

Lab Listas en Lenguaje R

Objetivo

- Mostrar al participante el uso de la estructura de datos lista y la funciones principales en el lenguaje R.

Contenido

Creación de Listas

1. Creación de Listas en R

Podemos entender una lista como un contenedor de objetos que pueden ser de cualquier clase: números, vectores, matrices, funciones, data.frames, incluso otras listas.

```
> edad <- c(22, 34, 29, 25, 30, 33, 31, 27, 25, 25)
> tiempo <- c(14.21, 10.36, 11.89, 13.81, 12.03, 10.99, 12.48, 13.37, 12.29, 11.92)
> sexo <- c("M", "H", "H", "M", "M", "H", "M", "M", "H", "H")
> misDatos <- data.frame(edad, tiempo, sexo)
> misDatos
  edad tiempo sexo
1   22  14.21    M
2   34  10.36    H
3   29  11.89    H
4   25  13.81    M
5   30  12.03    M
6   33  10.99    H
7   31  12.48    M
8   27  13.37    M
9   25  12.29    H
10  25  11.92    H
> |
```

```
> A=matrix(1:9,nrow=3)
```

```
> M=matrix(1,4,nrow=2)
```

```
> MiLista <- list(misDatos,A,M=M,x=c(1,2,3,4),e=exp(1))
```

```
> MiLista
```

2. Agrega elementos a la lista

```
> mi.lista <- list(a = 1:3, b = c("hola", "adiós"))
> mi.lista$z <- matrix(1:4, 2, 2)
```

Selección de Elementos

3. Selecciona los siguientes elementos de la lista

```
MiLista$misDatos
```

```
## NULL
```

```
MiLista[[1]]
```

```
## edad tiempo sexo
```

```
## 1 22 14.21 M
```

```
## 10 25 11.92 H
```

```
## 9 25 12.29 H
## 4 25 13.81 M
## 8 27 13.37 M
## 3 29 11.89 H
## 5 30 12.03 M
## 7 31 12.48 M
## 6 33 10.99 H
## 2 34 10.36 H
```

`MiLista$A`

```
## NULL
```

`MiLista[[2]]`

```
## [1] [2] [3]
## [1,] 1 4 7
## [2,] 2 5 8
## [3,] 3 6 9
```

`MiLista$M`

```
## [1] [2] [3] [4]
## [1,] 1 1 1 1
## [2,] 1 1 1 1
```

`MiLista$x`

```
## [1] 1 2 3 4
```

`MiLista$e`

```
## [1] 2.71828
```

4. Es conveniente recordar que en R las tablas son esencialmente listas de vectores de la misma longitud, verificar si iris es una lista.

```
> is.list(iris)
```

5. Explora iris con las funciones names y str

6. Aplica la funcion lapply y explica el resultado

```
> lapply(airquality[, 1:4], mean, na.rm = TRUE)
```

7. Aplica sapply a la lista airquality, Cual es la diferencia con lapply?

```
> sapply(airquality[, 1:4], mean, na.rm = TRUE)
```

8. Aplica la funcion split a iris, recordar que split permite partir una tabla en trozos de acuerdo con un vector que define los grupos.

```
> tmp <- split(iris, iris$Species)
```

9. Investiga el objeto tmp: ¿qué longitud tiene? ¿qué contiene cada uno de sus componentes?

10. Usa las funciones lapply y sapply para mostrar la dimensión de cada una de las tablas que contiene tmp

11. Usa split para partir iris en dos subtablas al azar con el mismo número de filas.

12. Usa split para partir iris cinco subtablas, cada una de ellas con 30 filas distintas.