

```
#-----  
#                               Solution Question #1  
#-----
```

```
#Créer un vecteur pour la longueur  
long<-1:10  
#Créer un vecteur pour la hauteur  
haut<-1:10  
#Vérifier les combinaisons  
long*haut  
#Recréer un vecteur de hauteur pour avoir les combinaisons  
haut<-rep(1:10, each=10)  
#Vérifier les combinaisons  
long*haut  
#Créer les paramètres aire et perimetre  
aire<-long*haut  
perim<-2*(long+haut)  
#Trouver quel perimetre = aire  
#Petit truc () = print()  
(solution<-which(aire==perim))  
haut[solution]  
long[solution]  
aire[solution]/haut[solution]  
#nettoyer la console de travail complètement!!!  
rm(list=ls())
```

```
#-----  
#                               Importer des données  
#-----
```

```
#choisir un dossier de travail  
setwd("C:/Documents and Settings/chroy203/Bureau/Formation R")  
#dir() #Vérifier le contenu du dossier  
#setwd("../") #Reculer de un dossier dans l'arborescence  
#getwd() #Vérifier ou je suis dans la console R  
#setwd("Formation R") avancer de un dossier dans l'arborescence  
#####  
#Fichier .txt ou .csv  
#####  
mydata <- read.table("mydata.txt", header=TRUE, sep="\t")  
#Vérifier si l'importation c'est passée correctement  
class(mydata) #Quelle est la classe de l'objet?  
head(mydata) #imprimer les premières lignes d'un dataframe  
str(mydata) #quelle est la structure de mon objet  
#View(mydata) Voir le dataframe dans une boîte  
#edit(mydata) Editer le dataframe....a vos risques  
#file.choose() Pour plus de flexibilité  
#####  
#Fichier Excel  
#####  
#Vous avez besoin du package RODBC  
library(RODBC)  
#help(package="RODBC")  
channel <- odbcConnectExcel("mydata.xls")  
#sqlTables(channel) afficher le contenu du fichier excel  
mydata1.xls <- sqlFetch(channel, "Feuil1")  
mydata2.xls <- sqlFetch(channel, "Feuil2")  
#class(mydata)  
#head(mydata)  
#str(mydata)  
#Combiner 2 fichier de données qui contiennent les mêmes colonnes  
#names(mydata1.xls) Vérifier le nom des colonnes pour un data frame  
#names(mydata2.xls)  
mydata.combined<-rbind(mydata1.xls, mydata2.xls)  
#Fermer la connexion une fois les fichiers importés  
odbcClose(channel)  
rm(channel) #Enlever un élément de la console  
#####  
#Fichier DBF
```

```
#####
library(foreign)
setwd("C:/Mes documents/Doctorat/Data/WBHS")
transects<-read.dbf("2008_segments_Chapter1.dbf")
str(transects)
head(transects)

#-----
#
#                               Exporter des données
#-----

setwd("C:/Documents and Settings/chroy203/Bureau/Formation R")
#####
#Créer un fichier de données fictives
#####
sites<-rep(1:10, each=5)
stations<-rep(1:5, time=10)
#length(Sites)      #Vérifier la longueur d'un vecteur
#length(Stations)
oiseaux<-rpois(50, lambda=5)          #Poisson distribution discrète
végétation<-rbeta(50, shape1=55, shape2=8) #Beta distribution entre 0 et 1
temperature<-rnorm(50, mean=0, sd=8)   #Distribution normale
#type<-rep(c("F", "R", "MR", "MF", "DH"), each=10) #Données catégoriques
#class(Type) #Vérifier la classe de Type
#type<-as.factor(Type)
type<-gl(5,10)
levels(Type)<-c("F", "R", "MR", "MF", "DH")
#####
#Combiner dans un objet
#####
export.file<-cbind(Sites, Stations, Type, Oiseaux, Végétation, Temperature)
head(export.file)
str(export.file)
class(export.file)
export.file<-as.data.frame(cbind(Sites, Stations, Type, Oiseaux, Végétation, Temperature))
head(export.file)
str(export.file)
export.file<-as.data.frame(cbind(Sites, Stations, Oiseaux, Végétation, Temperature))
head(export.file)
export.file$Type<-Type
#####
#Export
#####
write.table(export.file, file="export.me.txt", sep=";", row.names=F, col.names=T)

#-----
#
#                               Manipuler des données
#-----

#####
#With datasets
#####
#Importer des données
set1 <- read.table("export.me.txt", header=TRUE, sep=";")
set2 <- read.table("export.me2.txt", header=TRUE, sep=";")
#Combiner deux set de données qui partage de l'information
both.set <-merge(set1, set2, by=c("Sites","Stations"))
#Garder seulement certaines colonnes
both.set.2<- both.set[c("Sites", "Stations", "Oiseaux")]
#Changer l'ordre du data.frame
both.set.2 <- both.set.2[order(both.set.2$Sites, both.set.2$Stations),]
#Sélectionner certaines données
part.sets<-subset(both.set, both.set$Type=="F")
#subset(both.set, both.set$Type%in%c("F", "MF"))
#Operator
# == Equal
# != does not equal
# & and
```

```
# %in% is in the vector x
#####
#with vectors
#####
oiseaux
which(oiseaux >= 5)
oiseaux[which(Oiseaux >= 5)]
(oiseaux2<-as.factor(ifelse(oiseaux>=5, 2, 1)))
order(Oiseaux) #position
sort(Oiseaux) #Données
x<-c(5, 2, NA, 4, 0, 2, 1, NA, 8, 3, NA, NA, 0, 7)
is.na(x)
x[is.na(x)!=T]
```

```
#-----
#                               Exercise
#-----
data(SeedPred_wide, package = "emdbook")
head(SeedPred_wide)
data(SeedPred, package = "emdbook")
head(SeedPred)
```