

Programming for Data Science

Lists in R language

Marco Beccuti

Università degli Studi di Torino

Dipartimento di Informatica



List in R

- it is an ordered collection of components;
- its components may be arbitrary R objects (matrix, vectors, lists, ...);
- function `list()` can be used to create lists:

```
> x = c(1 : 4)
```

```
> y = rep("ACT", 2)
```

```
> k = c(TRUE, TRUE)
```

```
> l1 = list(x, y, k)  it creates a list contains three vectors (i.e. x,y,k)
```

```
> l1
```

```
[[1]]
```

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

```
[[2]]
```

```
[1] "ACT" "ACT"
```

```
[[3]]
```

```
[1] TRUE TRUE
```

List in R

- Two lists can be concatenated as follows:

```
> l2 = list(matrix(rnorm(10), nrow = 5), 1 : 10)
> l3 = c(l1, l2)
```

- names can be associated with list elements:

```
> names(l1) = c("first", "second", "third")
```

```
$first
```

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

```
$second
```

```
[1] "ACT" "ACT"
```

```
$third
```

```
[1] TRUE TRUE
```

List in R

- We can access the list elements in the following two ways:
 - 1 how to access the element in first position in the list `/l` returning a vector

```
> /l[[1]]  
[1] 1 2 3 4  
  
> /l$first  
[1] 1 2 3 4
```

- 2 how to access the first element in the vector in first position in the list `/l`

```
> /l[[1]][1]  
[1] 1
```

- 3 how to return a new list containing the first vector in the list `/l`

```
> /l[1]  
[[1]]  
[1] 1 2 3 4
```

Exercises on Lists

- Create the following three vectors and one matrix 2×2 :

① $X = \{1, 5, 6, 19, 5\};$

② $Y = \{\text{"HOME"}, \text{"WOLF"}, \text{"ROOM"}, \text{NA}\}$

③ $Z = \{1.25, 1.50, 1.75, \dots 10\}$

④ $M = \{ 2, 3; 4, 5\}$

and stores them in the list `/l`.

- Give a name to each list element (using `names` function).
- Use the two different ways to access the 2nd element of the list `/l`.
- Access the 2nd element of the 3rd element of the list `/l`.
- Access the 2nd and 4th elements of the 1st element of the list `/l`.

Exercises on Lists

- Create the following three vectors and one matrix 2×2 :

- 1 $X = \{1, 5, 6, 19, 5\};$
- 2 $Y = \{"HOME", "WOLF", "ROOM", NA\}$
- 3 $Z = \{1.25, 1.50, 1.75, \dots 10\}$
- 4 $M = \{ 2, 3; 4, 5\}$

and stores them in the list *l1*.

```
> X = c(1, 5, 6, 19, 5)
> Y = c("HOME", "WOLF", "ROOM", NA)
> Z = seq(1, 10, by = 0.25)
> M = matrix(c(2, 3, 4, 5), nrow = 2)
> l1 = list(X, Y, Z, M)
```

Exercises on Lists

- Give a name to each list element (using `names` function).

```
> names(l1) = c("X", "Y", "Z", "M")
```

```
> l1
```

```
X
```

```
[1] 1 5 6 19 5
```

```
Y
```

```
[1] "HOME" "WOLF" "ROOM" NA
```

```
...
```

```
...
```

```
...
```

Exercises on Lists

- Use the two different ways to access the 2nd element of the list `/1`.

```
> /1[[2]]  
[1] "HOME" "WOLF" "ROOM" NA
```

```
> /1$Y  
[1] "HOME" "WOLF" "ROOM" NA
```


Exercises on Lists

- Access the 2nd element of the 3rd element of the list /1.

```
> /1[[3]][2]  
[1]1.25
```

Exercises on Lists

- Access the 2nd and 4th elements of the 1st element of the list /1.

```
> /1[[1]][c(2,4)]  
[1] 5 19
```