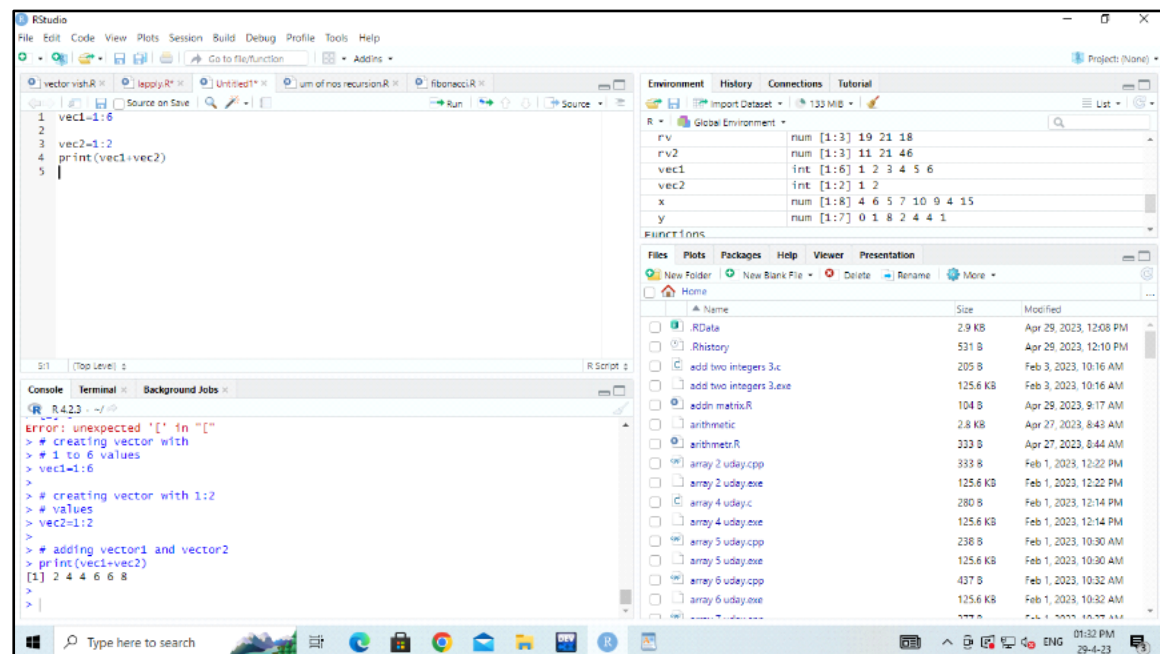


## 1. Demonstrate Vector Recycling in R.

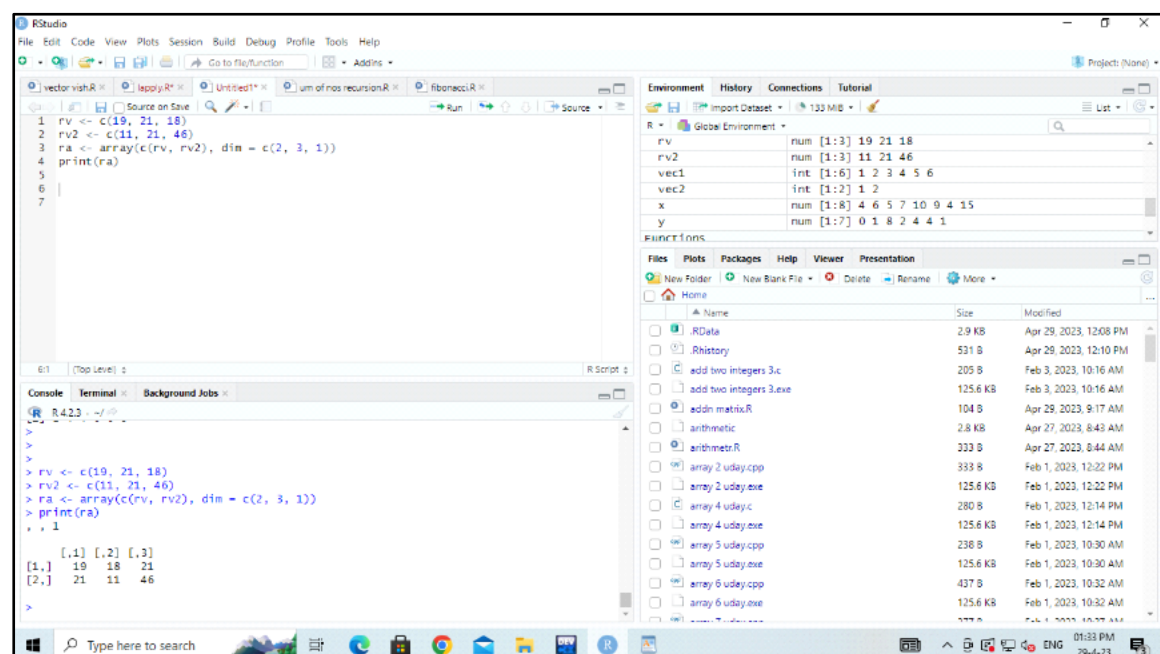


```
1 vec1=1:6
2
3 vec2=1:2
4 print(vec1+vec2)
5
```

```
R 4.2.3 > # creating vector with
> # 1 to 6 values
> vec1=1:6
>
> # creating vector with 1:2
> # values
> vec2=1:2
>
> # adding vector1 and vector2
> print(vec1+vec2)
[1] 2 4 4 6 6 8
```

