1. Dataframes

\* Busca los datasets “beaver1” y “beaver2” que contienen información sobre la temperatura corporal de dos castores. Añade una columna llamada “ID” al dataset beaver1 que tenga siempre el valor 1. De forma similar añade una columna “ID” al dataset beaver2 que tenga siempre el valor 2. A continuación concatena de forma vertical los dos dataframes y busca el subset de datos donde ambos Castores están activos.

\* Vamos a trabajar con un ejemplo que viene por defecto en la instalación de R USArrests. Este data frame contiene la información para cada estado Americano de las tasas de criminales (por 100.000 habitantes). Los datos de las columnas se refieren a Asesinatos, violaciones y porcentaje de la población que vive en áreas urbanas. Los datos son de 1973. Contesta a las siguientes preguntas sobre los datos

- Las dimensiones del dataframe

- La longitud del dataframe (filas o columnas)

- Numero de columnas

- ¿Cómo calcularías el número de filas?

- Obtén el nombre de las filas y las columnas para este data frame

- échale un vistazo a los datos, por ejemplo a las seis primeras filas

- Ordena de forma decreciente las filas de nuestro data frame según el porcentaje de población en el área urbana. Para ello investiga la función order () y sus parámetros.

- ¿Podrías añadir un segundo criterio de orden?, ¿cómo?

- Muestra por pantalla la columna con los datos de asesinato

- Muestra las tasas de asesinato para el segundo, tercer y cuarto estado

- Muestra las primeras cinco filas de todas las columnas

- Muestra todas las filas para las dos primeras columnas

- Muestra todas las filas de las columnas 1 y 3

- Muestra solo las primeras cinco filas de las columnas 1 y 2

- Extrae las filas para el índice Murder

Vamos con expresiones un poco mas complicadas:…

-¿Que estado tiene la menor tasa de asesinatos? ¿qué línea contiene esa información?, obtén esa información

¿Que estados tienen una tasa inferior al 4%?, obtén esa información?

¿Que estados están en el cuartil superior (75) en lo que a población en zonas urbanas se refiere?

\* Carga el set de datos CO2 y realiza las siguientes acciones:

a. Ordena alfabéticamente los datos en función de la variable Plant. Recuerda que Plant es un factor. Imprime el resultado por pantalla para comprobarlo

b. Ordena los datos en función del incremento de la variable uptake y el orden alfabético de la planta (en ese orden)

c. Ordena de nuevo los datos en function del increment de la variable uptake y el orden alfabético reverso de la planta (en ese orden)

* Para este ejercicio vamos a usar el dataset state.x77. Asegurate de que el objeto es un dataframe, si no lo es fuerza su conversión.
  + Averigua cuantos estados tienen ingresos (Income) menores de 4300. Pista investiga subset()
  + Averigua cual es el estado con los ingresos mas altos.
  + Crea un data frame 2 df2 con los datasets existentes en R: state.abb, state.area, state.division, state.name, state.region. Las filas tienen que ser los nombres de los estados.
  + Elimina de todas las variables la palabra state. Busca alguna función para strings.
  + Elimina de todas las variables la palabra state. Busca alguna función para strings.
  + Añade por columnas el nuevo dataframe df2 al dataframe state.x77  
    Elimina las variables Life Exp, HS Grad, Frost, abb, y are.
  + Añade una variable que categorice el nivel de formacion (illiteracy) de manera que

[0,1) is low, [1,2) is some, [2, inf) is high.

Pista. Hazlo de dos formas usando la función cut() y usando ifelse()

* + Encuentra que estado del oeste (west) tiene la formación mas baja y los mayors ingresos. ¿Que estado es?

\* Crea un dataframe df with 40 columns, as follows:  
df <- as.data.frame(matrix(sample(1:5, 2000, T), ncol=40))

1. Ordena el dataframe por columnas, de izquierda a derecha en orden creciente
2. Ordena el dataframe por columnas, de izquierda a derecha en orden decreciente
3. Ordena el dataframe por columnas, de derecha a izquierda en orden creciente
4. Importando información

\* Vamos a trabajar con otro dataframe. Descarga el fichero student.txt de la plataforma PRADO, almacena la información en una variable llamada “students”. Ten en cuenta que los datos son tab-delimited y tienen un texto para cada columna. Comprueba que R ha leído correctamente el fichero imprimiendo el objeto en la pantalla

> students

-Imprime solo los nombres de la columnas

-Llama a la columna “height” solo

-¿Cuantas observaciones hay en cada grupo?. Utiliza la función table(). Este comando se puede utilizar para crear tablas cruzadas (cross-tabulation)

-Crea nuevas variables a partir de los datos que tenemos. Vamos a crear una variable nueva “sym” que contenga M si el genero es masculino y F si el genero es femenino. Busca en la ayuda información sobre la función ifelse(). Crea una segunda variable “colours” cuyo valor será “Blue” si el estudiante es de kuopio y “Red” si es de otro sitio.

- Con los datos anteriores de height y shoesize y las nuevas variables crea un nuevo data.frame que se llame students.new

- Comprueba que la clase de student.new es un dataframe

- Crea dos subsets a partir del set de datos student. Divídelo dependiendo del sexo. Para ello primero comprueba que estudiantes son hombres (male). Pista: busca información sobre la función which().

-Basándote en esa selección dada por which() toma solo esas filas del dataset student para generar el subset stundent.male

- Repite el procedimiento para seleccionar las estudiantes mujeres (females)

- Utiliza la function write.table() para guarder el contenido de student.new en un archivo.

1. Lists

\*Las listas son colecciones de objetos que pueden tener modos diferentes (e.g. numéricos, vectores, arrays..)

# Ejemplo de cómo crear una lista. Ejecuta los comandos y describe que es lo que ocurre

* my\_list <- list(name="Fred", wife="Mary", no.children=3, child.ages=c(4,7,9)) attributes(my\_list);
* names(my\_list) my\_list[2];
* my\_list[[2]]
* my\_list$wife.
* my\_list[[4]][2]
* length(my\_list[[4]])
* my\_list$wife <- 1:12
* my\_list$wife <- NULL.
* my\_list <- c(my\_list, list(my\_title2=month.name[1:12]))  
  unlist(my\_list);
* data.frame(unlist(my\_list));
* matrix(unlist(my\_list));

1. table()

\* La función table() cuenta el numero de elementos repetidos en un vector. Es la función más básica de clustering.

Cuenta el numero de entradas idénticas en la variable Sepal.Length del dataset iris.

1. Como ordenar datos, hacer selecciones con if(), calcular condicionales totales, transponer columnas y filas

Vamos a volver a utilizar el datasets mtcars.

* Ordena este data set de forma ascendente según su valo de hp. PISTA: with()
* Hazlo ahora de forma descendente
* Calcula la media de la columna mpg.
* Calcula la media de mpg para aquellos datos cuyo valor de hp sea menor que 150 y por separado para aquellos cuyo valor de hp sea mayor o igual a 150
* Busca los valores únicos de la columna cyl de mtcars. PISTA unique()
* Obten los datos de mpg cyl disp hp para “Toyota Corolla"
* Crea una nueva variable mpgClass de tipo categórico cuyo valor es “Low“ si el valor de mpg es menor que la media de la columna mpg y “High” si es mayor que la media de mpg. PISTA ifelse(). Combina ese comando con with() para añadir la nueva variable a mtcars
* ¿qué pasa cuando ejecutas este comando?