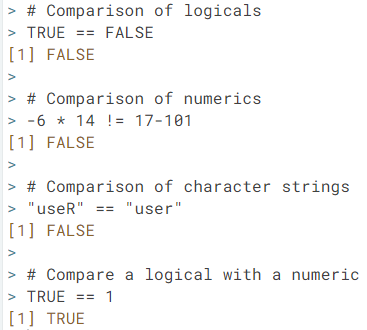
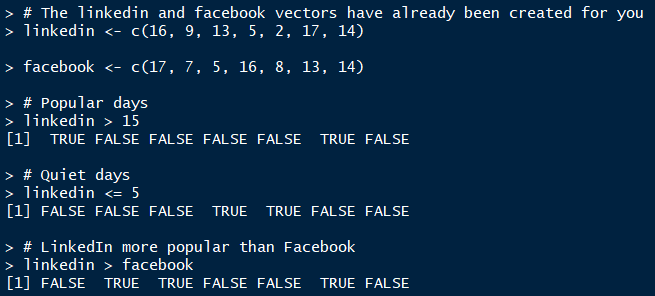
Intermediate R

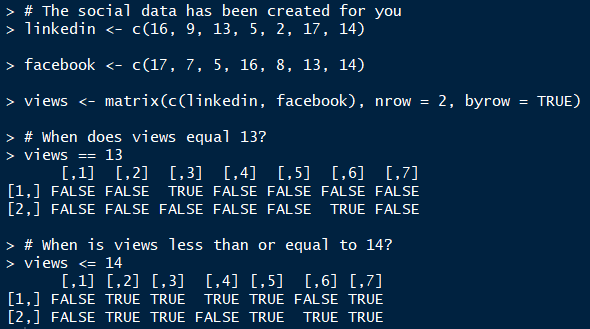
**Comparison Operators**

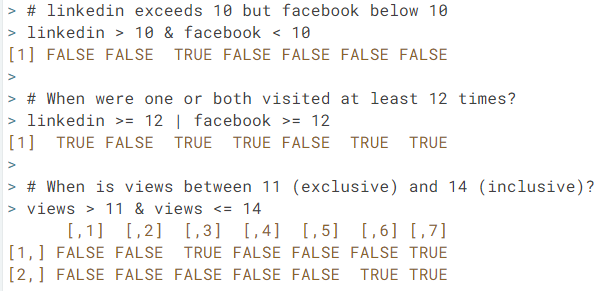
****

**Note**: in R, TRUE coerces to 1 and FALSE to 0. Also when comparing character strings, R uses alphabetical order to test greater/less than comparisons.