Variable	Class	Value
rv	num	[1:3] 19 21 18
rv2	num	[1:3] 11 21 46
vec1	int	[1:6] 1 2 3 4 5 6
vec2	int	[1:2] 1 2
x	num	[1:8] 4 6 5 7 10 9 4 15
y	num	[1:7] 0 1 8 2 4 4 1

## 2. Demonstrate the usage of apply function in R.

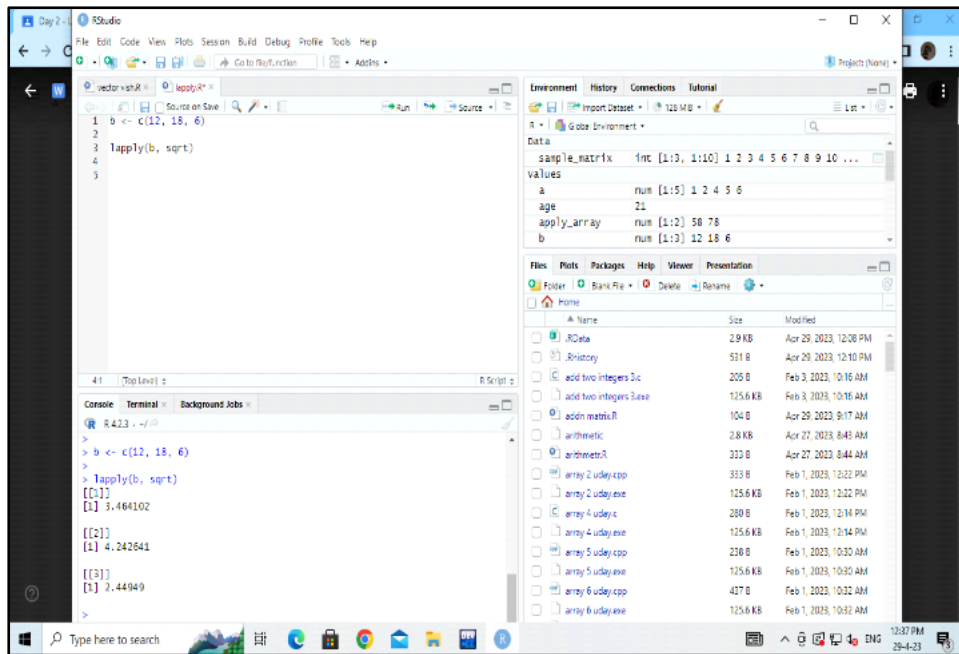


```
1 rv <- c(19, 21, 18)
2 rv2 <- c(11, 21, 46)
3 ra <- array(c(rv, rv2), dim = c(2, 3, 1))
4 print(ra)
5
6
7
```

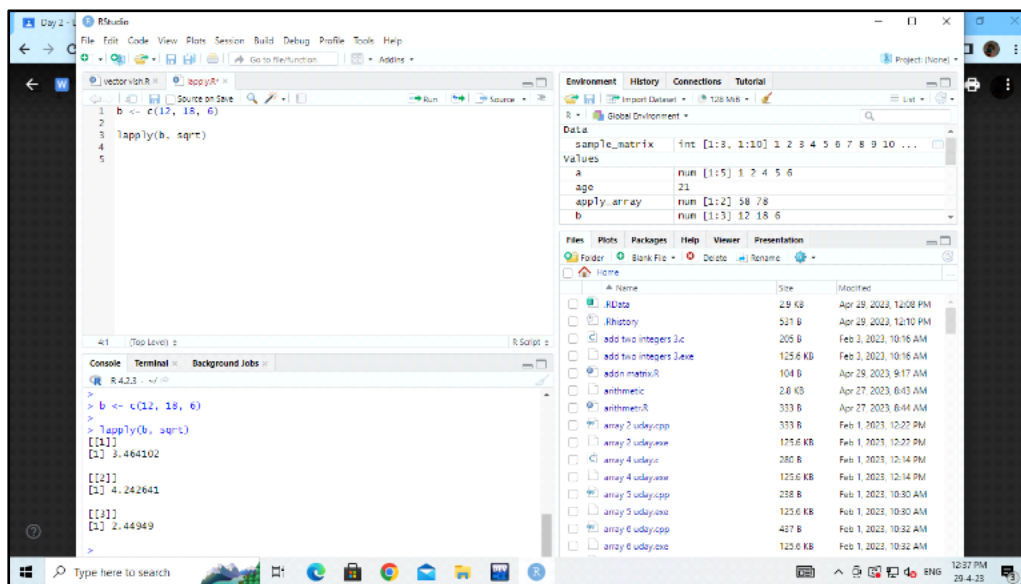
```
R 4.2.3 >
>
> rv <- c(19, 21, 18)
> rv2 <- c(11, 21, 46)
> ra <- array(c(rv, rv2), dim = c(2, 3, 1))
> print(ra)
, , 1
[1,] [1,2] [1,3]
[1,] 19 18 21
[2,] 21 11 46
```

Variable	Class	Value
rv	num	[1:3] 19 21 18
rv2	num	[1:3] 11 21 46
vec1	int	[1:6] 1 2 3 4 5 6
vec2	int	[1:2] 1 2
x	num	[1:8] 4 6 5 7 10 9 4 15
y	num	[1:7] 0 1 8 2 4 4 1

