No	East
132500	2030	3	2	3	Yes	East
123000	1870	2	2	2	Yes	East
102600	1910	3	2	4	No	North
126300	2150	3	3	5	Yes	North
176800	2590	4	3	4	No	West
144000	1780	4	2	1	No	West

4. Kembali ke R Studio, lalu instal dahulu library yang dibutuhkan pada RStudio. Jika belum tersedia, maka lakukan instruksi:

install.packages("RMySQL")



```

Console Background Jobs x
R 4.3.1 · ~/
also installing the dependency 'DBI'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.3/DBI_1.1.3.zip'
Content type 'application/zip' length 774701 bytes (756 KB)
downloaded 756 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.3/RMySQL_0.10.25.zip'
Content type 'application/zip' length 475309 bytes (464 KB)
downloaded 464 KB

package 'DBI' successfully unpacked and MD5 sums checked
package 'RMySQL' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\satria\AppData\Local\Temp\RtmpULh3wt\downloaded_packages
> |

```

5. Lalu jalankan perintah dibawah ini

```

library(RMySQL)
con = dbConnect(MySQL(), user = 'root', password = "", dbname = 'houseprices', host =
'localhost')
dbListTables(con)
myQuery <- "select * from nama;"
df <- dbGetQuery(con, myQuery)

```

Output:



```
11 Satria <- c("C1", "C2")
12 mymatrix <- matrix(cells, nrow=2, ncol=2, byrow=TRUE,
13                     dimnames=list(Satria, Satria))
14 print(mymatrix)
15
16 satria1 <- c(1,2,3,4)
17 satria2 <- c("red", "white", "red", NA)
18 satria3 <- c(TRUE,TRUE,TRUE,FALSE)
19 dataku <- data.frame(satria1, satria2, satria3)
20 names(dataku) <- c("ID","Color","Passed") # variable names
21 print(dataku)
22
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
24 print (data_satria)
25
26 install.packages("RMySQL")
27 library(RMySQL)
28 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
29 dbListTables(con)
30 myQuery <- "select * from nama;"
31 df <- dbGetQuery(con, myQuery)
32
```

31:31 (Top Level) R Script

Console Background Jobs

```
> con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
Error in dbConnect(MySQL(), user = "root", password = "", dbname = "houseprices", :
could not find function "dbConnect"
> con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
Error in dbConnect(MySQL(), user = "root", password = "", dbname = "houseprices", :
could not find function "dbConnect"
> library(RMySQL)
Loading required package: DBI
> con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
> dbListTables(con)
[1] "satriapratama"
> myQuery <- "select * from nama;"
> df <- dbGetQuery(con, myQuery)
Error in .local(conn, statement, ...) :
could not run statement: Table 'houseprices.nama' doesn't exist
>
```

