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.

```
> beaver1.new <- beaver1
> beaver1.new$ID<-1
> beaver2.new <- beaver2
> beaver2.new$ID<-2

> beaver12[c(which(beaver12$activ == 1)),]
```

* 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 tienela 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

```
> CO2.ordered <- CO2[c(order(as.vector(CO2$Plant))),]
> CO2.ordered
   Plant        Type  Treatment conc uptake
64   Mc1 Mississippi    chilled   95   10.5
65   Mc1 Mississippi    chilled  175   14.9
66   Mc1 Mississippi    chilled  250   18.1
67   Mc1 Mississippi    chilled  350   18.9
68   Mc1 Mississippi    chilled  500   19.5
69   Mc1 Mississippi    chilled  675   22.2
70   Mc1 Mississippi    chilled 1000   21.9
71   Mc2 Mississippi    chilled   95    7.7
72   Mc2 Mississippi    chilled  175   11.4
73   Mc2 Mississippi    chilled  250   12.3
74   Mc2 Mississippi    chilled  350   13.0
75   Mc2 Mississippi    chilled  500   12.5
76   Mc2 Mississippi    chilled  675   13.7
77   Mc2 Mississippi    chilled 1000   14.4
78   Mc3 Mississippi    chilled   95   10.6
79   Mc3 Mississippi    chilled  175   18.0
80   Mc3 Mississippi    chilled  250   17.9
81   Mc3 Mississippi    chilled  350   17.9
82   Mc3 Mississippi    chilled  500   17.9
83   Mc3 Mississippi    chilled  675   18.9
84   Mc3 Mississippi    chilled 1000   19.9
```

43	Mn1	Mississippi	nonchilled	95	10.6
44	Mn1	Mississippi	nonchilled	175	19.2
45	Mn1	Mississippi	nonchilled	250	26.2
46	Mn1	Mississippi	nonchilled	350	30.0
47	Mn1	Mississippi	nonchilled	500	30.9
48	Mn1	Mississippi	nonchilled	675	32.4
49	Mn1	Mississippi	nonchilled	1000	35.5
50	Mn2	Mississippi	nonchilled	95	12.0
51	Mn2	Mississippi	nonchilled	175	22.0
52	Mn2	Mississippi	nonchilled	250	30.6
53	Mn2	Mississippi	nonchilled	350	31.8
54	Mn2	Mississippi	nonchilled	500	32.4
55	Mn2	Mississippi	nonchilled	675	31.1
56	Mn2	Mississippi	nonchilled	1000	31.5
57	Mn3	Mississippi	nonchilled	95	11.3
58	Mn3	Mississippi	nonchilled	175	19.4
59	Mn3	Mississippi	nonchilled	250	25.8
60	Mn3	Mississippi	nonchilled	350	27.9
61	Mn3	Mississippi	nonchilled	500	28.5
62	Mn3	Mississippi	nonchilled	675	28.1
63	Mn3	Mississippi	nonchilled	1000	27.8
22	Qc1	Quebec	chilled	95	14.2
23	Qc1	Quebec	chilled	175	24.1
24	Qc1	Quebec	chilled	250	30.3
25	Qc1	Quebec	chilled	350	34.6
26	Qc1	Quebec	chilled	500	32.5
27	Qc1	Quebec	chilled	675	35.4
28	Qc1	Quebec	chilled	1000	38.7
29	Qc2	Quebec	chilled	95	9.3
30	Qc2	Quebec	chilled	175	27.3
31	Qc2	Quebec	chilled	250	35.0
32	Qc2	Quebec	chilled	350	38.8
33	Qc2	Quebec	chilled	500	38.6
34	Qc2	Quebec	chilled	675	37.5
35	Qc2	Quebec	chilled	1000	42.4
36	Qc3	Quebec	chilled	95	15.1
37	Qc3	Quebec	chilled	175	21.0
38	Qc3	Quebec	chilled	250	38.1
39	Qc3	Quebec	chilled	350	34.0
40	Qc3	Quebec	chilled	500	38.9
41	Qc3	Quebec	chilled	675	39.6
42	Qc3	Quebec	chilled	1000	41.4
1	Qn1	Quebec	nonchilled	95	16.0
2	Qn1	Quebec	nonchilled	175	30.4
3	Qn1	Quebec	nonchilled	250	34.8
4	Qn1	Quebec	nonchilled	350	37.2
5	Qn1	Quebec	nonchilled	500	35.3
6	Qn1	Quebec	nonchilled	675	39.2
7	Qn1	Quebec	nonchilled	1000	39.7
8	Qn2	Quebec	nonchilled	95	13.6
9	Qn2	Quebec	nonchilled	175	27.3
10	Qn2	Quebec	nonchilled	250	37.1
11	Qn2	Quebec	nonchilled	350	41.8
12	Qn2	Quebec	nonchilled	500	40.6
13	Qn2	Quebec	nonchilled	675	41.4
14	Qn2	Quebec	nonchilled	1000	44.3
15	Qn3	Quebec	nonchilled	95	16.2
16	Qn3	Quebec	nonchilled	175	32.4
17	Qn3	Quebec	nonchilled	250	40.3
18	Qn3	Quebec	nonchilled	350	42.1
19	Qn3	Quebec	nonchilled	500	42.9
20	Qn3	Quebec	nonchilled	675	43.9
21	Qn3	Quebec	nonchilled	1000	45.5

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

```
> CO2.ordered <- CO2[order(CO2.ordered$uptake,CO2.ordered$Plant),]
```

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)

```
> CO2.ordered <- CO2[order(c(CO2.ordered$uptake),c(CO2.ordered$Plant
),decreasing = c(F,T)),]
```

- 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()

