



SESSION 5: Data Management Using R

Assignment 2

Table of Contents

1.Introduction	3
2.Objective	3
3.Prerequisites	3
4.Associated Data Files	3
5.Problem Statement	3

Data Analytics

6.Expected Output
3

7.Approximate Time to Complete Task
3

1. Introduction

This assignment will help you understand the concepts learnt in the session.

2. Objective

This assignment will test your skills on Performing SET operations in R.

3. Prerequisites

Not applicable.

4. Associated Data Files

Not applicable.

5. Problem Statement

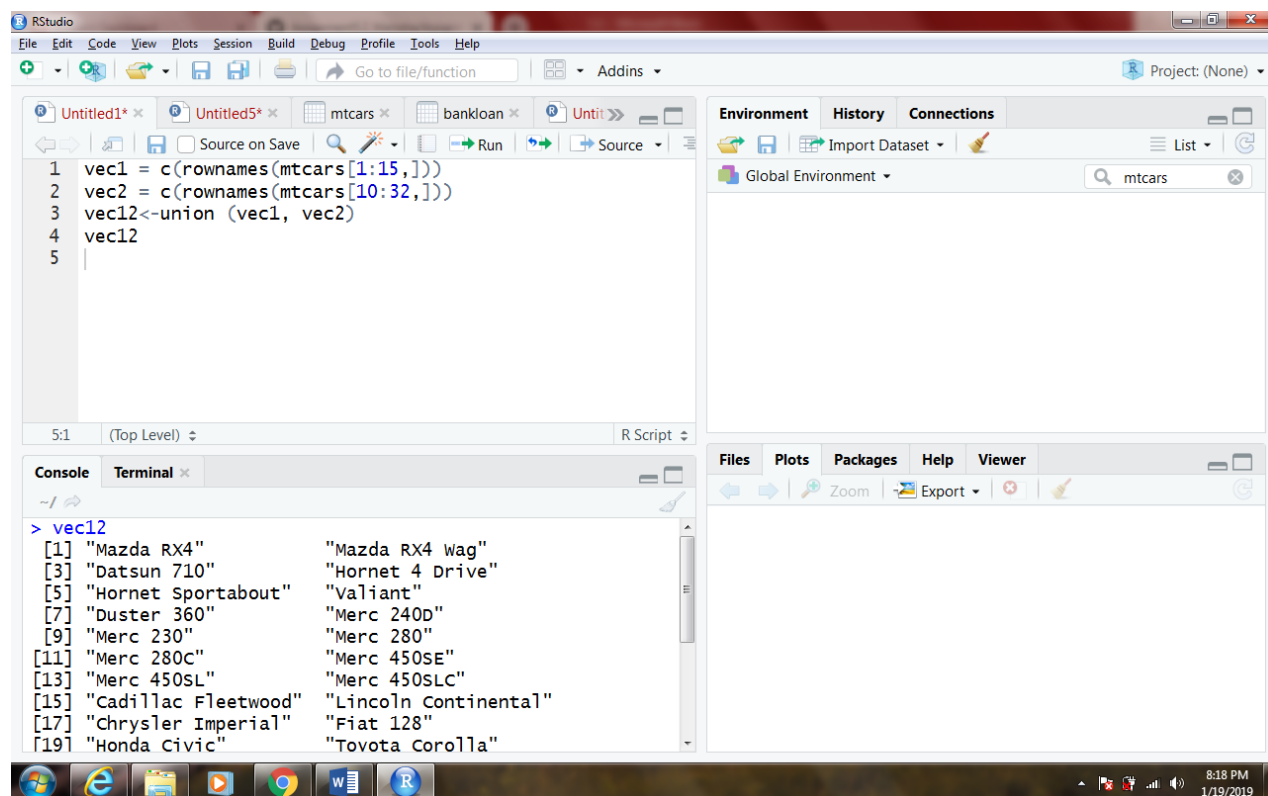
1. obtain the elements of the union between two character vectors.

```
vec1 = c(rownames(mtcars[1:15,]))  
vec2 = c(rownames(mtcars[10:32,]))
```