e. Latihan Kelima – Teknik Filter Data

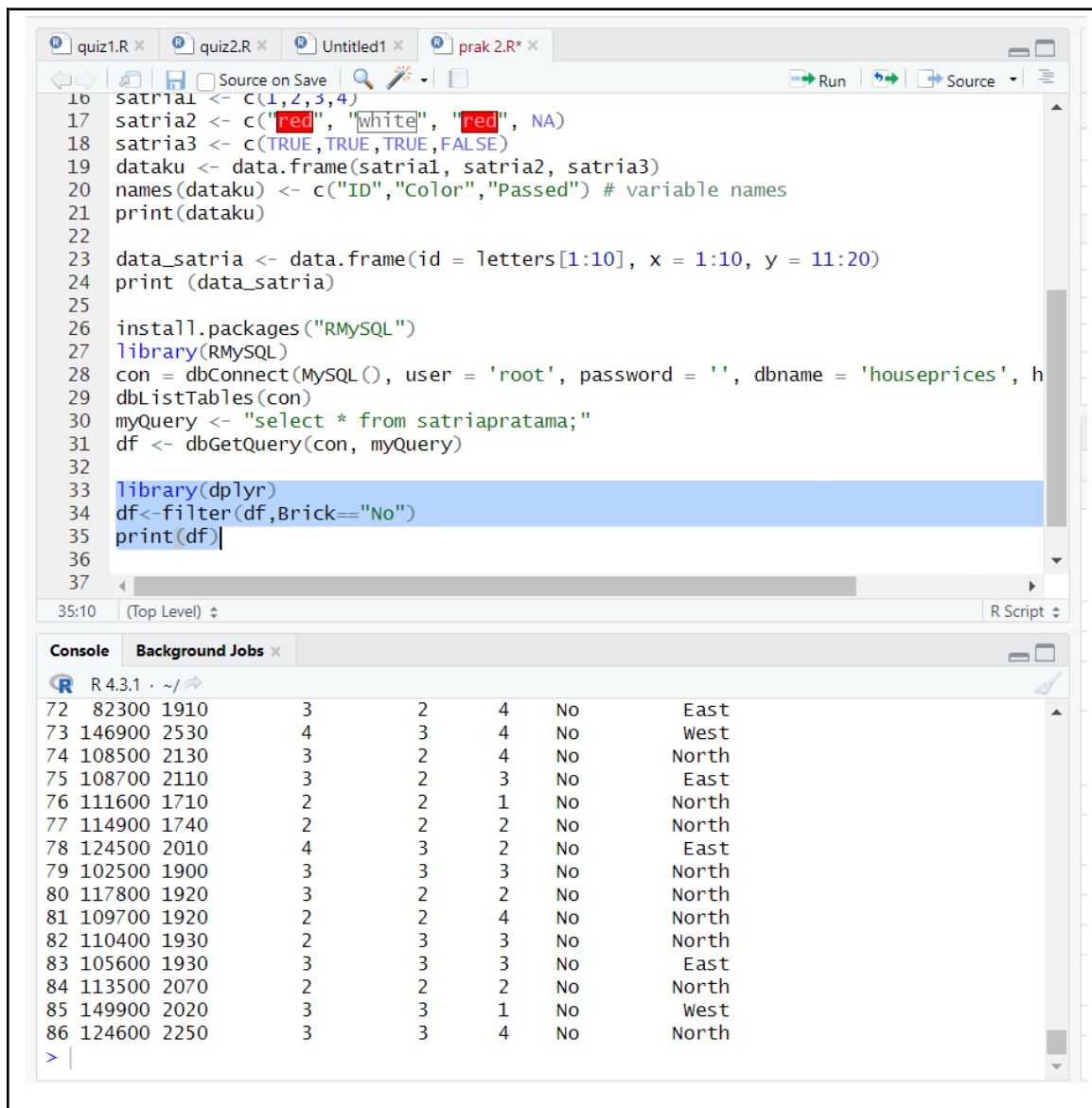
1. Jalankan perintah dibawah ini:

***Perintah ini akan menampilkan 86 baris data hasil filter.**

```
library(dplyr)
df<-filter(df,Brick=="No")
print(df)
```

Output:





```
16 satria1 <- c(1,2,3,4)
17 satria2 <- c("red", "white", "red", NA)
18 satria3 <- c(TRUE,TRUE,TRUE,FALSE)
19 dataku <- data.frame(satria1, satria2, satria3)
20 names(dataku) <- c("ID","Color","Passed") # variable names
21 print(dataku)
22
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
24 print (data_satria)
25
26 install.packages("RMySQL")
27 library(RMySQL)
28 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', h
29 dbListTables(con)
30 myQuery <- "select * from satriapratama;"
31 df <- dbGetQuery(con, myQuery)
32
33 library(dplyr)
34 df<-filter(df,Brick=="No")
35 print(df)
36
37
```

35:10 (Top Level) R Script

Console Background Jobs

R 4.3.1 ~ /

72	82300	1910	3	2	4	No	East
73	146900	2530	4	3	4	No	West
74	108500	2130	3	2	4	No	North
75	108700	2110	3	2	3	No	East
76	111600	1710	2	2	1	No	North
77	114900	1740	2	2	2	No	North
78	124500	2010	4	3	2	No	East
79	102500	1900	3	3	3	No	North
80	117800	1920	3	2	2	No	North
81	109700	1920	2	2	4	No	North
82	110400	1930	2	3	3	No	North
83	105600	1930	3	3	3	No	East
84	113500	2070	2	2	2	No	North
85	149900	2020	3	3	1	No	West
86	124600	2250	3	3	4	No	North