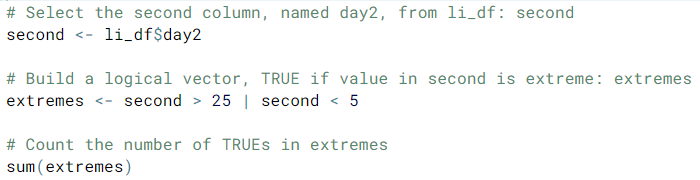


**vectors**



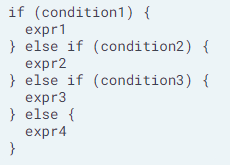


**matrices**

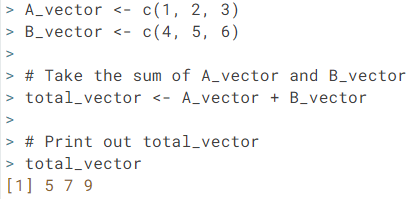


**a good way to count how many elements meet a given condition**

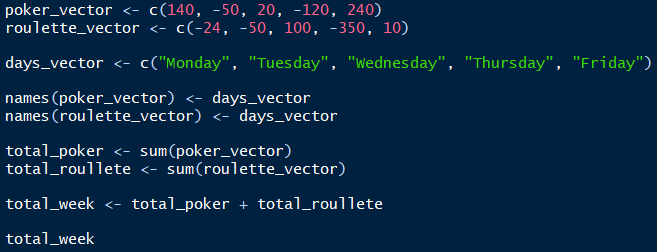
**Conditional Statements**



**Vector Math**



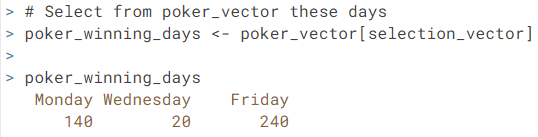
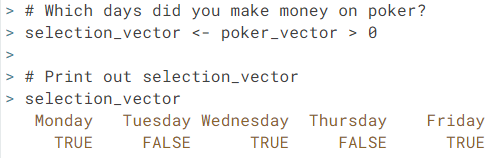
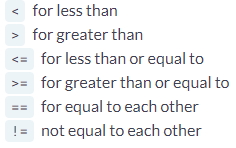
**CAREFUL  
R adds vector by position ONLY. If you give the vectors names, R will NOT add elements based on the names associated with each element.**



You can test whether one value is greater than another 🡺 total\_poker > total\_roulette

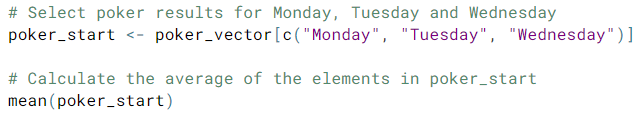
sum()

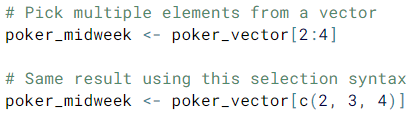
What happens when you run 🡺 poker\_vector > roulette\_vector ?



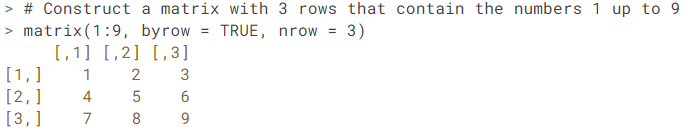
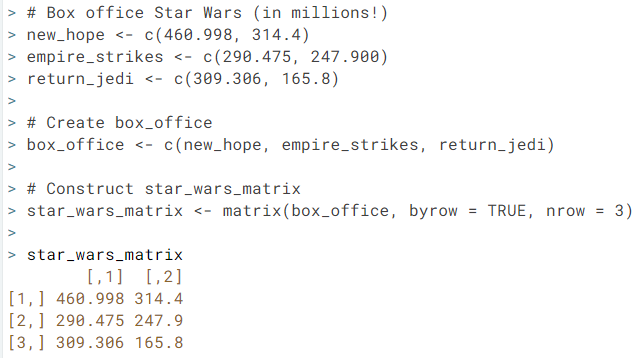
**Vector Element Selection**

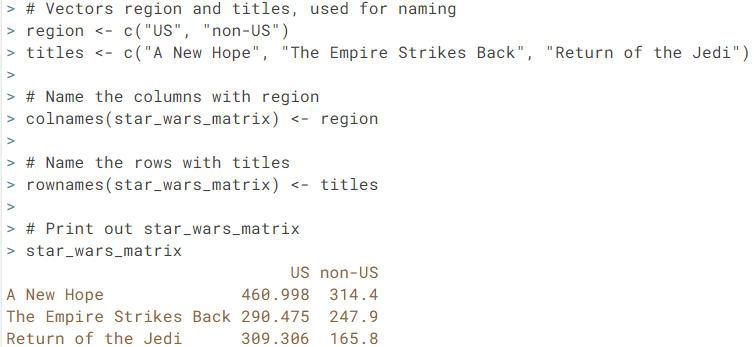




**Matrices**

mean()

A matrix is a collection of elements of the same data type  
that is arranged in rows by columns (m x n)

