

## TD1 DÉBUTER AVEC R ET RStudio

L'objectif est de découvrir le fonctionnement du logiciel RStudio en écrivant des lignes de code dans un *script*. Il faudra ensuite *exécuter* ces lignes de code dans la console. Premièrement, il faut créer un fichier `td1.R` et y noter toutes les réponses des exercices. Voici quelques raccourcis claviers pratiques :

- █ [Ctrl] + [S] pour enregistrer votre fichier `td1.R` (si besoin, choisir l'encodage UTF-8)
- █ [Ctrl] + [Entrée] pour *exécuter* la ligne de code de votre script où il y a votre curseur.
- █ [Ctrl] + [↑] + [Entrée] pour *exécuter* toutes les lignes de code de votre script `td1.R`
- █ [Alt] + [↑] + [K] pour afficher tous les raccourcis clavier

▲ Ce qui est écrit dans la *console* n'est pas enregistré, il faut donc écrire son code dans le script (c'est le fichier `td1.R`) et l'*exécuter* ensuite via les raccourcis ci-dessus.

Pour faire votre compte-rendu (CR), il faut remplir le questionnaire dans Moodle. Le CR du TD1 est noté à titre d'exemple pour les 8 autres CR de TD.

$$\text{Note finale} = \frac{\text{CR2} + \text{CR3} + \text{CR4} + \text{CR5} + \text{CR6} + \text{CR7} + \text{CR8} + \text{CR9}}{8} \quad (1)$$

Toute absence injustifiée équivaudra à un 0 pour la note du CR.

**Remarque :** La question est beaucoup plus longue que la réponse attendue.

### Exercice 1.1 Création de variable numérique

1. Créer une variable `test` en lui affectant la valeur 0. Vérifier que la variable apparaît bien dans l'environnement.
2. Créer une variable `autre` en lui affectant la valeur 3. Vérifier que la variable apparaît bien dans l'environnement.
3. Calculer le produit de 5 fois la valeur contenue dans la variable `autre`. Est-ce que la valeur est dans l'environnement ? Que faut-il faire si l'on souhaite réutiliser le résultat de ce calcul par la suite ?
4. Sachant que
  - la variable `numero` vaut 16
  - et que
    - la variable `donnee` vaut 23,
    - calculer le produit de 7 fois la valeur contenue dans la variable `numero` plus la valeur contenue dans la variable `donnee` fois 4 et affecter le résultat dans la variable `valeur`.
5. Afficher la valeur contenue dans la variable `valeur`

### Correction 1.1

```
1 test <- 0
2 autre <- 3
3 autre * 5
4 numero <- 16
5 donnee <- 23
6 valeur <- 7 * numero + 4 * donnee
7 print(valeur)
```

### Exercice 1.2 Création de vecteur

1. **Créer** un vecteur allant de 1 à 5 par pas de 1 et **affecter** ce vecteur à la variable `vecteur_1`
2. **Créer** un vecteur allant de 2 à 14 par pas de 2 et **affecter** ce vecteur à la variable `vecteur_2`
3. **Afficher** la 2<sup>ème</sup> valeur du vecteur contenu dans la variable `vecteur_2`.

### Correction 1.2

```
1 vecteur_1 <- 1:5
2 vecteur_2 <- seq(from = 2, to = 14, by = 2)
3 print(vecteur_2[2])
```

### Exercice 1.3 Création de variables avec d'autres types

1. **Créer** la chaîne de caractère `Modélisation` et affecter cette chaîne de caractère à la variable `nom`
2. **Affecter** le symbole d'une valeur absente à la variable `rien`.
3. **Créer** un vecteur avec 3 chaînes de caractère et **affecter** ce vecteur à la variable `texte`.
4. **Créer** un vecteur avec 2 booléens différents et **affecter** ce vecteur à la variable `vrai_ou_faux`.
5. **Calculer et afficher** la conjonction des deux éléments du vecteur `vrai_ou_faux`

### Correction 1.3

```
1 nom <- "Modélisation"
2 rien <- NA
3 texte <- c("Patate", "Carotte", "Oignon")
4 vrai_ou_faux <- c(TRUE, FALSE)
```



## TD1 DÉBUTER AVEC R ET RSTUDIO

5 `print(vrai_ou_faux[1] & vrai_ou_faux[2])`