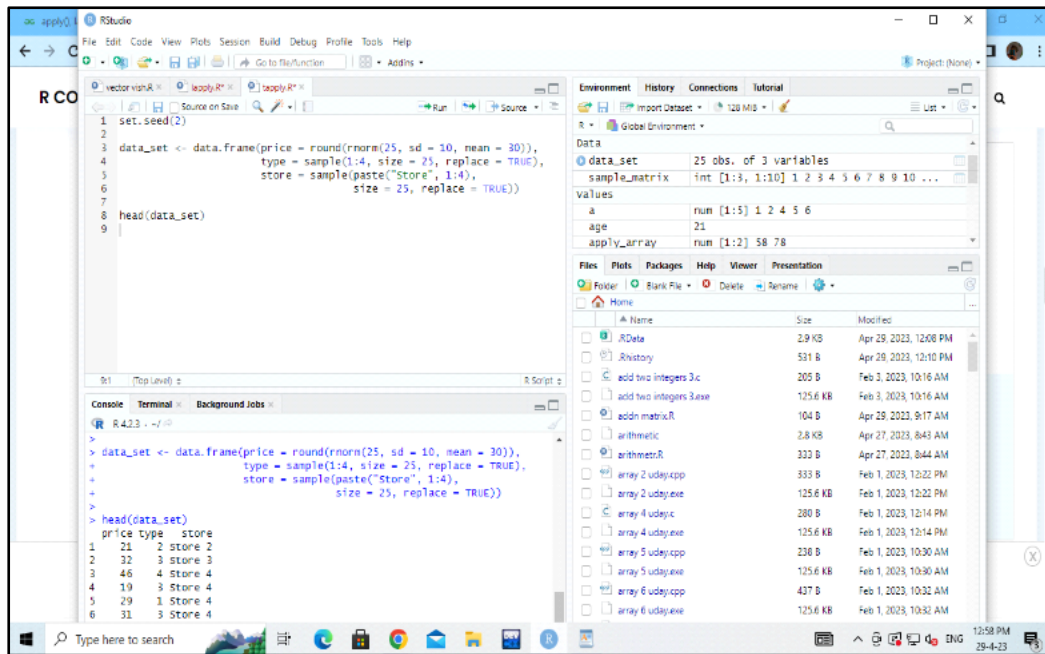
## 3. Demonstrate the usage of lapply function in R.



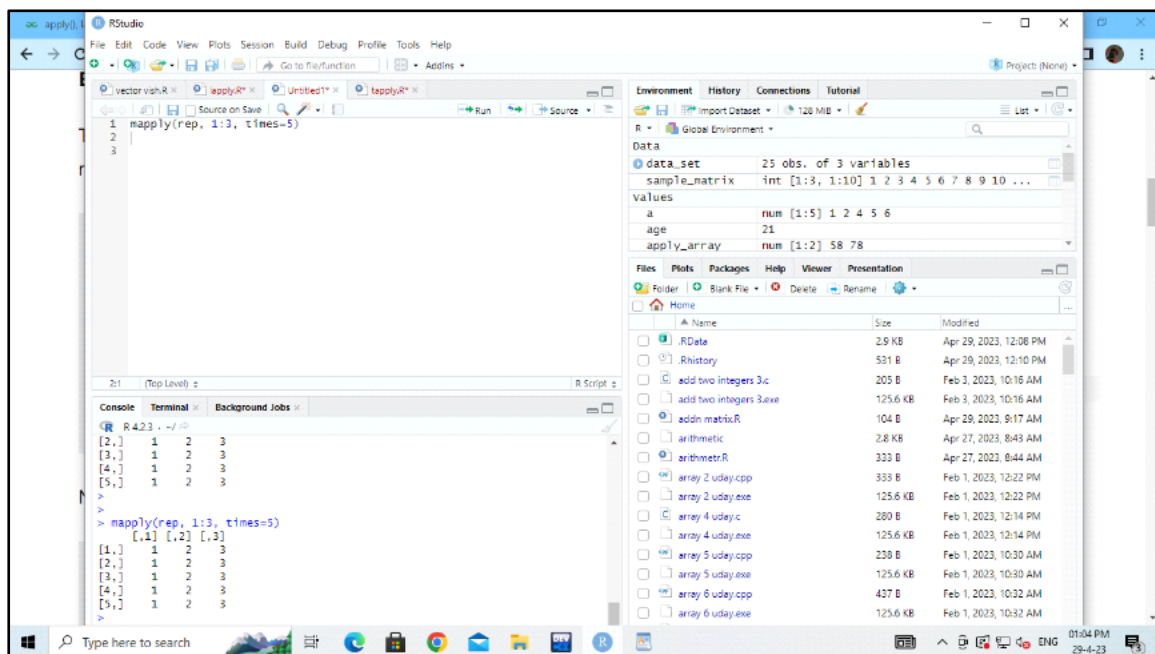
4. Demonstrate the usage of supply function in R.