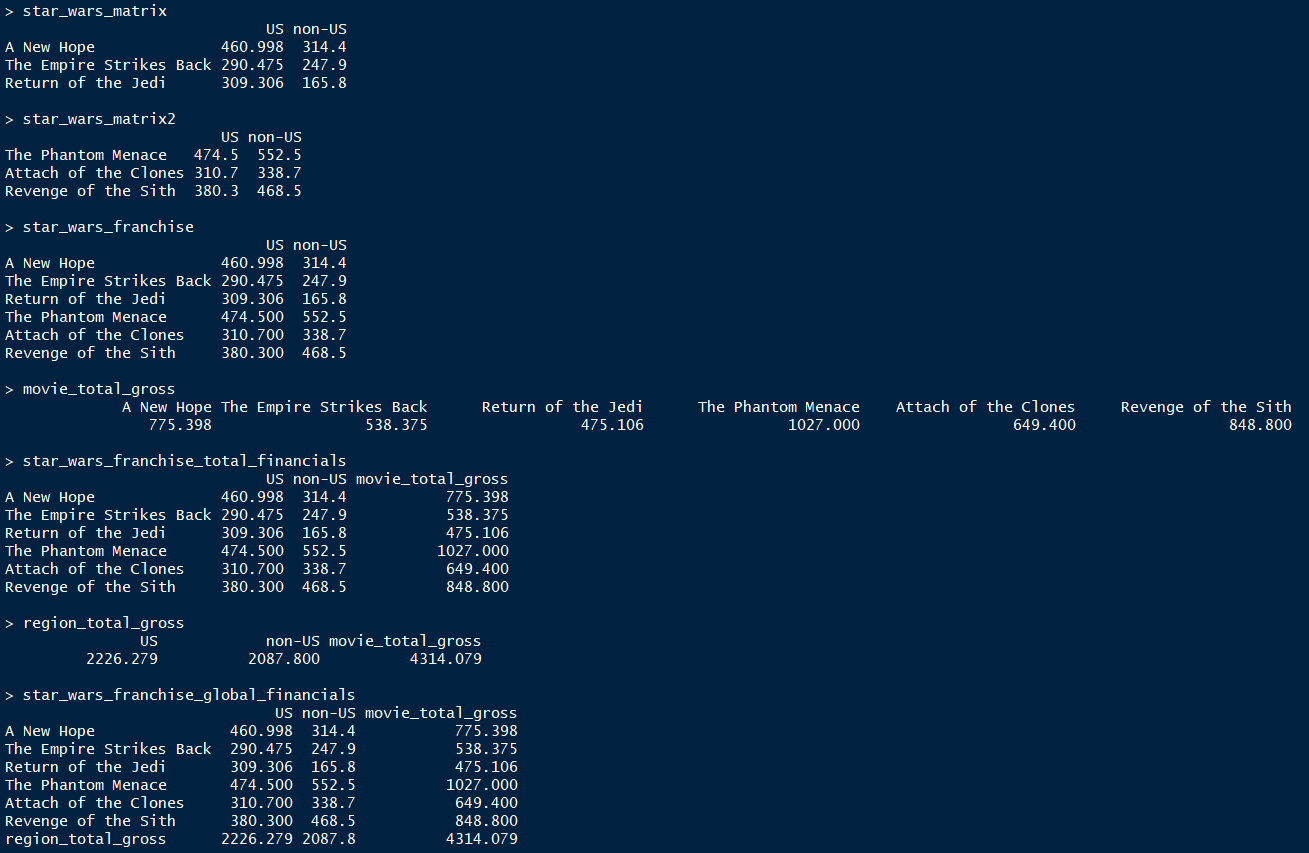
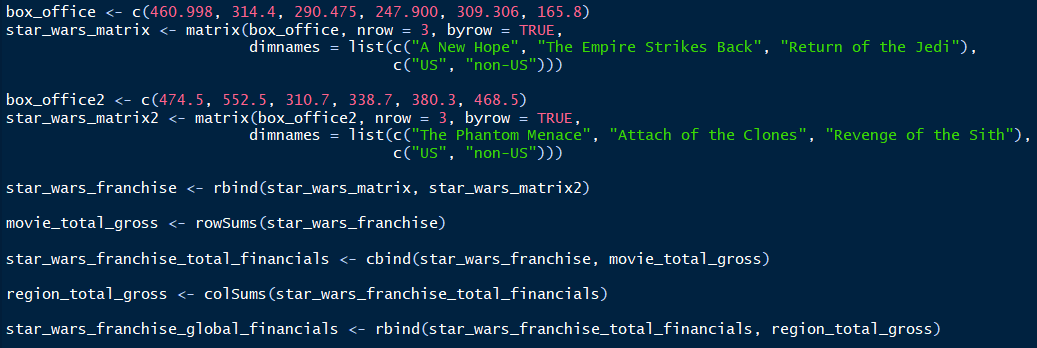