Ans-

```
vec1 = c(rownames(mtcars[1:15,]))  
vec2 = c(rownames(mtcars[10:32,]))  
vec12<-union (vec1, vec2) # returns all the elements of vec1  
and vec2 without repeating common elements  
vec12
```

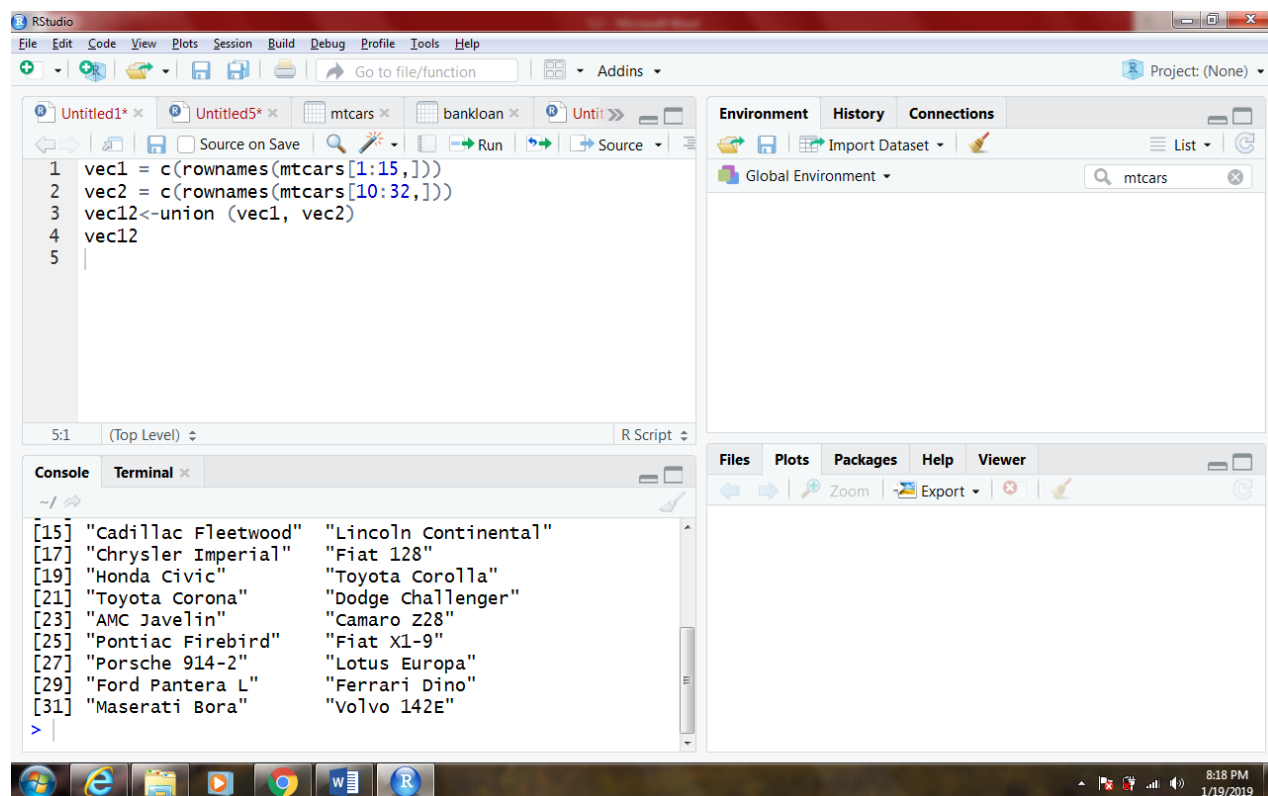
Data Analytics



RStudio interface showing the execution of R code. The script defines two vectors, `vec1` and `vec2`, and combines them into `vec12` using `union`. The console output shows the first 19 rows of the `mtcars` dataset.

```
1 vec1 = c(rownames(mtcars[1:15,]))
2 vec2 = c(rownames(mtcars[10:32,]))
3 vec12<-union (vec1, vec2)
4 vec12
5
```

```
> vec12
[1] "Mazda RX4"           "Mazda RX4 Wag"
[3] "Datsun 710"          "Hornet 4 Drive"
[5] "Hornet Sportabout"   "Valiant"
[7] "Duster 360"          "Merc 240D"
[9] "Merc 230"            "Merc 280"
[11] "Merc 280C"           "Merc 450SE"
[13] "Merc 450SL"          "Merc 450SLC"
[15] "Cadillac Fleetwood"  "Lincoln Continental"
[17] "Chrysler Imperial"  "Fiat 128"
[19] "Honda Civic"         "Toyota Corolla"
```



