

# Exercices

Hubert LEVIEL

20/06/2022

## 1 - Variables et types

Créer une variable numérique x valant 30.25

```
## [1] 30.25
```

Quel est le type de x

```
## [1] "numeric"
```

Créer une variable entiere y valant 30L

```
## [1] 30
```

Sommer x et y

```
## [1] 60.25
```

Créer une variable z en transformant x en entier

```
## [1] 30
```

Créer la variable i qui contient la somme de z et y

```
## [1] 60
```

Vérifier que i est de type integer

```
## [1] TRUE
```

## 2- Formats

### Vecteurs

Créer un vecteur v contenant les nombre 1 à 10

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

Calculer le vecteur v2 contenant les carrés des nombres 1 à 10

```
## [1] 1 4 9 16 25 36 49 64 81 100
```

```
## [1] 1 4 9 16 25 36 49 64 81 100
```

```
## [1] 1 4 9 16 25 36 49 64 81 100
```

Créer un vecteur w valant 2 fois v

```
## [1] 2 4 6 8 10 12 14 16 18 20
```

Soustraire w à v2

```
## [1] -1 0 3 8 15 24 35 48 63 80
```

Retrouver ce résultat en n'utilisant que v

```
## [1] -1 0 3 8 15 24 35 48 63 80
```

Calculer la somme des 10 premiers entiers

```
## [1] 55
```

Calculer la somme des carrés des 10 premiers entiers

```
## [1] 385
```

Calculer la racine carrée de v2

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

Vérifier qu'elle vaut v

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
## [1] TRUE
```

```
## [1] FALSE
```

## Data frames

Afficher les 6 premières lignes de `iris`

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1         5.1         3.5         1.4         0.2   setosa
## 2         4.9         3.0         1.4         0.2   setosa
## 3         4.7         3.2         1.3         0.2   setosa
## 4         4.6         3.1         1.5         0.2   setosa
## 5         5.0         3.6         1.4         0.2   setosa
## 6         5.4         3.9         1.7         0.4   setosa
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1         5.1         3.5         1.4         0.2   setosa
## 2         4.9         3.0         1.4         0.2   setosa
## 3         4.7         3.2         1.3         0.2   setosa
## 4         4.6         3.1         1.5         0.2   setosa
## 5         5.0         3.6         1.4         0.2   setosa
## 6         5.4         3.9         1.7         0.4   setosa
```

Afficher la toute première ligne de `iris`

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1         5.1         3.5         1.4         0.2   setosa
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1         5.1         3.5         1.4         0.2   setosa
```

Afficher les 6 premières valeurs de la première colonne de `iris`

```
## [1] 5.1 4.9 4.7 4.6 5.0 5.4
```

```
## [1] 5.1 4.9 4.7 4.6 5.0 5.4
```

Afficher les 6 premières valeurs de la colonne `Petal.Length`

```
## [1] 1.4 1.4 1.3 1.5 1.4 1.7
```

```
## [1] 1.4 1.4 1.3 1.5 1.4 1.7
```

```
## [1] 1.4 1.4 1.3 1.5 1.4 1.7
```

Calculer la longueur maximale d'une pétale dans `iris`

```
## [1] 6.9
```

Afficher les 6 premières lignes de `iris` pour l'espèce `versicolor`

```
##      Petal.Length Sepal.Length
## 51          4.7          7.0
## 52          4.5          6.4
## 53          4.9          6.9
## 54          4.0          5.5
## 55          4.6          6.5
## 56          4.5          5.7
```

```
##      Petal.Length Sepal.Length
## 51          4.7          7.0
## 52          4.5          6.4
## 53          4.9          6.9
## 54          4.0          5.5
## 55          4.6          6.5
## 56          4.5          5.7
```

Afficher la (ou les) ligne pour laquelle la longueur de pétale est la plus longue

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width  Species
## 119          7.7          2.6          6.9          2.3 virginica
```