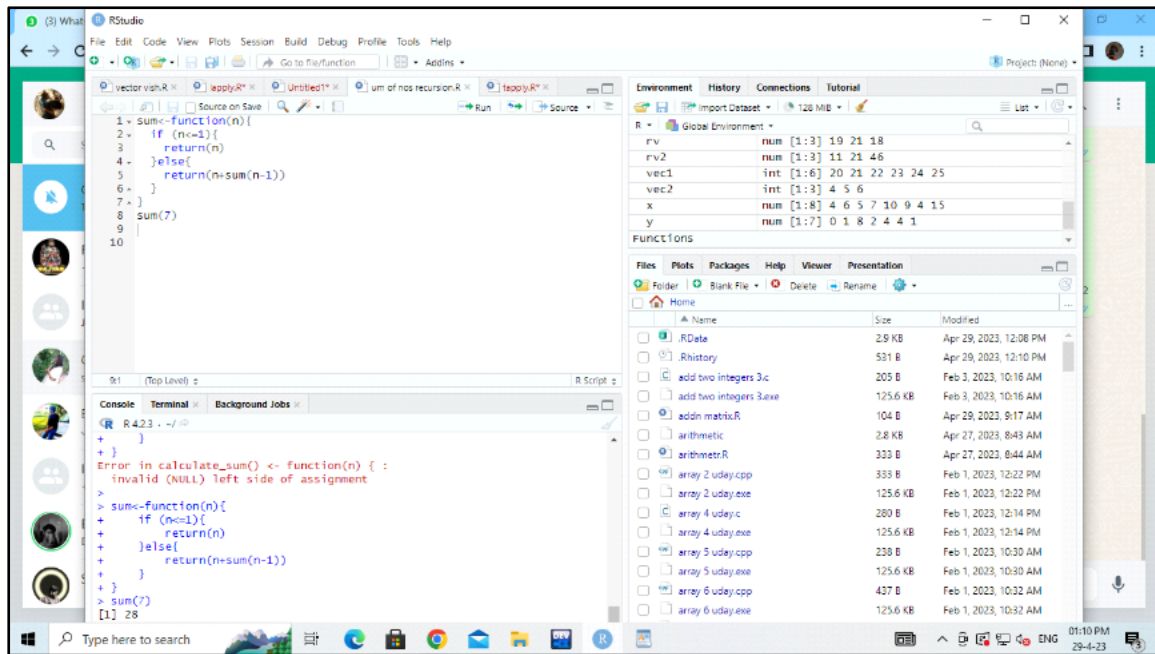
5. Demonstrate the usage of tapply function in R



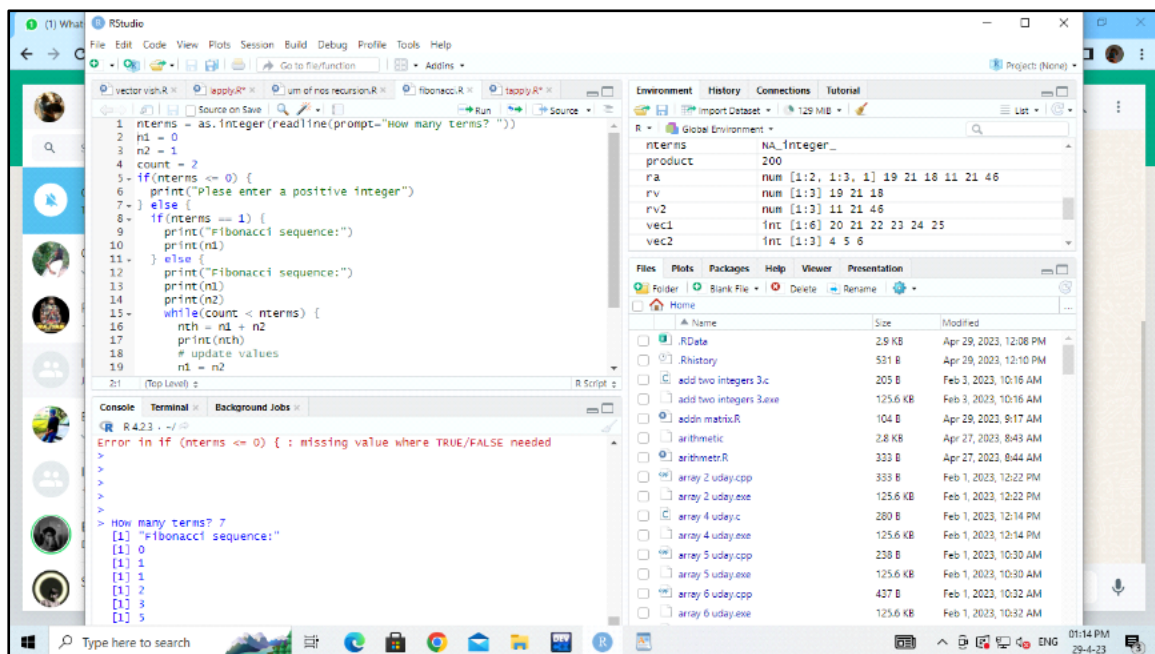
6. Demonstrate the usage of mapply function in R.



