Session 5 – Data Management using R

Assignment - 3

Problem Statement

1. Test whether two vectors are exactly equal (element by element).

vec1 = c(rownames(mtcars[1:15,]))

vec2 = c(rownames(mtcars[11:25,]))

Ans.

```
#assignment5.3
vec1 = c(rownames(mtcars[1:15,]))
vec2 = c(rownames(mtcars[11:25,]))
all.equal.character(vec1,vec2)
```

```
> library(readxl)
Warning message:
package 'readxl' was built under R version 3.4.4
> Cereals_data <- read_xlsx("cereals_practice.xlsx",sheet = 1)
Error: `path` does not exist: 'cereals_practice.xlsx'
> #assignment5.3
> vec1 = c(rownames(mtcars[1:15,]))
> vec2 = c(rownames(mtcars[11:25,]))
> all.equal.character(vec1,vec2)
[1] "15 string mismatches"
>
```

2. Sort the character vector in ascending order and descending order.

vec1 = c(rownames(mtcars[1:15,]))

vec2 = c(rownames(mtcars[11:25,]))

Ans.

```
vec1_asc <- sort(vec1,decreasing = FALSE)
vec1_asc
vec2_asc <- sort(vec2,decreasing = FALSE)
vec2_asc
vec2_desc <- sort(vec2,decreasing = TRUE)
vec2_desc
vec1_desc <- sort(vec1,decreasing = TRUE)
vec1_desc
```

```
> vec1_asc <- sort(vec1,decreasing = FALSE)
> vec1_asc
 [1] "Cadillac Fleetwood" "Datsun 710"         "Duster 360"
 [4] "Hornet 4 Drive"     "Hornet Sportabout"  "Mazda RX4"
 [7] "Mazda RX4 Wag"      "Merc 230"           "Merc 240D"
[10] "Merc 280"           "Merc 280C"          "Merc 450SE"
[13] "Merc 450SL"         "Merc 450SLC"        "Valiant"
> vec2_asc <- sort(vec2,decreasing = FALSE)
> vec2_asc
 [1] "AMC Javelin"          "Cadillac Fleetwood"  "Camaro Z28"
 [4] "Chrysler Imperial"    "Dodge Challenger"    "Fiat 128"
 [7] "Honda Civic"          "Lincoln Continental" "Merc 280C"
[10] "Merc 450SE"           "Merc 450SL"          "Merc 450SLC"
[13] "Pontiac Firebird"     "Toyota Corolla"      "Toyota Corona"
> vec2_desc <- sort(vec2,decreasing = TRUE)
> vec2_desc
 [1] "Toyota Corona"        "Toyota Corolla"      "Pontiac Firebird"
 [4] "Merc 450SLC"          "Merc 450SL"          "Merc 450SE"
 [7] "Merc 280C"            "Lincoln Continental" "Honda Civic"
[10] "Fiat 128"             "Dodge Challenger"    "Chrysler Imperial"
[13] "Camaro Z28"           "Cadillac Fleetwood"  "AMC Javelin"
> vec1_desc <- sort(vec1,decreasing = TRUE)
> vec1_desc
 [1] "Valiant"            "Merc 450SLC"        "Merc 450SL"
 [4] "Merc 450SE"         "Merc 280C"          "Merc 280"
 [7] "Merc 240D"          "Merc 230"           "Mazda RX4 Wag"
[10] "Mazda RX4"          "Hornet Sportabout"  "Hornet 4 Drive"
[13] "Duster 360"         "Datsun 710"         "Cadillac Fleetwood"
>
```

3. What is the major difference between str() and paste() show an example.

Ans.

Paste()

paste converts its arguments (*via* as.character) to character strings, and concatenates them (separating them by the string given by sep). If the arguments are vectors, they are concatenated term-by-term to give a character vector result. Vector arguments are recycled as needed, with zero-length arguments being recycled to "".

Usage

```
paste (…, sep = " ", collapse = NULL)
paste0(…, collapse = NULL)
```

```
paste("Hello","World",sep = " ",collapse = "-")
```

```
> paste("Hello","World",sep = " ",collapse = "-")
[1] "Hello World"
>
```

Str()

Compactly Display The Structure Of An Arbitrary R Object

Compactly display the internal structure of an R object, a diagnostic function and an alternative to summary (and to some extent, dput). Ideally, only one line for each 'basic' structure is displayed. It is especially well suited to compactly display the (abbreviated) contents of (possibly nested) lists. The idea is to give reasonable output for any R object. It calls args for (non-primitive) function objects.

```
str(vec1)
str(vec2_asc)
```

16:1 (Top Level) R Script

Console Terminal

```
[13] "Duster 360"          "Datsun 710"          "Cadillac Fleetwood"
> paste("Hello","World",sep = " ",collapse = -)
Error: unexpected ')' in "paste("Hello","World",sep = " ",collapse = -)"
> paste("Hello","World",sep = " ",collapse = -)
Error: unexpected ')' in "paste("Hello","World",sep = " ",collapse = -)"
> paste("Hello","World",sep = " ",collapse = "-")
[1] "Hello World"
> str(vec1)
 chr [1:15] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Drive" ...
> str(vec2_asc)
 chr [1:15] "AMC Javelin" "Cadillac Fleetwood" "Camaro Z28" ...
>
```

4. Introduce a separator when concatenating the strings.

Ans.

```
paste("Hi","This","Is","R","Language",sep = "--")
```

17:1 (Top Level) R Script

Console Terminal

```
Error: unexpected ')' in "paste("Hello","World",sep = " ",collapse = -)"
> paste("Hello","World",sep = " ",collapse = -)
Error: unexpected ')' in "paste("Hello","World",sep = " ",collapse = -)"
> paste("Hello","World",sep = " ",collapse = "-")
[1] "Hello World"
> str(vec1)
 chr [1:15] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Drive" ...
> str(vec2_asc)
 chr [1:15] "AMC Javelin" "Cadillac Fleetwood" "Camaro Z28" ...
> paste("Hi","This","Is","R","Language",sep = "--")
[1] "Hi--This--Is--R--Language"
>
```