colnames()

rownames()

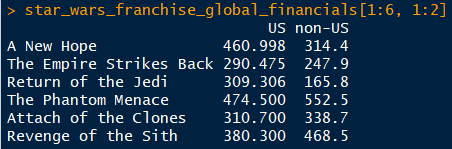
matrix()

**Matrix Math**



dimnames()

rowSums()  
colSums()  
cbind()  
rbind()

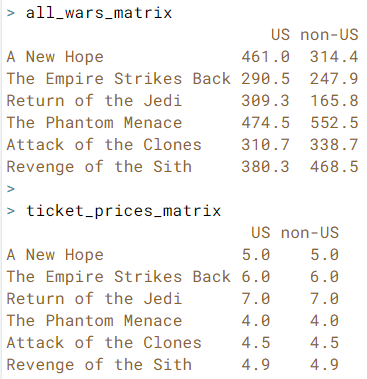


**row selection**

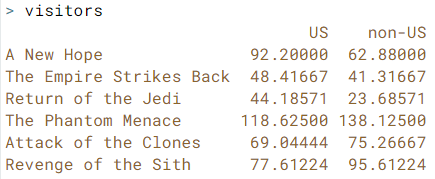
**column selection**

**Matrix Selection**

**More Matrix Math**



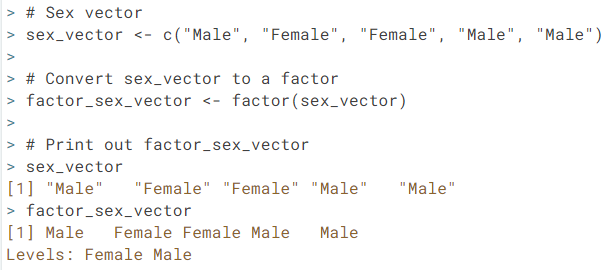
Note: in R, A \* B, where A and B are matrices, is simply performing an element-wise multiplication. To multiply two matrices in the classical linear algebra definition of matrix multiplication, you have to use %\*%



