

1



RÉPUBLIQUE TUNISIENNE - MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ DE CARTHAGE - ÉCOLE SUPÉRIEURE DE LA STATISTIQUE ET DE L'ANALYSE DE L'INFORMATION



TEST LOGICIEL R - NOVEMBRE 2022
Ines Abdeljaoued Tej - ines.tej@essai.ucar.tn

Année : 1ère Année du Cycle Ingénieur en Statistique et Analyse de l'Information.
Module : Logiciel R
Durée : 1h

Yahya chammami.

- ① Quelles sont les différentes structures de données en R ?
2. Comment charger un fichier .csv dans R ?
- ③ Qu'est-ce que Rmarkdown ? Quelle en est l'utilité ?
- ④ Comment installer un package dans R ?
5. Comment écrivez-vous une fonction personnalisée dans R ? Donnez un exemple.
6. Nommez quelques fonctions disponibles dans la librairie dplyr.
7. Dis-moi quelque chose à propos de ShinyR.
8. Expliquez ce que fait le code suivant :

```
help(mtcars)
cars.data <- mtcars[,c(1,3,4,5,6,7)]
samp.range <- function(x){myrange <- diff(range(x)); return(myrange) }
my.ranges <- apply(cars.data,2,samp.range)
cars.std <- sweep(cars.data,2,my.ranges,FUN='/' )
```

- ⑨ Donnez un exemple d'utilisation des fonctions rbind() et cbind() dans R :

10. Donner les sorties des fonctions select et filter du paquet dplyr :

```
library(dplyr)
head(diamonds)
diamonds %>% select(1,2,3) -> birth
diamonds %>% select(-5) -> birth
diamonds %>% filter(depth>78) -> birth
diamonds %>% filter(clarity=="VS2" & cut=="Very Good" & price>18500) -> birth
```

W

#birth

	carat <dbl>	cut <ord>	color <ord>	clarity <ord>	depth <dbl>	table <dbl>	price <int>	x <dbl>	y <dbl>	z <dbl>
A tibble: 6 × 10	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
	0.29	Premium	I	VS2	62.4	58	334	4.20	4.23	2.63
	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48

11. Quelle est l'utilité du package `stringR`. Donner quelques exemples des fonctions de `stringR` :

- Convertir les chaînes de caractères en majuscules
- Trouver le nombre de lettres

12. Comment créer un nuage de points en utilisant le paquet `ggplot2`?

13. Étant donné un vecteur de valeurs, comment le convertir en un objet de série chronologique?

```
[ ]: a <- c(1,2,3,4,5,6,7,8,9)
```

14. Quelle est la différence entre un diagramme à barres et un histogramme? Où utiliseriez-vous un diagramme à barres et où utiliseriez-vous un histogramme?

```
[ ]: library(ggplot2)
#ggplot(data = iris,aes(x=Sepal.Length)) + geom_histogram(fill="palegreen4",
#col="green")
#ggplot(data = iris,aes(x=Species)) + geom_bar(fill="palegreen4")
```

15. Quelle est la réponse de cette commande `rbind(df1, df2)`?

```
[ ]: company <- c("Ford","BMW")
models <- c(10,23)
df1 <- data.frame(company,models)

company <- c("Lamborghini","Mercedes")
models <- c(15,28)
df2 <- data.frame(company,models)
```

16. Qu'est-ce qu'un factor? Comment créer un factor en R? Initialement, nous avons un vecteur de caractères de noms de fruits, convertissons-le en un factor :

```
[ ]: fruits <- c("apple", "orange", "grape", "banana", "guava")
```

17. Écrivez une fonction qui remplace toutes les valeurs manquantes dans un vecteur par la moyenne des valeurs.

```
: a <- c(1,2,3,NA,4,5,NA,NA)
```

18. Nommez quelques fonctions qui peuvent être utilisées pour le debugging dans R ?

19. Comment vérifier la distribution d'une variable catégorielle dans R ? Montrer sur le jeu de données iris.

```
[ : head(iris)
```

A data.frame : 6 × 5

	Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fct>
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

20. Comment renommer les colonnes d'un dataframe ?

21. Comment trouver le nombre de valeurs manquantes dans un ensemble de données et les supprimer toutes ?

22. Comment extraire un mot particulier d'une chaîne de caractères ?

```
[ : hannibal <- "This is Hannibal! This is Hannibal! This is Hannibal! This is_
    _Hannibal! This is Hannibal!"
```

23. À partir de l'ensemble de données ci-dessous, extraire uniquement les valeurs où Age > 60 et Sex == "F".

```
: Age <- c(70, 35, 65, 90, 3)
Sex <- c("M", "F", "F", "M", "M")
df <- data.frame(Age, Sex)
```

24. Donnez cinq caractéristiques de R.

25. Expliquez RStudio.

26. Quels sont les avantages et les inconvénients de R ?

27. Quelle sera la réponse de l'expression all(NA==NA) ?

28. Donnez la commande permettant de supprimer un vecteur de l'espace de travail R ?

29. Quelle est la valeur de f(3) ?

```
[ ]: x <- 7
     f <- function(y) { x <- 3; x^2 + g(y) }
     g <- function(y) { y + x }
```