2. Jalankan perintah dibawah ini:

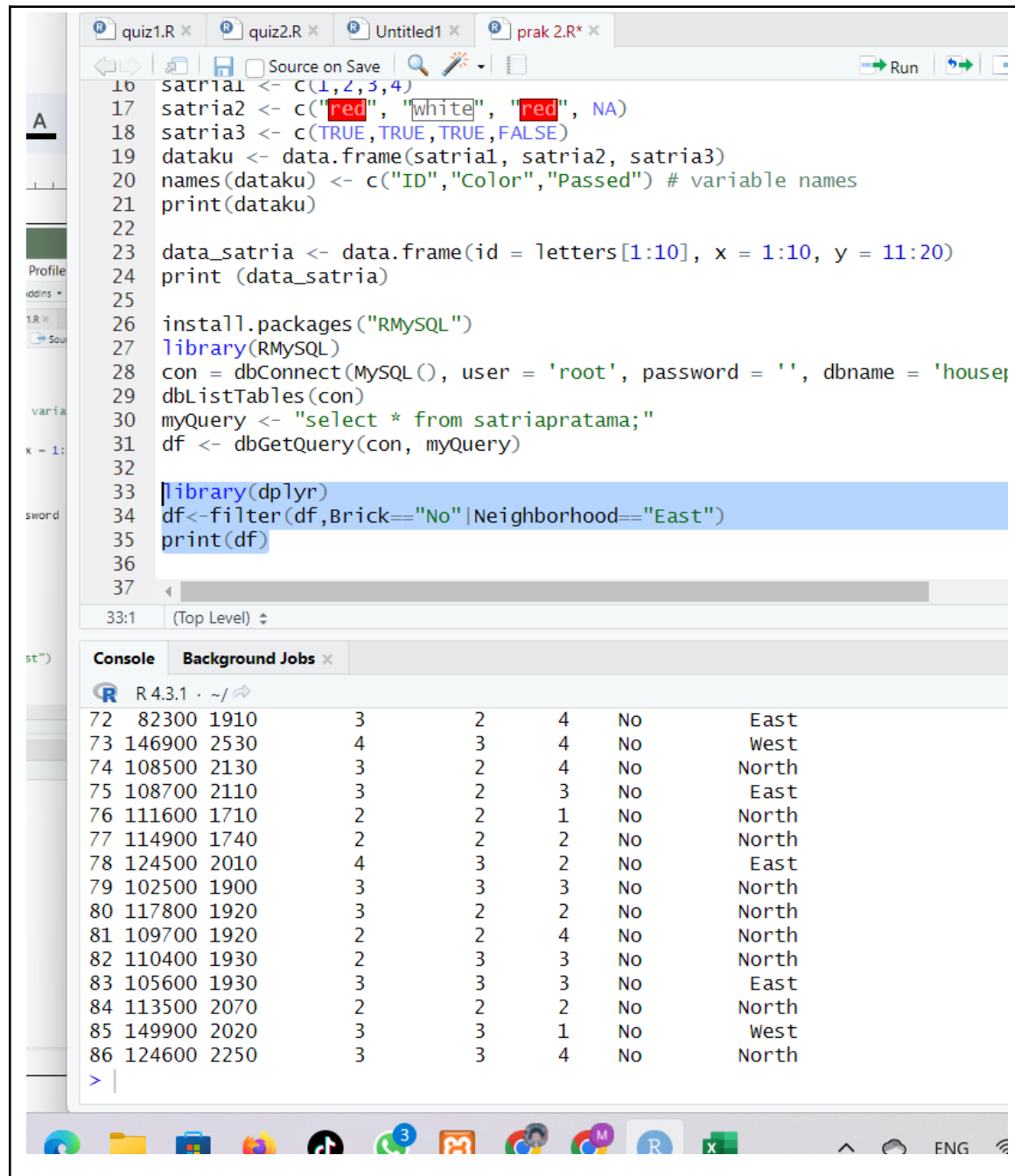
**Perintah ini akan menampilkan 105 baris data hasil filter.*

```
library(dplyr)
```



```
df<-filter(df,Brick=="No"|Neighborhood=="East")  
print(df)
```

Output:



The screenshot shows the RStudio IDE with several open files: quiz1.R, quiz2.R, Untitled1, and prak 2.R*. The active file is prak 2.R*, which contains the following R code:

```
16 satria1 <- c(1,2,3,4)  
17 satria2 <- c("red", "white", "red", NA)  
18 satria3 <- c(TRUE,TRUE,TRUE,FALSE)  
19 dataku <- data.frame(satria1, satria2, satria3)  
20 names(dataku) <- c("ID","Color","Passed") # variable names  
21 print(dataku)  
22  
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)  
24 print (data_satria)  
25  
26 install.packages("RMySQL")  
27 library(RMySQL)  
28 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseq  
29 dbListTables(con)  
30 myQuery <- "select * from satriapratama;"  
31 df <- dbGetQuery(con, myQuery)  
32  
33 library(dplyr)  
34 df<-filter(df,Brick=="No"|Neighborhood=="East")  
35 print(df)  
36  
37
```

The console output shows the result of the filter operation, displaying a data frame with 8 rows and 8 columns. The output is as follows:

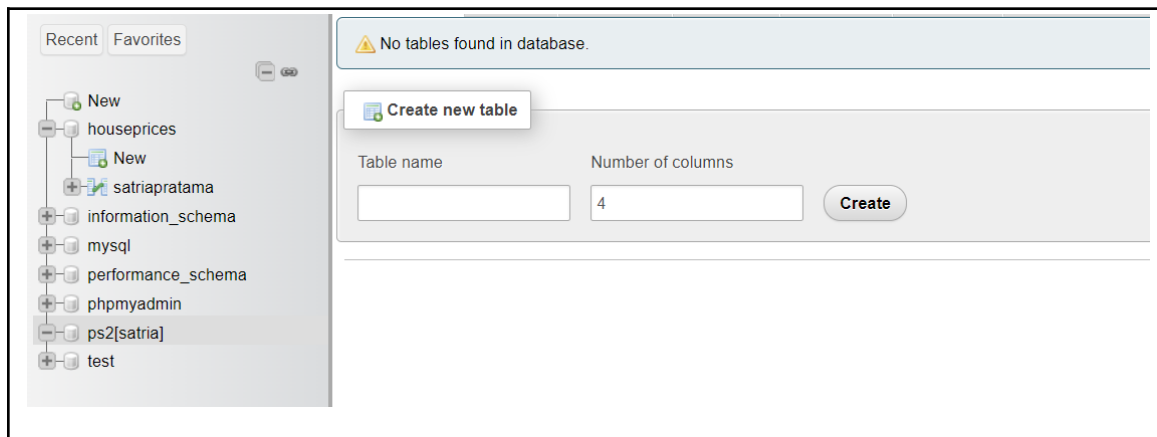
	id	x	y	Brick	Color	Passed	Neighborhood
72	82300	1910	3	2	4	No	East
73	146900	2530	4	3	4	No	West
74	108500	2130	3	2	4	No	North
75	108700	2110	3	2	3	No	East
76	111600	1710	2	2	1	No	North
77	114900	1740	2	2	2	No	North
78	124500	2010	4	3	2	No	East
79	102500	1900	3	3	3	No	North
80	117800	1920	3	2	2	No	North
81	109700	1920	2	2	4	No	North
82	110400	1930	2	3	3	No	North
83	105600	1930	3	3	3	No	East
84	113500	2070	2	2	2	No	North
85	149900	2020	3	3	1	No	West
86	124600	2250	3	3	4	No	North



f. Latihan Keenam – Tugas

Buat sebuah database serta tabel di dalamnya (bisa gunakan data teman dipraktikum ke 1). Lakukan koneksi R ke database serta berikan beberapa filter data sesuai yang anda inginkan. Tampilkan data tersebut

1. Buatlah terlebih dahulu Database baru dengan format PS2[NamaAnda]



2. Import file berformat csv yang telah anda buat sebelumnya di Praktikum 1, dimana file tersebut berisi 6 kolom dan 20 baris data.



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 9 (10 total. Query took 0.0011 seconds.)

`SELECT * FROM `satria_laprak2``

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

Nama	Gender	Angkatan	Tinggi Badan	Waktu Perjalanan	Wilayah Tinggal
Jamil	p	2021	168	31	Tangerang
Kidut	p	2021	165	41	Tangerang
Febri	p	2021	165	51	Tangerang
Eko	p	2021	165	21	Tangerang
Dimas	p	2021	165	13	Tangerang
Putra	p	2021	165	26	Tangerang
Kijang	p	2021	167	38	Jatinegara
Rodrick	p	2021	167	29	Jatinegara
Shemi	p	2021	167	30	Jatinegara
Daut	p	2021	167	48	Jatinegara

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

3. Koneksikan R ke Database tersebut sebagaimana yang telah anda lakukan pada Elemen Kompetensi 1 dimodul kedua ini dengan menyesuaikan kembali nama Database baru yang sudah dibuat.

```
con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
dbListTables(con)
myQuery <- "select * from satria_laprak2;"
df <- dbGetQuery(con, myQuery)

library(dplyr)
df <- filter(df, Gender == "L")
print(df)

36
37 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
38 dbListTables(con)
39 myQuery <- "select * from satria_laprak2;"
40 df <- dbGetQuery(con, myQuery)
41
42 library(dplyr)
43 df <- filter(df, Gender == "P")
44 print(df)
45
```

4. Lakukan filter data terhadap Kolom Gender, untuk melihat berapa baris data Pria/Wanita (Pilih salah 1).



The image displays two screenshots of the RStudio interface, showing R code in the script editor and the corresponding output in the console.