Nombre d'individus ayant une pétale plus courte que la moyenne

```
## [1] 57
```

```
## [1] 57
```

### 3 - Conditions

Si tous les achats sont inférieurs à 20 afficher “Raisonnable” sinon afficher “Dépensier”

```
## [1] "Raisonnable"
```

```
## [1] "Raisonnable"
```

Afficher “Raisonnable” ou “Dépensier” pour chaque jour

```
##      lundi      mardi      mercredi      jeudi      vendredi
## "Raisonnable 1" "Raisonnable 2" "Raisonnable 3" "Raisonnable 4" "Raisonnable 5"
```

Afficher “Econome” pour les jours où il y a des pourboire et pas d'achat, “NA” sinon

```
##      lundi      mardi mercredi      jeudi vendredi
## "Econome"      "NA"      "NA"      "NA"      "NA"
```

Afficher “Econome” si la somme des achats est inférieure à 25 et celle des pourboires est supérieure à 30, “NA” sinon

```
## [1] "Econome"
```

```
## [1] "Econome"
```

## 4 - Boucles

En utilisant une boucle `for`, afficher “Econome” pour les jours où il y a des pourboire et pas d’achat, “NA” sinon

```
## [1] "Econome"
## [1] "NA"
## [1] "NA"
## [1] "NA"
## [1] "NA"
```

## 5 - Fonctions

En utilisant `lapply`, afficher “Econome” pour les jours où il y a des pourboire et pas d’achat, “NA” sinon

```
## [[1]]
## [1] "Econome"
##
## [[2]]
## [1] "NA"
##
## [[3]]
## [1] "NA"
##
## [[4]]
## [1] "NA"
##
## [[5]]
## [1] "NA"
```

```
##      lundi      mardi mercredi   jeudi  vendredi
## "Econome"    "NA"      "NA"    "NA"    "NA"
```

## 6 - CSV

Réécrire le filtre sur les “female” sans le `which()`

```
##      Music Slow.songs.or.fast.songs Dance Folk Country Classical.music
## 1      5                        3      2      1          2              2
## 2      4                        4      2      1          1              1
## 3      5                        5      2      2          3              4
## 4      5                        3      2      1          1              1
## 5      5                        3      4      3          2              4
## 7      5                        5      5      3          1              2
```

Réécrire le filtre sur les “female” âgées de “Moins de 20 ans” sans le `which()`

```
##      Music Slow.songs.or.fast.songs Dance Folk Country Classical.music
## 2      4                        4      2      1          1              1
## 9      5                        3      3      1          1              2
## 10     5                        3      2      5          2              2
```

```
## 11      5      3      3      2      1      2
## 12      5      3      1      1      1      4
## 14      5      3      5      3      2      1
```

Ecrire la fonction `my_which()` qui retourne le même résultat que `which()`

```
which(c(TRUE, FALSE, TRUE))
```

```
## [1] 1 3
```

```
my_which(c(TRUE, FALSE, TRUE))
```

```
## [1] 1 3
```

## 8 - Variable catégorielle

Existe-t'il existe un lien statistique entre le fait d'aimer les films d'action et le genre?

```
##           Action
## Genre    Non Oui
##  female 361 230
##   male   87 324
```

```
## [1] 1.69027e-35
```

Utiliser `sapply()` pour obtenir les variables ayant un for lien statistique avec le genre

```
##           Life.struggles           PC           Action
##           2.341173e-47       1.610272e-36       1.690270e-35
##           Romantic           Cars           War
##           2.372227e-35       1.394057e-34       1.113837e-31
## Science.and.technology       Shopping       Reading
##           1.258774e-30       6.381170e-25       4.215641e-23
##           Spiders           Dancing       Sci.fi
##           3.014540e-20       2.348340e-16       4.289943e-16
```

```
##           Cars
## Genre    Non Oui
##  female 494  96
##   male  193 218
```

```
## [1] 1.394057e-34
```