```r
# Data structure: 4. data.frame - continued

rm(list=ls())

# Merging two data frames horizontally: merge()

name <- c("Tom", "James", "Mary", "Paker")
score <- c(9, 7, 6, 10)
grade <- factor(c("A", "C", "D", "A"), ordered=TRUE, levels=c("D","C","B","A"))
Econ_dep <- data.frame(name, score, grade, stringsAsFactors = FALSE)

name <- c("James", "Tom", "Mary")  # Note that Paker is missing here.

attendance <- c("all", "some", "never")
attitude <- c("good","good","bad")
Econ_dep_1 <- data.frame(name, attendance, attitude)

rm(name, score, grade)
rm(attendance, attitude)

Econ_dep_f <- merge(Econ_dep,Econ_dep_1, by="name", all = TRUE, sort=FALSE)
Econ_dep_f                                      # Note that the orders of columns can be different

# Adding more units to a data.frame

name <- c("Jane", "Kim", "John")
score <- c(8, 9, 5)
grade <- factor(c("A", "A", "D"), ordered=TRUE, levels=c("D","C","B","A"))
Econ_dep2 <- data.frame(score, name, grade)   # Note that the orders of columns can be different

Econ_dep_all <- rbind(Econ_dep, Econ_dep2)
Econ_dep_all

################################################################################################
# Data structure: 5. list

# List is a collection of various types of data structures under one name. Data structures can be matrices,
# vectors, factor, data frames, and other lists, etc. Lists are used to collect different types of items together.

age1 <- c(3,5,6)
age2 <- c(4,3)
emp <- list(employee = c("Anna","Tom"), children = c(3,2), children_age = list(age1,age2))
length(emp)
str(emp)
class(emp)
# "employee, children, children_age are the name of components

gdp <- c(1549.13, 1371.15, 1613.46, 1788.65, 1824.29, 1792.88, 1552.81, 1529.76)
year <- 2008:2015
names(gdp) <- year

provinces <- c("AB", "BC", "MB", "NB", "NS", "ON", "PE", "QC", "SK")

cities <- data.frame(name = c("Toronto", "Montreal", "Vancouver", "Calgary"), pop = c(5930, 4100, 2463, 1393))

Canada <- list(gdp, provinces, cities)
Canada
str(Canada)

names(Canada) <- c("GDP", "Prov", "City")  # specify the name of each object in the list.
rm(gdp, year, provinces, cities)
Canada
```

```
str(Canada)
Canada$GDP

# Select a subset of a list

# 1. list slicing: We get a list slice with the single square bracket [].
Canada[3]       # list with one object
class(Canada[3])
Canada[-3]      # list that contains two objects
Canada[c(1,3)]
Canada[c(T,F,T)]
Canada[names(Canada)=="City"]
Canada["City"]

is.list(Canada["City"]) # How can you select a component from the GDP vector?
Canada[2][1]       # Does not present the first component of "Prov".

# 2. Select components from the inside of a list. Not slicing.
Canada[[3]]
class(Canada[[3]])
Canada[[3]][1,]              # vs. Canada[3] is still a list with one component
                             # In contrast, Canada[[3]] is a data.frame.
Canada[[2]][3]
Canada[2][3]
Canada[["GDP"]][3]

Canada$GDP[names(Canada$GDP)==2010]   # Instead of [[]], we can also use $

# Adding additional component to a list

Lan = c("English", "French")
Canada$Language <- Lan
str(Canada)

# Exercise 1.
################################################################################

mtcars
nrow(mtcars)

# Construct two subsets from mtcars. The first one, "mtcar1", contains rows above
"Cadillac Fleetwood."
# The second one, "mtcars2", contains rows from "Cadillac Fleetwood."
# You can try which() and row.names() functions to find the row index number of
"Cadillac Fleetwood."

a <-
mtcars1 <-
mtcars2 <-

# remove the columns "vs" and "am" in mtcars2
mtcars2[,c("vs","am")] <- NULL

# Attach mtcars2 to mtcars1. Since we have removed two columns in mtcars, you should
first make
# two empty columns in mtcars2 and name them as "vs" and "am". Hint: rbind()


# Exercise 2.
################################################################################

# Construct two subsets from mtcars. The first one, "mtcars3", includes the columns
before "wt"
# and the second one, "mtcars4", include the columns from "wt".
a <- which(names(mtcars) == "wt")
mtcars3 <- mtcars[,1:a-1]
mtcars4 <- mtcars[,a:ncol(mtcars)]
name <- row.names(mtcars)

```

```
mtcars3 <- cbind(name, mtcars3)
mtcars4 <- cbind(name, mtcars4)

row.names(mtcars3) <- NULL
row.names(mtcars4) <- NULL

mtcars3
mtcars4

# Regarding mtcars4, redefine "mtcars4" after eliminating the rows for Mercedes series.
# You can use substr() function.

a <- substr(mtcars4$name, 1, 4)
mtcars4 <- mtcars4[!(a=="Merc"),]
mtcars4

# Now, merge mtcars3 and mtcars4. Keep the data for Mercedes series.
mtcars_all <-
mtcars_all

# Exercise 3.
################################################################################
# First create the components to create a list for the movie "The Shining"
mov <- "The Shining"
act <- c("Jack Nicholson", "Shelley Duvall", "Danny Lloyd", "Scatman Crothers", "Barry Nelso")

scores <- c(4.5, 4.0, 5.0)
sources <- c("IMDb1", "IMDb2", "IMDb3")
comments <- c("Best Horror Film I have ever seen", "A truly brilliant and scary ESAim from Stanley Kubrick"
              , "A masterpiece of psychological horror")
rev <- data.frame(scores, sources, comments)

# Create a list "shining_list" with mov, act and rev
shine_list <- list(mov, act, rev)

rm(mov, act, scores, comments, rev)
# Name the components in shine_list, mov, act and rev with "moviename", "actors" and "reviews" repectively.
names(shine_list) <- c("moviename", "actors","reviews")
shine_list

# Print out the vector that contains the actor names that start with "S"


# Print the second element of the vector that contains the actors


# Print the comments.


# Select the comment that is corresponding with score 5.


# Create a list "shining_list_new" by adding the year of release (1980) to shining_list

```