Top Screenshot:

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
quiz1.R x quiz2.R x Untitled1 x prak 2.R x
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
24 print(data_satria)
25
26 install.packages("RMySQL")
27 library(RMySQL)
28 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
29 dbListTables(con)
30 myQuery <- "select * from satriapratama;"
31 df <- dbGetQuery(con, myQuery)
32
33 library(dplyr)
34 df <- filter(df, Brick == "No" | Neighborhood == "East")
35 print(df)
36
37 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
38 dbListTables(con)
39 myQuery <- "select * from satria_laprak2;"
40 df <- dbGetQuery(con, myQuery)
41
42 library(dplyr)
43 df <- filter(df, Gender == "L")
44 print(df)
45
```

Console Output (Top):

```
R 4.3.1 ~ /
> print(df)
[1] Nama Gender Angkatan Tinggi Badan Waktu Perjalanan Wilayah Tinggal
<0 rows> (or 0-length row.names)
> con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
> dbListTables(con)
[1] "satria_laprak2"
> myQuery <- "select * from satria_laprak2;"
> df <- dbGetQuery(con, myQuery)
>
> library(dplyr)
> df <- filter(df, Gender == "L")
> print(df)
[1] Nama Gender Angkatan Tinggi Badan Waktu Perjalanan Wilayah Tinggal
<0 rows> (or 0-length row.names)
>
```

Bottom Screenshot:

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
quiz1.R x quiz2.R x Untitled1 x prak 2.R x
23 data_satria <- data.frame(id = letters[1:10], x = 1:10, y = 11:20)
24 print(data_satria)
25
26 install.packages("RMySQL")
27 library(RMySQL)
28 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'houseprices', host = 'localhost')
29 dbListTables(con)
30 myQuery <- "select * from satriapratama;"
31 df <- dbGetQuery(con, myQuery)
32
33 library(dplyr)
34 df <- filter(df, Brick == "No" | Neighborhood == "East")
35 print(df)
36
37 con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
38 dbListTables(con)
39 myQuery <- "select * from satria_laprak2;"
40 df <- dbGetQuery(con, myQuery)
41
42 library(dplyr)
43 df <- filter(df, Gender == "P")
44 print(df)
45
```

Console Output (Bottom):

```
R 4.3.1 ~ /
> print(df)
[1] Nama Gender Angkatan Tinggi Badan Waktu Perjalanan Wilayah Tinggal
<0 rows> (or 0-length row.names)
> con = dbConnect(MySQL(), user = 'root', password = '', dbname = 'ps2[satria]', host = 'localhost')
> dbListTables(con)
[1] "satria_laprak2"
> myQuery <- "select * from satria_laprak2;"
> df <- dbGetQuery(con, myQuery)
>
> library(dplyr)
> df <- filter(df, Gender == "P")
> print(df)
[1] Nama Gender Angkatan Tinggi Badan Waktu Perjalanan Wilayah Tinggal
<0 rows> (or 0-length row.names)
>
```

5. Lampirkan Screenshot

- Hasil import file kedalam Database
- Berhasilnya koneksi R ke Database
- Kode serta hasil filter di RStudio



4. File Praktikum

Github Repository:

5. Soal Latihan

Soal:

1. Ada berapa tipe data yang data di deklarasikan oleh Rstudio?
2. Bagaimana cara memanggil sebuah tabel di dalam database ke Rstudio?

Jawaban:

1. Tipe data yang dideklarasikan oleh RStudio meliputi numerik (angka), karakter (teks), faktor (kategorikal), logikal (benar/salah), dan kompleks (angka kompleks).
2. Untuk memanggil tabel dari database ke RStudio: a. Pastikan terhubung dengan database yang sesuai menggunakan paket R yang tepat. b. Gunakan fungsi yang sesuai untuk mengambil data dari database, seperti "dbGetQuery()" untuk MySQL. c. Simpan hasil pengambilan data ke dalam objek di RStudio. d. Tabel tersebut dapat digunakan untuk analisis atau manipulasi data di RStudio.

6. Kesimpulan

- a. Dalam pengerjaan praktikum Statistika, ...
- b. Kita juga dapat mengetahui...

7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	...	
2.	Latihan Kedua	...	
3.	Latihan Ketiga	...	



4.	Latihan Keempat	...	
5.	Latihan Kelima	...	
6.	Latihan Keenam	...	

8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	... Menit	...
2.	Latihan Kedua	... Menit	...
3.	Latihan Ketiga	... Menit	...
4.	Latihan Keempat	... Menit	...
5.	Latihan Kelima	... Menit	...
6.	Latihan Keenam	... Menit	...

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang

