

Mesures et Analyses Statistiques de Données - Probabilités

CORRECTION TP 1 - Initiation au logiciel R

Exercice 1

1. Créer le vecteur $x = (101; 102; \dots; 112)$.

```
> x<-c(seq(101,112,1)) #ou  
> x<-c(101 :112)
```

2. Créer un vecteur de longueur 12 formé de 4 fois la suite de nombres (4; 6; 3).

```
>rep(c(4,6,3),4)
```

3. Créer un vecteur composé de huit 4, de sept 6 et de cinq 3.

```
>c(rep(4,8),rep(6,7),rep(3,5)) #ou  
>rep(c(4,6,3),c(8,7,5))
```

Exercice 2

1. Saisir la variable `poids` contenant les 15 valeurs suivantes :

```
28 ; 27.5 ; 27 ; 28 ; 30.5 ; 30 ; 31 ; 29.5 ; 30 ; 31 ; 31 ; 31.5 ; 32 ; 30 ; 30.5
```

```
>poids<-c(27,27.5,...,30.5) # ou  
>poids<-scan()
```

2. Saisir la variable `poids1` contenant les 5 valeurs suivantes :

```
40 ; 39 ; 41 ; 37.5 ; 43
```

```
>poids1<-c(40,39,...,43) # ou  
>poids1<-scan()
```

3. Sans refaire de saisie, créer la variable `nouveau.poids` contenant 20 valeurs (les 5 valeurs de `poids1` répétées 2 fois et les 10 dernières valeurs de `poids`).

La question peut être comprise de deux manières différentes :

```
>nouveau.poids<-c(rep(poids1,2),poids[c(6 :15)]) # si on répète 2 fois le vecteur "poids1" #ou  
>nouveau.poids<-c(rep(poids1,c(2,2,2,2,2)),poids[c(6 :15)]) # si on répète chaque valeur de  
"poids1" 2 fois
```

4. Enregistrer dans votre répertoire de travail la variable `nouveau.poids` dans une feuille nommée « Nouveau Poids » du classeur Excel « Poids.xls ».

Veillez à bien changer le répertoire courant selon votre convenance. Vous y retrouverez le fichier xls créé.

```
>nouveau.poidsDF<-data.frame(nouveau.poids) # on doit au préalable transformer notre vecteur  
en data.frame  
>library(RODBC)  
>connexion<-odbcConnectExcel(xls.file ="poids.xls",readOnly=FALSE)  
>sqlSave(connexion, nouveau.poidsDF, tablename = "NouveauPoids")  
>odbcClose(connexion)
```

Exercice 3

1. Créer le vecteur `nom` contenant les noms de 10 personnes.

```
>nom<-c("nom1","nom2","nom3","nom4","nom15","nom6","nom7","nom8","nom9","nom10")
```

2. Créer le vecteur `age` contenant l'âge des 10 personnes précédentes (entre 20 et 60 ans). Les noms des personnes seront utilisées comme légende pour le vecteur `age`.

```
>age<-c(36,37,35,21,25,45,53,59,24,42)  
>names(age)<-nom
```

3. Créer le vecteur `poids` contenant le poids des 10 personnes (entre 50 et 100 kg) en utilisant à nouveau le nom des personnes comme légende pour ce vecteur.

```
>poids<-c(60,53,65,78,87,82,59,73,79,89)  
>names(poids)<-nom
```

4. Même chose pour le vecteur `taille` contenant la taille des 10 personnes.

```
>taille<-c(160,168,175,169,185,180,178,190,179,188)  
>names(taille)<-nom
```

5. Créer le vecteur `poids.lourds` contenant le poids des personnes de plus de 80 kg.

```
>poids.lourds<-poids[poids>80]
```

6. Créer le vecteur `taille.poids.lourds` contenant la taille des personnes de plus de 80 kg.

```
>taille.poids.lourds<-taille[poids>80]
```

7. Créer le vecteur `taille.vieux.poids.lourds` contenant la taille des personnes de plus de 80 kg et âgées de plus de 30 ans. Pour répondre à cette question, vous pourrez utiliser le connecteur logique ET dont la syntaxe est donnée dans l'aide sur opérateurs logiques accessible en tapant l'instruction `?Logic` .

```
>taille.vieux.poids.lourds<-taille[poids>80&age>30]
```