SESION 2

1.- CON LOS VECTORES:NOMBRE, EDAD, PESO , caseid CONSTRUIR UN DATA FRAME

nombre <- c("juan", "pp", "iñaky", "amparo", "mariano", "juancar", "fulano", "elefante")

edad <- c(23, 24, 45, 67, 32, 56, 78, 45)

peso <- c(34, 34, 56, 78, 34, 56, 76, 87)

caseid <- c(1:8)

df1<- **data.frame**(caseid, nombre, edad, peso)

Ampliar un dataframe: con los vectores diabetes, estado

diabetes <- **c**("Tipo1", "Tipo1", "Tipo2", "Tipo2", "Tipo1", "Tipo1", "Tipo2", "Tipo1")

estado <- **c**("bueno", "malo", "bueno", "bueno", "bueno", "malo", "bueno", "malo")

df2 <- **data.frame**(df1, diabetes, estado)

**Extra**: Crear un tabla cruzada.

**table**(df$diabetes, df$estado)

**Funciones** attach(),para acceder más *fácilmente*

**La grámatica de dplyr**

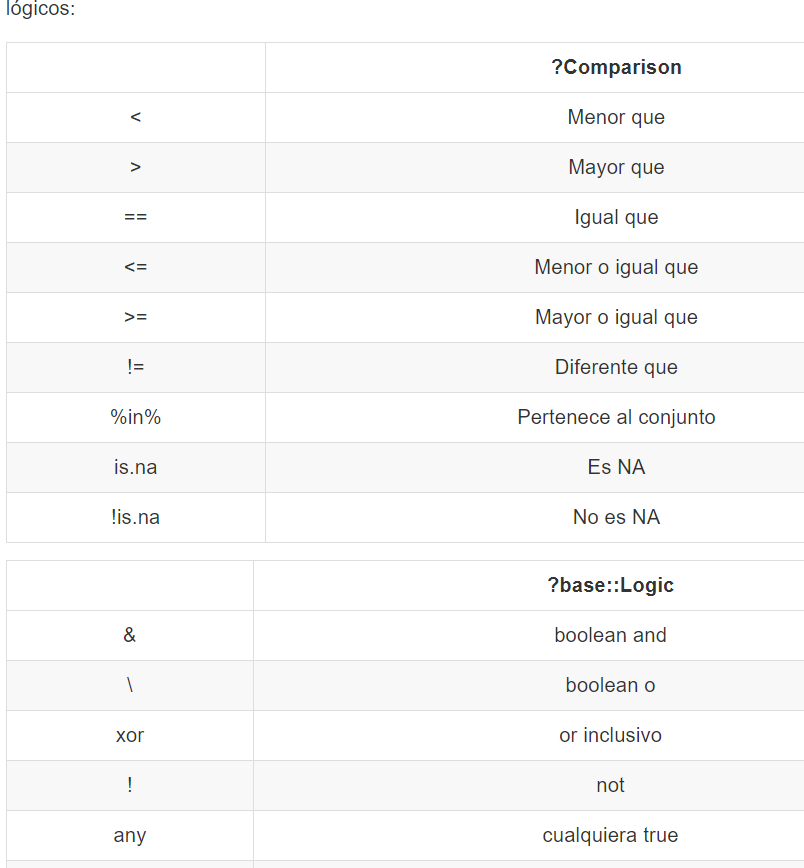
Algunas de los principales "verbos" del paquete **dplyr** son:

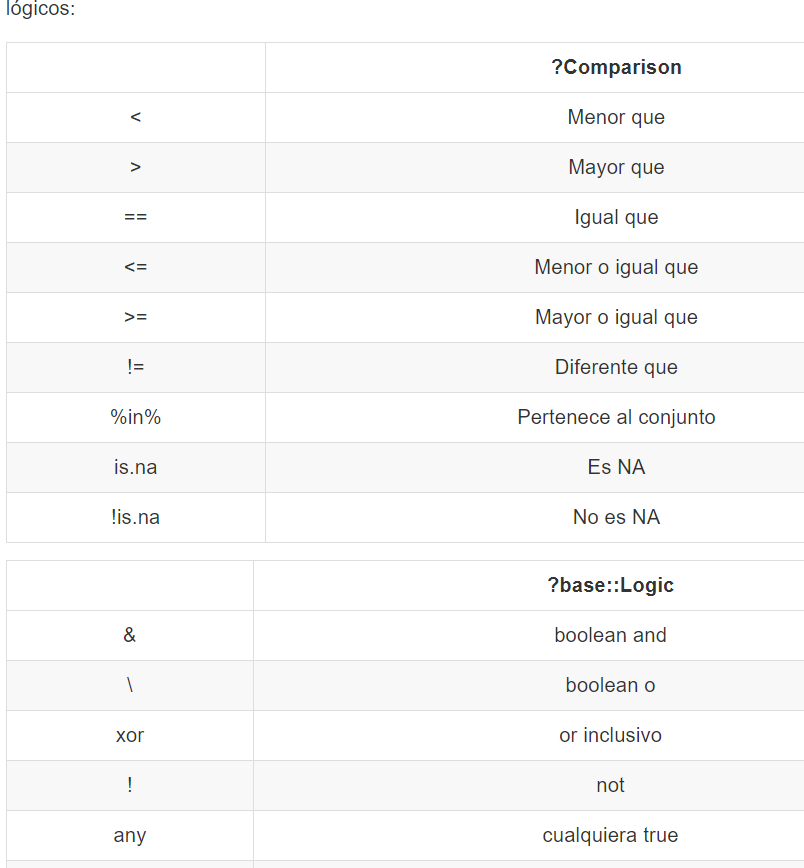
* ***select***: devuelve un conjunto de columnas
* ***filter***: devuelve un conjunto de filas según una o varias condiciones lógicas
* ***arrange***: reordena filas de un data frame
* ***rename***: renombra variables en una data frame
* ***mutate***: añade nuevas variables/columnas o transforma variables existentes
* ***summarise/summarize***: genera resúmenes estadísticos de diferentes variables en el data frame

Todas las funciones que discutiremos en este capítulo tienen en común una serie de argumentos. En particular,

1. El primer argumento es el data frame
2. Los otros argumentos describen que hacer con el data frame especificado en el primer argumento, podemos referirnos a las columnas en el data frame directamente sin utilizar el operador $, es decir sólo con el nombre de la columna/variable.
3. El valor de retorno es un nuevo data frame.
4. Los data frames deben estar bien organizados/estructurados, es decir debe existir una observación por columna y, cada columna representar una variable, medida o característica de esa observación. Para ello, es muy útil es uso del paquete **tidy**. (lo veremos en capítulos posteriores).

Select(data, variable)





Filter(data, variable<=?)

# arrange()

La función **arrange()** se utiliza para ordenar las filas de un data frame de acuerdo a una o varias columnas/variables.

Por defecto **arrange()** ordena las filas por orden ascendente:

arrange(storms, desc(wind))

# rename()

Renombrar una variable en un data frame es sorprendentemente en **R** muy difícil de realizar. La función **rename()** esta diseñada para hacer este proceso de una forma más fácil.

Para cambiar los nombres de las variables en el data frame  a castellano, podriamos hacerlo de la forma siguiente:

> rename(data.frame, tormenta = storm, viento = wind, presion = pressure, fecha = date )

**mutate()**

Con la función mutate() podemos computar tranformaciones de variables en un data frame. A menudo, tendremos la necesidad de crear nuevas variables que se calculan a partir de variables existentes,mutate() nos proporciona una interface clara para realizar este tipo de operaciones.

A continuación podemos crear una nueva varible **ratio** con el ratio entre la presión y el viento:

> mutate(data, ratio = pressure/wind)