Ass

Aruna

10 November 2018

# 1. Test whether two vectors are exactly equal(element by element)   
vec1= c(rownames(mtcars[1:15,]))  
vec2= c(rownames(mtcars[11:25,]))  
identical(vec1,vec2)

## [1] FALSE

# 2. Sort the Character vector in asecending order and descending order  
vec1= c(rownames(mtcars[1:15,]))  
vec2= c(rownames(mtcars[11:25,]))  
sort(vec1, na.last = NA, decreasing = FALSE)

## [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"

# 3. What is the Major difference between str c() and paste(). show an example  
 #The difference is for str\_c() the default is no separator,   
 #so it acts just like paste0() as a default.   
 #Paste() and paste0() are both functions from the base package,   
 #whereas str\_c() comes from the stringr package.  
library(stringr)  
str\_c("I","am","trying","to","learn R") # there is no seperator just like paste0

## [1] "Iamtryingtolearn R"

str\_c("Add"," a"," space"," before"," each letter")

## [1] "Add a space before each letter"

#create a vector words  
 words = c("I","am","trying","to","learn","R")  
#paste with seperator with \_  
paste(words,collapse = "\_")

## [1] "I\_am\_trying\_to\_learn\_R"

#paste with seperator with " "  
paste(words, collapse = " ")

## [1] "I am trying to learn R"

# 4. Introduce a seperator when concatenating the strings  
try= c("Concatenating","string","example")  
#paste with seperator with \_  
paste(try,collapse = "\_")

## [1] "Concatenating\_string\_example"

#paste with seperator with " "  
paste(try, collapse = " ")

## [1] "Concatenating string example"

## R Markdown

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.