RStudio interface showing the execution of R code. The script defines two vectors, `vec1` and `vec2`, and combines them into `vec12` using `union`. The console output shows rows 15 through 31 of the `mtcars` dataset.

```
1 vec1 = c(rownames(mtcars[1:15,]))
2 vec2 = c(rownames(mtcars[10:32,]))
3 vec12<-union (vec1, vec2)
4 vec12
5
```

```
[15] "Cadillac Fleetwood" "Lincoln Continental"
[17] "Chrysler Imperial" "Fiat 128"
[19] "Honda Civic"       "Toyota Corolla"
[21] "Toyota Corona"     "Dodge Challenger"
[23] "AMC Javelin"       "Camaro Z28"
[25] "Pontiac Firebird"  "Fiat X1-9"
[27] "Porsche 914-2"     "Lotus Europa"
[29] "Ford Pantera L"    "Ferrari Dino"
[31] "Maserati Bora"     "Volvo 142E"
```

2. Get those elements that are common to both vectors

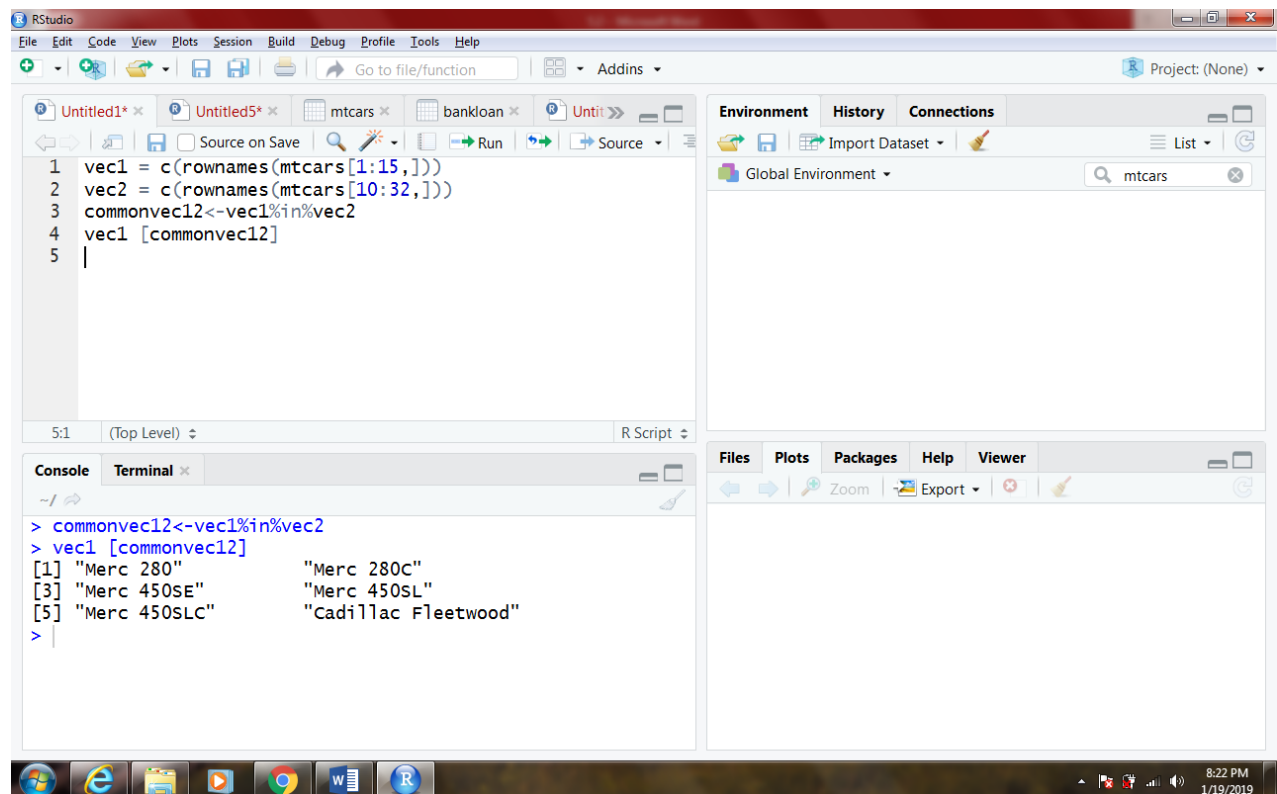
```
vec1 = c(rownames(mtcars[1:15,]))
vec2 = c(rownames(mtcars[10:32,]))
```