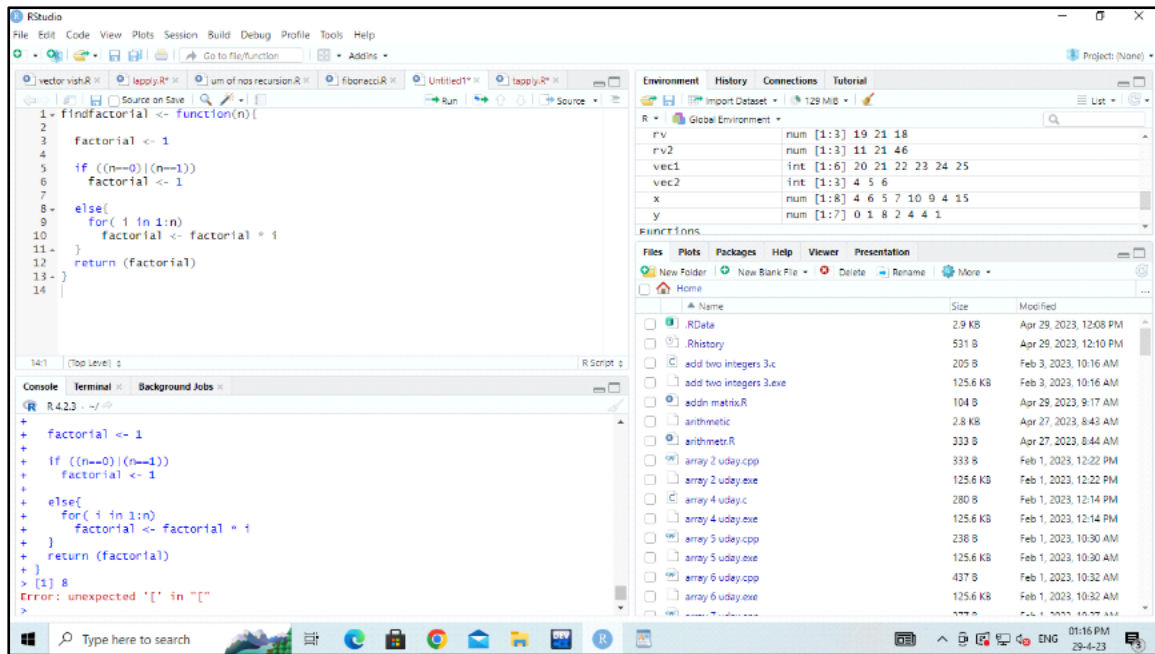
7. Sum of Natural Numbers using Recursion.



8. Write a program to generate Fibonacci sequence using Recursion in R.



9. Write a program to find factorial of a number in R using recursion.



## Exercise 1

Consider two vectors:  $x = \text{seq}(1, 43, \text{along.with} = \text{Id})$

$y = \text{seq}(-20, 0, \text{along.with} = \text{Id})$

Create a data frame `df` as shown below.