      Dos formas:

```
> nrow(subset(state.x77,state.x77$Income<4300))
[1] 20

> nrow(state.x77[which(state.x77$Income < 4300),])
[1] 20
```

    - Averigua cual es el estado con los ingresos mas altos.
    -

```
> state.x77[which(state.x77$Income == max(state.x77$Income)),]
       Population Income Illiteracy Life Exp Murder HS Grad Fros   Area
Alaska        365   6315        1.5    69.31   11.3    66.7   152 566432
```

    - Crea un data frame2 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.

```
> df2 <- data.frame(state.abb,state.area,state.division,state.reg
ion,row.names=state.name)
```

- Elimina de todas las variables la palabra state. Busca alguna función para strings.

```
> paso1 <- unlist(strsplit(colnames(df2),"state."))
> paso1
[1] ""         "abb"      ""         "area"     ""         "divis
ion" ""         "region"
> paso1[which(paso1 != "")]
[1] "abb"      "area"     "division" "region"
> colnames(df2) <- paso1[which(paso1 != "")]
> colnames(df2)
[1] "abb"      "area"     "division" "region"
```

- Bora la variable div de df2. Estas borrando una única variable del dataframe

```
> df2$division<-NULL
```

- Añade por columnas el nuevo dataframe df2 al dataframestate.x77 Eliminalas variables Life Exp, HS Grad, Frost, abb, y are.

```
> state.x77 <- data.frame(state.x77,df2)
> state.x77[c("Life.Exp","HS.Grad","Frost","abb","area")] <- NULL
```

- 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()

```
> state.x77$categorized <- ifelse(state.x77$Illiteracy>=2,"high",ife
lse(state.x77$Illiteracy < 1, "low", "some"))
```

- Encuentra que estado del oeste (west) tiene la formación mas baja y los mayores ingresos. ¿Qué estado es?

* Crea un dataframe df with 40 columns, as follows:

```
df<- as.data.frame(matrix(sample(1:5, 2000, T), ncol=40))
```

a. Ordena el dataframe por columnas, de izquierda a derecha en orden creciente

b. Ordena el dataframe por columnas, de izquierda a derecha en orden decreciente
c. Ordena el dataframe por columnas, de derecha a izquierda en orden creciente

2. 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 datosstudent. 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 datasetstudent para generar el subsetstundent.male

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

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

3. 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);`

En este primer comando creamos una lista que tiene 4 campos: nombre, mujer, numero de hijos y edad de los hijos.

Con el comando *attributes* podemos ver los campos que tienen cada uno de los elementos de la lista.

- `names(my_list) my_list[2];`

Con el comando *names* podemos ver el nombre de los campos que posee nuestra lista. El comando *my_list[2]* muestra el elemento 2 de la lista, tanto el nombre como el valor que tiene.

- `my_list[[2]]`

En este caso nos muestra solamente el contenido de la componente 2 de la lista, sin decirnos el nombre de la misma.

- `my_list$wife.`

Muestra el valor del campo *wife* de la lista.

- `my_list[[4]][2]`

En este caso veremos la edad del segundo hijo, en primer lugar los dos primeros corchetes dicen que queremos ver el valor del atributo 4 y al ser un vector podemos especificar la componente que queremos ver, en este caso la 2.

- `length(my_list[[4]])`

Como my_list[[4]] es un vector, con length averiguamos el tamaño del mismo.

- `my_list$wife<- 1:12`

Ha asignado a la componente *wife* de la lista un vector con los valores del 1 al 12.

- `my_list$wife<- NULL.`

Con este comando eliminamos la componente *wife* de la lista.

- ```
  my_list<- c(my_list,
  list(my_title2=month.name[1:12]))
  ```

Se ha añadido un atributo a la lista que se llama my_title2 y que contiene un vector con los nombres de los meses del año.

```
unlist(my_list);
```

En este caso convierte todos los elementos de la lista a carácter.

- `data.frame(unlist(my_list));`

Crea un data.frame con los elementos de la lista, pero sin sentido aparente.

- `matrix(unlist(my_list));`

Crea una matriz con 17 filas y 1 columna colocando los elementos de la lista como strings.

4. 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.

```
> table(iris$Sepal.Length)

4.3 4.4 4.5 4.6 4.7 4.8 4.9   5 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9
  6 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 6.9   7 7.1 7.2 7.3 7.4 7.6 7.7
7.9
  1   3   1   4   2   5   6  10   9   4   1   6   7   6   8   7   3
  6   6   4   9   7   5   2   8   3   4   1   1   3   1   1   1   4
  1
```

5. 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()

```
> mtcars <- mtcars[order(mtcars$hp),]
```

- ✓ Hazlo ahora de forma descendente

```
> mtcars <- mtcars[order(mtcars$hp,decreasing = TRUE),]
```

- ✓ Calcula la media de la columna mpg.

```
> mean(mtcars$mpg)
[1] 20.09062
```

- ✓ 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

```
> mean(mtcars$hp[which(mtcars$hp<150)])
```

```
[1] 93.52941

> mean(mtcars$hp[which(mtcars$hp>=150)])
[1] 206.9333
```

- ✓ Busca los valores únicos de la columna cyl de mtcars. PISTA unique()

```
> unique(mtcars$cyl)
[1] 8 6 4
```

- ✓ Obten los datos de mpg cyl disp hp para "Toyota Corolla"

```
> mtcars["Toyota Corolla",c("mpg","cyl","disp","hp")]
                mpg cyl disp hp
Toyota Corolla 33.9   4 71.1 65
```

- ✓ Crea una nueva variable mpgClass de tipo categórico cuyovalor 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

```
> mtcars$mpgClass <- ifelse(mtcars$mpg < mean(mtcars$mpg), "Lo
w", "High")
```

- ✓ ¿qué pasa cuando ejecutas este comando?