

Exercice de préparation à la séance 2

#Créer un objet Sexe

```
set.seed(13) #fixe la graine du générateur pour créer un objet identique  
Sexe<-rbinom(100,1,.52)
```

#Vérifier combien on a d'hommes et de femmes

```
table(as.factor(Sexe))
```

#regarder les données dans leur ensemble

```
View(Sexe)
```

#On se propose de recoder la variable Sexe, avec le code 1 attribué aux Hommes('M') et le code 0 attribué aux Femmes('F')

```
Sexe<-ifelse(Sexe==1,'M','F')
```

#Créer la variable Age selon les normes données avec un chiffre après la virgule

```
set.seed(13)  
Age<-round(rnorm(100, mean=25, sd=2),digits=1)
```

#Créer la variable Poids selon les normes données avec un chiffre après la virgule, attention il faudra différencier ici les hommes et les femmes

```
Poids<-ifelse (Sexe=="F",round(rnorm(100, mean=59,sd=4),digits=1),  
round(rnorm(100, mean=75, sd=5),digits=1))
```

#Créer la variable Taille selon les normes données sans chiffre après la virgule, attention il faudra différencier ici les hommes et les femmes

```
Taille<-ifelse(Sexe=="F",round(rnorm(50,mean=165,sd=3),digits=0),round(rnorm(100,  
mean=174, sd=2.5),digits=0))
```

#Créer la variable Maladie qui prendra la valeur 1 si « malade » et 0 si « non malade »

```
set.seed(14) #fixe la graine du générateur  
Maladie<-rbinom(100,1,.35)
```

#Créer une table Mydata1 qui contiendra l'ensemble des vecteurs créés

```
Mydata<-data.frame(Sexe,Age,Poids, Taille, Maladie)
```

```
Mydata
```

```
View(Mydata)
```

#edit(Mydata)

```
head(Mydata)
```

```
table(Sexe, Maladie)
table(Mydata$Sexe, Mydata$Maladie)
```

#Changer les noms des variables de Mydata

```
names(Mydata)[1]<-"SEXE"
names(Mydata)[2]<-"AGE"
names(Mydata)[3]<-"POIDS"
names(Mydata)[4]<-"TAILLE"
names(Mydata)[5]<-"MALADIE"
```

#ou vous pourriez écrire : names(Mydata)<-c("SEXE", "AGE", "POIDS", "TAILLE", "MALADIE")

```
head(Mydata)
```

```
table(Sexe, Maladie)
```

```
table(SEXE, MALADIE)
```

#Taper maintenant

```
table(Mydata$SEXE, MALADIE)
```

```
table(Mydata$SEXE, Mydata$MALADIE)
```

#pour avoir la somme des lignes

```
apply(table(Mydata$SEXE, Mydata$MALADIE), MARGIN=1, sum)
```

#pour simplifier l'écriture on aurait pu écrire:

```
Mytable<-table(Mydata$SEXE, Mydata$MALADIE)
apply(Mytable, MARGIN=1, sum)
```

#avoir la somme des colonnes

```
apply(Mytable, MARGIN=2, sum)
```

```
summary(Mydata)
```

```
summary(Mydata$POIDS)
```

```
tapply(Mydata$POIDS, Mydata$SEXE, summary)
```

#Sauvegarder dans un fichier appelé DataTD1.csv

```
write.csv2(Mydata, "DataTD1.csv")
```