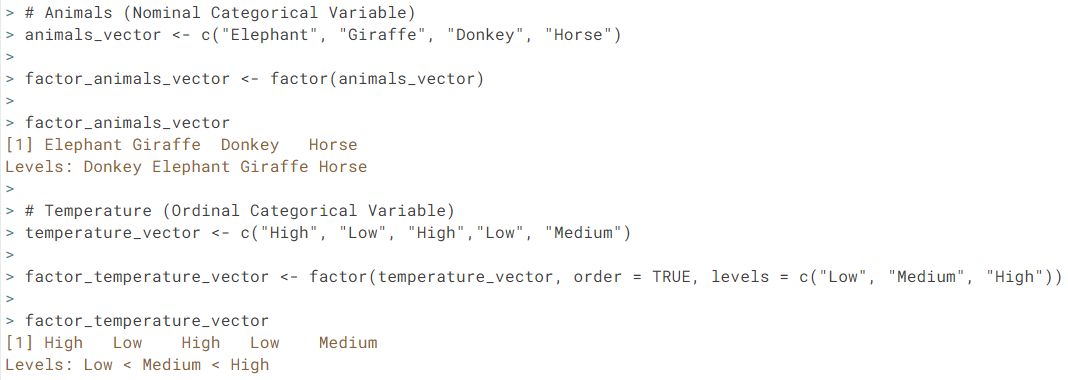
**Factors**

A factor refers to a statistical data type used to store categorical variables. Categorical variables can belong to a limited number of categories. It is important that R knows whether it is dealing with a continuous or a categorical variable, as the statistical models you will develop in the future treat both types differently.

The factor() function is used to create factors in R.

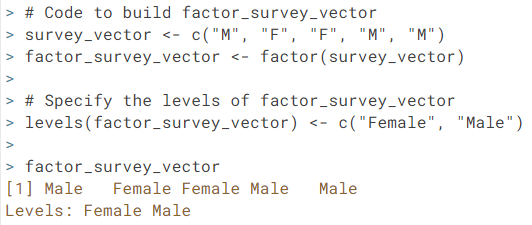


factor()

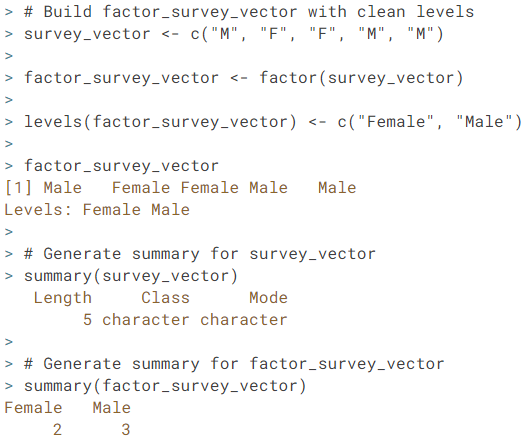


**order level**

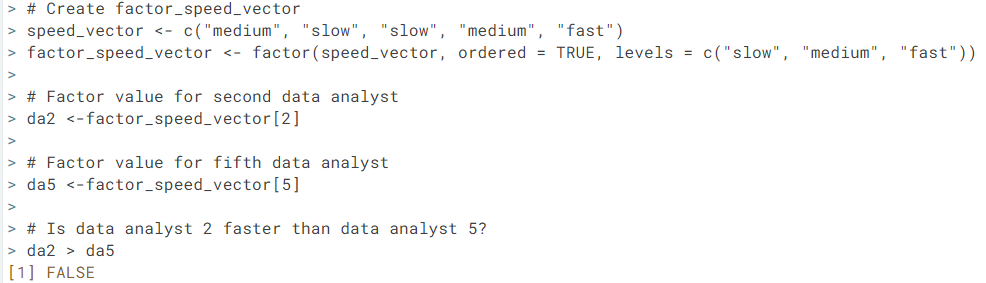
**Factor Levels**



**Careful:** The order with which you assign levels is important. If you do not specify the levels of the factor when creating a vector, R will automatically assign them alphabetically. Therefore when relabeling levels the order of elements in the assignment vector is critical.



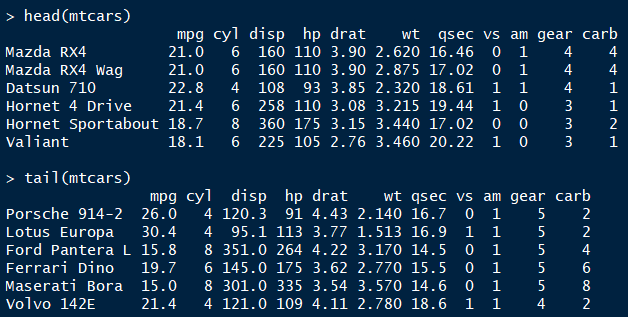
summary()