`&gt; df`

`Id Letter x y`

1 1 a 1.000000 -20.000000

2 1 b 4.818182 -18.181818

3 1 c 8.636364 -16.363636

4 2 a 12.454545 -14.545455

5 2 b 16.272727 -12.727273

6 2 c 20.090909 -10.909091

7 3 a 23.909091 -9.090909

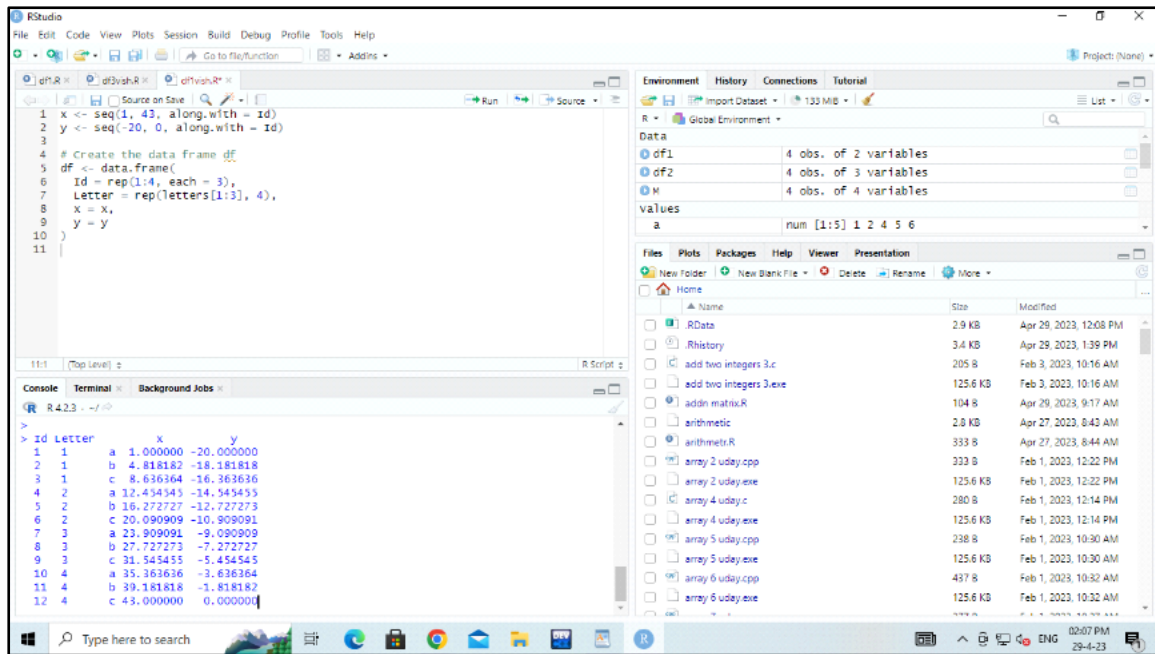
8 3 b 27.727273 -7.272727

9 3 c 31.545455 -5.454545



Edit with WPS Office





## Exercise 2

Using the data frame df in Exercise1, Construct the following data frame. Id

x.ay.ax.by.bx.cy.c 1 1 1.00000 -20.000000 4.818182 -18.181818

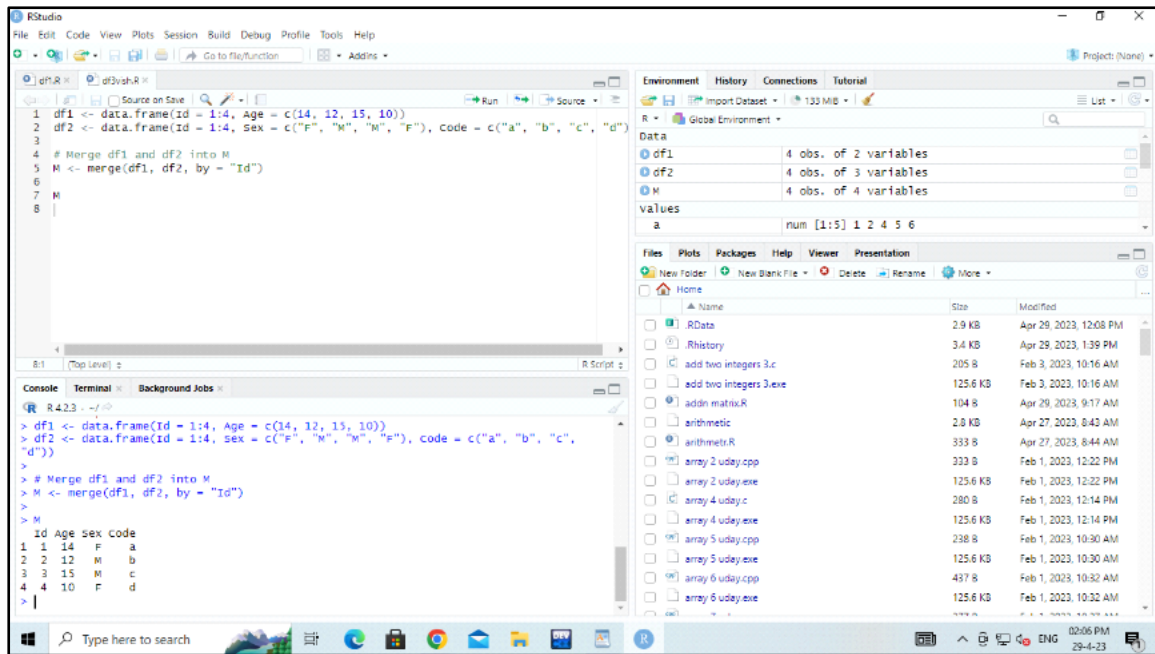
8.636364 -16.363636 4 2 12.45455 -14.545455 16.272727 -12.727273

20.090909 -10.909091 7 3 23.90909 -9.090909 27.727273 -7.272727

31.545455 -5.454545 10 4 35.36364 -3.636364 39.181818 -1.818182

43.000000 0.000000





## Exercise 4

Create a data frame df3:

&gt; df3 id2

score 1 4

100

2 3 98

3 2 94

4 1 99

From M (used in Exercise-3) and df3 create N:

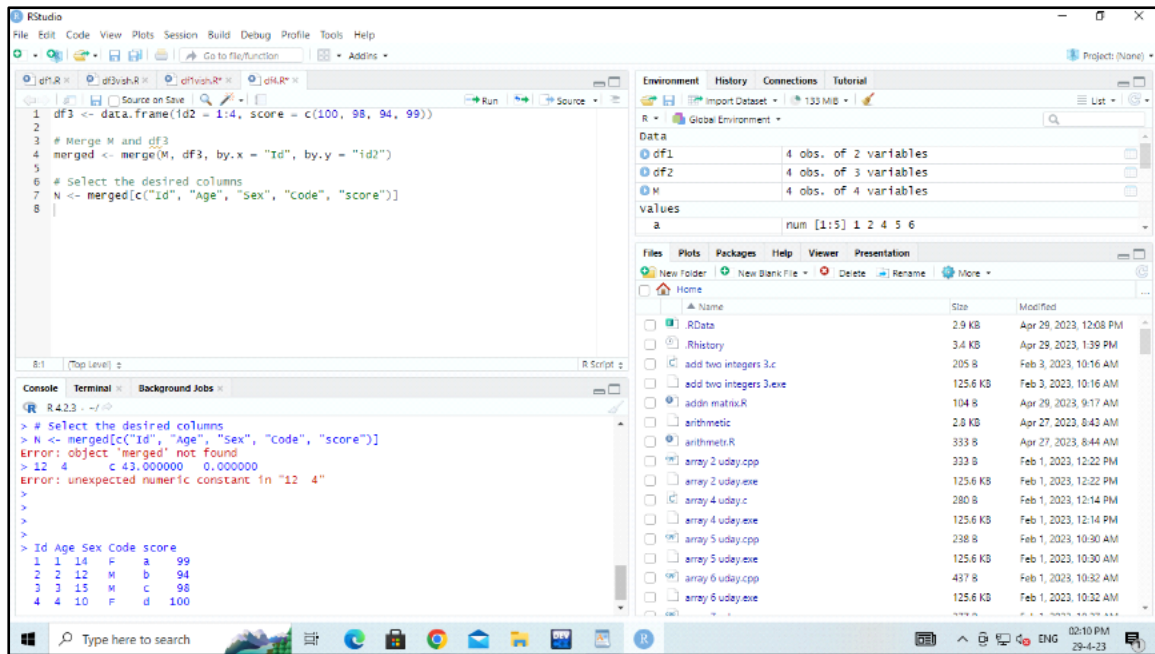
Id Age Sex Code score

1 1 14 F a 99

2 2 12 M b 94

3 3 15 M c 98 4 4 10 F d 100

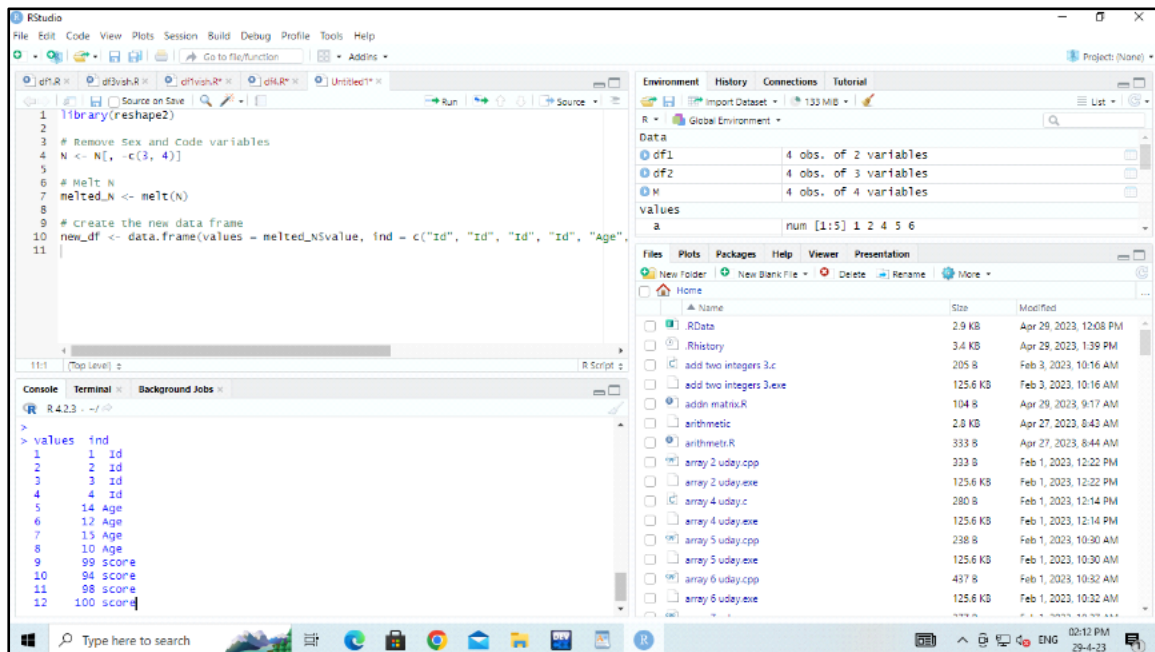




## Exercise 5

Consider the previous one data frame N:

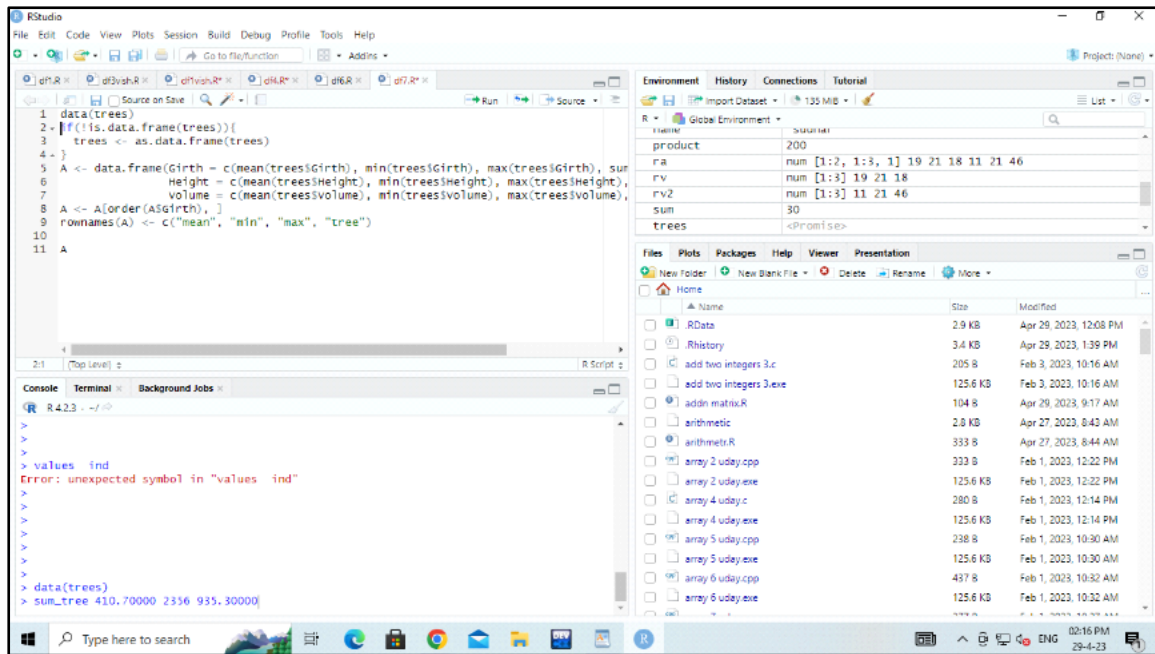
- 1) Remove the variables Sex and Code
- 2) From N, create a data frame:



## Exercise 6

For this exercise, we'll use the (built-in) dataset trees.