ans.

```
vec1 = c(rownames(mtcars[1:15,]))
vec2 = c(rownames(mtcars[10:32,]))
```

```
commonvec12<-vec1%in%vec2
# gives position of common elements
vec1 [commonvec12] # gives elements
```

```
intersect(vec1,vec2)
# alternate way to get intersection of 2
sets of data
```



Data Analytics

The screenshot shows the RStudio environment with the following components:

- Source Editor:** Contains R code to find common elements between two vectors derived from the `mtcars` dataset.
- Environment:** Shows the `Global Environment` with the `mtcars` object loaded.
- Console:** Displays the execution of the code and the resulting output for each step.

```
1 vec1 = c(rownames(mtcars[1:15,]))
2 vec2 = c(rownames(mtcars[10:32,]))
3 commonvec12<-vec1%in%vec2
4 vec1 [commonvec12]
5 intersect(vec1,vec2)
6 |
```

Console Output:

```
> commonvec12<-vec1%in%vec2
> vec1 [commonvec12]
[1] "Merc 280"           "Merc 280C"
[3] "Merc 450SE"         "Merc 450SL"
[5] "Merc 450SLC"        "Cadillac Fleetwood"
> intersect(vec1,vec2)
[1] "Merc 280"           "Merc 280C"
[3] "Merc 450SE"         "Merc 450SL"
[5] "Merc 450SLC"        "Cadillac Fleetwood"
>
```

- Get the difference of the elements between two character vectors.

```
vec1 = c(rownames(mtcars[1:15,]))
vec2 = c(rownames(mtcars[10:32,]))
```

Ans.

```
vec1 = c(rownames(mtcars[1:15,]))
vec2 = c(rownames(mtcars[10:32,]))
vec1[!vec1%in%vec2] # elements of vec1 which are not present
in vec2
vec2[!vec2%in%vec1] # elements of vec2 which are not present
in vec1
```

```
union(vec1[!vec1%in%vec2],vec2[!vec2%in%vec1]) #elements
which are not common in vec1 and vec2
```

The screenshot shows the RStudio interface with the following code in the script editor:

```
1 vec1 = c(rownames(mtcars[1:15,]))
2 vec2 = c(rownames(mtcars[10:32,]))
3 vec1[!vec1%in%vec2]
4 vec1
5 vec2[!vec2%in%vec1]
6 vec2
7 union(vec1[!vec1%in%vec2],vec2[!vec2%in%vec1])
8 |
```

The console output shows the result of the union operation:

```
> union(vec1[!vec1%in%vec2],vec2[!vec2%in%vec1])
[1] "Mazda RX4" "Mazda RX4 Wag"
[3] "Datsun 710" "Hornet 4 Drive"
[5] "Hornet Sportabout" "Valiant"
[7] "Duster 360" "Merc 240D"
[9] "Merc 230" "Lincoln Continental"
[11] "Chrysler Imperial" "Fiat 128"
[13] "Honda Civic" "Toyota Corolla"
[15] "Toyota Corona" "Dodge Challenger"
[17] "AMC Javelin" "Camaro Z28"
```