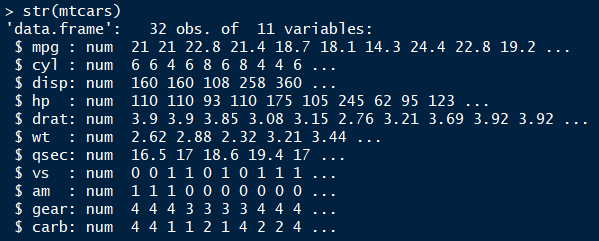
**comparison in context of factor vectors works for ordinal categorical variables**

**Data Frame**

A data frame has variables of a data set as columns and the observations as rows. With each column, the data type must be the same, however, the data type can differ across columns.



head()  
tail()  
str()



**Creating a Data Frame**

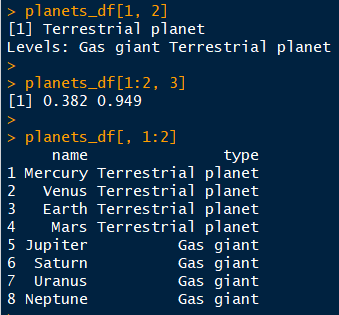


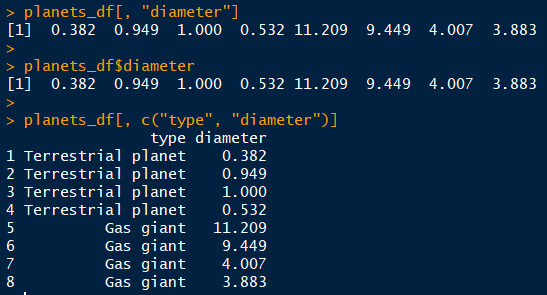
data.frame()

**Selecting Elements from a Data Frame**



data.frame()

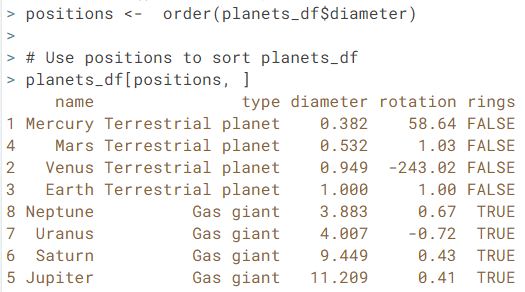
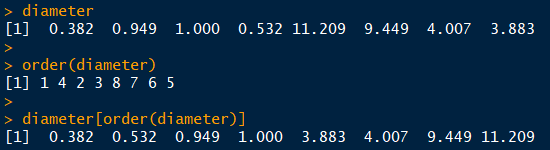


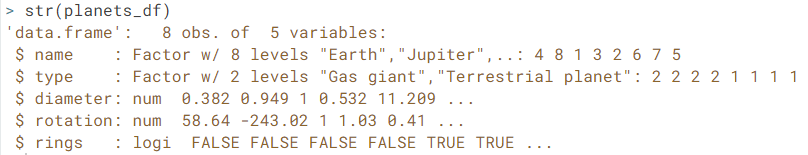
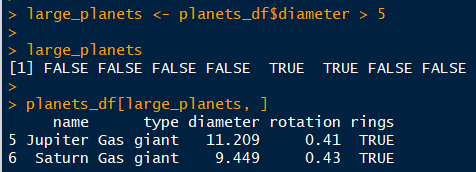




subset()

order()