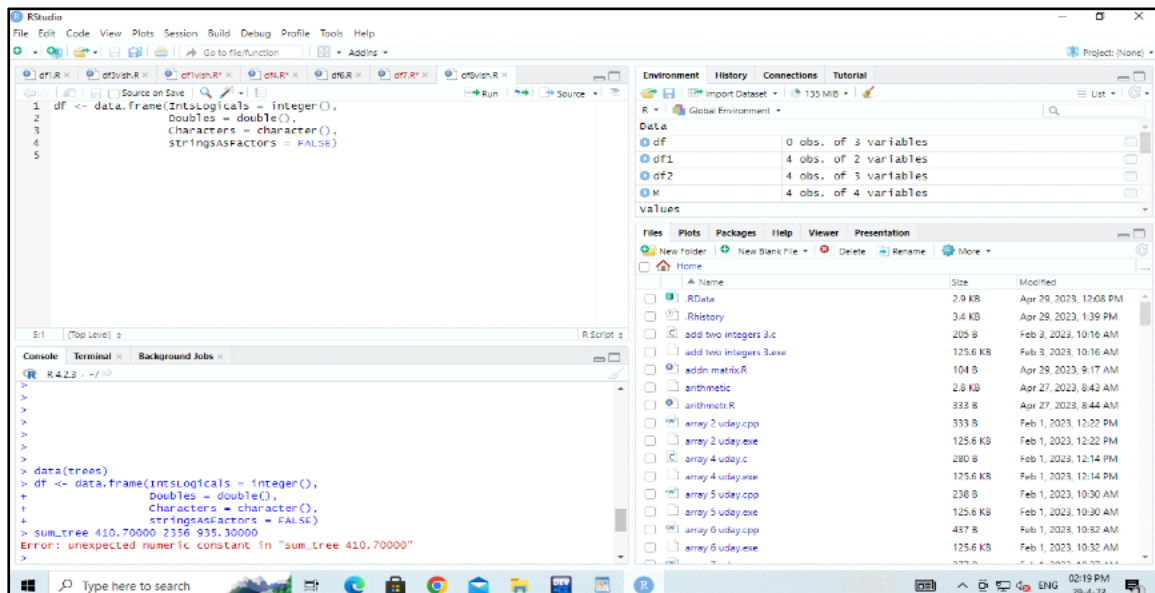
## Exercise 8

Create an empty data frame with column types:

&gt;df

IntsLogicals Doubles Characters

(or 0-length row.names)



## Exercise 9

Create a data frame XY

X=c(1,2,3,1,4,5,2)

Y=c(0,3,2,0,5,9,3)

> XY

XY

1 1 0

2 2 3

3 3 2

4 1 0

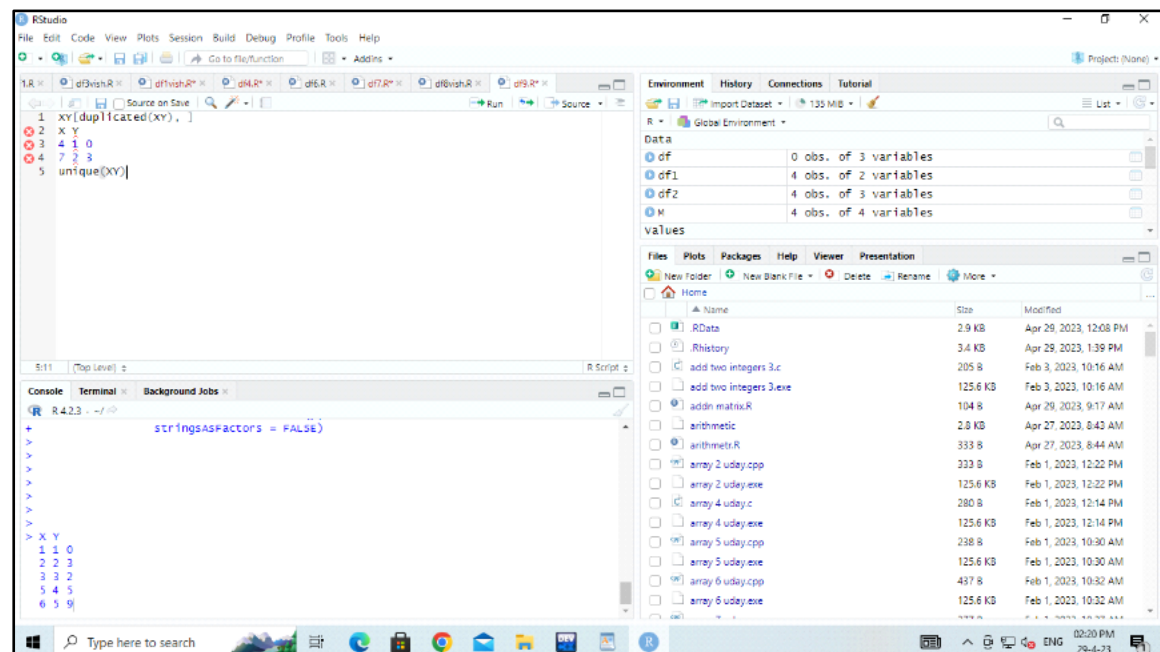
5 4 5

6 5 9

7 2 3

1) look at duplicated elements using a provided R function.

2) keep only the unique lines on XY using a provided R function.



## Exercise 10

Use the (built-in) dataset Titanic.

a) Make sure the object is a data frame, if not change it to a data frame.

b) Define a data frame with value 1st in Class variable, and value NO in Survived variable

and variables Sex, Age and Freq.

Sex Age Freq

1 Male Child 0

5 Female Child 0

9 Male Adult 118

13 Female Adult 4