4. Test the equality of two character vectors

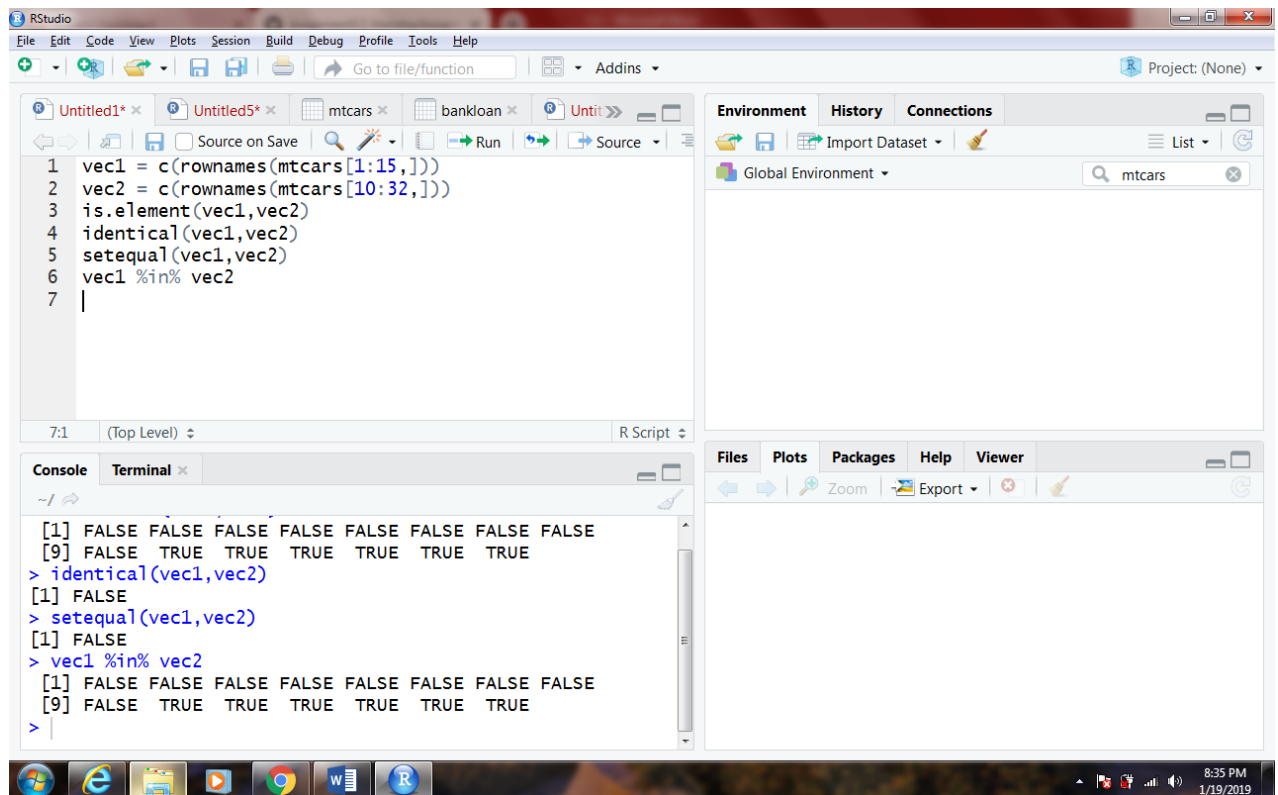
```
vec1 = c(rownames(mtcars[1:15,]))  
vec2 = c(rownames(mtcars[11:25,]))
```

```
is.element(vec1,vec2)
```

```
identical(vec1,vec2)
```

```
setequal(vec1,vec2)
```

```
vec1 %in% vec2
```



6. Expected Format

1. R file should be submitted where applicable.
2. R file should be in PDF or in .r format
3. Proper screenshots of the outputs should be submitted as well
4. The r codes, if submitted in any other format, will be subjected to deduction in marks

Data Analytics

Note: Your solution will not be entertained if it is any other format, e.g., .zip, .doc, .rtf etc.

7. Approximate Time to Complete Task

30 mins.