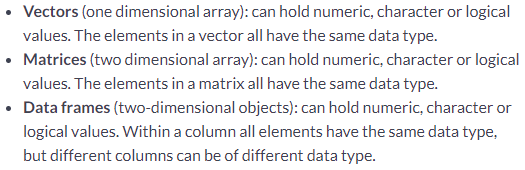


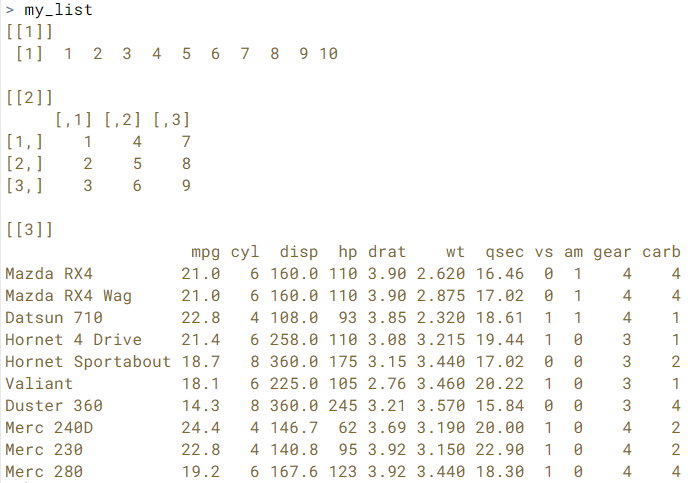
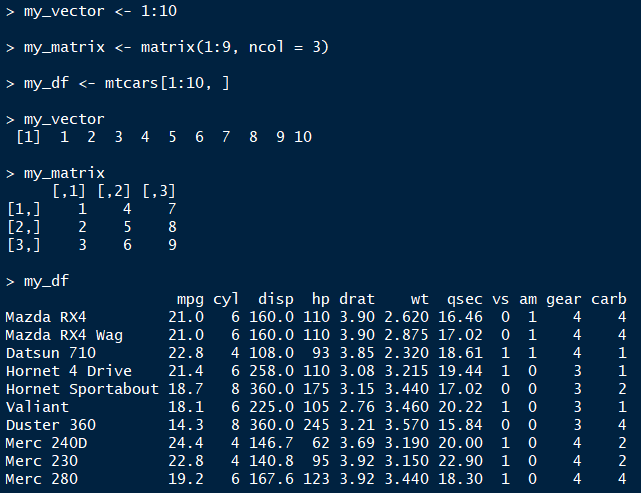
**Data Structure So Far…**

**List**



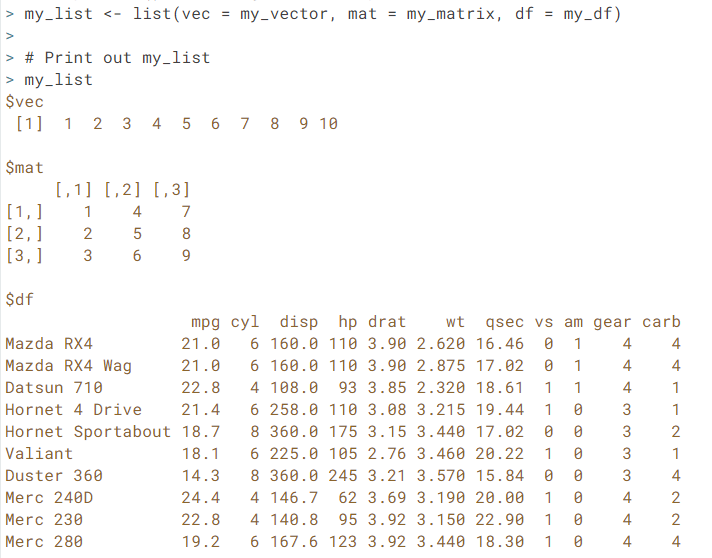
data.frame()

A list in R allows you to gather a variety of objects under one name (i.e. the name of the list) in an ordered way. These objects can be matrices, vectors, data frames, even other lists, etc. It is not even required that these objects are related to each other in any way.



list()

**List (continued)**



Named List 🡺 the names in the list (vec, mat, df, …) are referred to as ***components***.

**List Element Selection and Appending Components**