30. Étant donné l'ensemble de données `iris`, où appliqueriez-vous Barplot? Tracez un histogramme pour la variable `Sepal`.

```
`: #ggplot(data = iris, aes(x=Sepal.Width)) + .....
```

31. Le code suivant retournera-t-il une erreur? Expliquez la raison de votre réponse.

```
[ ]: val <- numeric()
     result <- vector("list", length(val))
     for (index in 1:length(val)) {
       result[index] <- val[index] ^ 2
     }
     result
```

32. Quelles sont les erreurs présentes dans le code ci-dessous? Corrigez ces erreurs.

```
[ ]: #iris[iris$Sepal.Length = 5, ]
     #iris[-1:5, ]
     #iris[iris$Sepal.Length <= 5]
```

33. Vous avez pour tâche de trouver les noms de tous les fichiers `.csv` dans le répertoire. Comment allez-vous effectuer cette tâche en utilisant l'opération `regex`? Ces fichiers `csv` sont stockés selon le modèle suivant.

```
[1] "birds.csv" "boston.csv" "data.csv" "grass.csv" "table.csv"
```

34. Supposons que vous ayez la chaîne suivante :

```
data_string <- "I learn Big Data for 4 hours daily with DataFlair"
```

Vous devez effectuer les opérations suivantes en utilisant diverses séquences `regex` :

```
Match the digit 4 and replace it with a dash '_'
Replace the non-digit part of the string with '_'
Replace the non-word character in string with '&'
```

35. Quelle est la différence entre une matrice et un dataframe?

36. Quelle est la différence entre `lapply` et `sapply`?

37. Quelle est la différence entre `seq` et `rep`?

38. Quel sera le résultat du programme suivant :

```
: f <- function(a, b){ a^2 }  
f(2)
```

39. Quel sera le résultat du code suivant :

```
[ ]: paste("a", "b", se = ":")
```

40. Quelle sera la sortie de la fonction suivante :

```
: recurse_fibonacci <- function(n) {  
  if(n <= 1) {  
    return(n)  
  } else {  
    return(recurse_fibonacci(n-1) + recurse_fibonacci(n-2))  
  }  
}  
#recurse_fibonacci(10)
```

41. Comment lister les données préinstallées dans R?

42. Fusionner les deux jeux de données par la variable id.

```
[ ]: id <- c(1:5)  
names <- c("Paul", "Sara", "John", "Sandra", "Helen")  
years <- c("24", "30", "45", "29", "18")  
  
people <- data.frame(id, names)  
age <- data.frame(id, years)
```

43. Ajoutez du code à la boucle for qui passe les éléments du vecteur linkedin :

- Si la valeur de l'élément du vecteur est supérieure à 16, le message **Vous êtes populaire** s'affiche.
- Si la valeur de l'élément du vecteur ne dépasse pas 5, le message **Faites un effort !** s'affiche.

```
[ ]: linkedin <- c(16, 9, 13, 5, 2, 17, 14)
```

```
# Code the for loop with conditionals  
for (li in linkedin) {
```

```
  print(li)  
}
```


44. Le vecteur pionniers a déjà été créé pour vous : `pioneers <- c("GAUSS:1777", "BAYES:1702", "PASCAL:1623", "PEARSON:1857")`

Séparer les noms de l'année de naissance.

45. Trouvez le plus petit et le plus grand nombre entre 7000 et 70000 divisible par 233.

46. Créer le vecteur d'entiers `vecX` avec les valeurs 1 à 20. Calculer le carré des 5 premiers éléments de `vecX`. Calculer la somme de `vecX`.

47. Créer le vecteur d'entiers `vecPasMultiples` contenant tous les nombres de 1 à 100 qui ne sont pas des multiples de 5.

48. Créer une matrice `A` de dimension 10 lignes x 10 colonnes telle que : `A[i,i]=2` ; `A[i,i+1]=-1` ; `A[i+1,i]=1` le reste des valeurs = 0.

49. Créer une séquence d'ADN aléatoire `S` de 1000 nucléotides. Combien de fois est présent chaque nucléotide A,C,T,G ?

50. La tâche est d'extraire un sous-ensemble de la dataframe qui contient seulement les colonnes pour le lieu et l'humidité.

```
df = data.frame(Loc=paste("Loc", 1:4),
  Temp=c(20, 21, 22, 19),
  Humid=c(70,85,60,75),
  Press=c(1006, 1005, 1007.2, 1010));
```

QCM sur la programmation R inspiré de datacamp.com

51 Complete the code to return the output

```
x <- .....( a, b ) .  
a = .....(1,2,3),  
b = .....('uno', 'dos', 'tres')  
)  
x
```

Expected Output

```
$a  
[1] 1 2 3  
$b  
[1] "uno" "dos" "tres"
```

52 Complete the code to return the output

```
x <- c(1, 2, 3, 4)  
4 ... >= ... x
```

Expected Output

```
[1] TRUE
```

53 Complete the code by (x[a] = c(1,2,3) or x\$a <- c(1, 2, 3) or x) to return the output

```
x <- list(  
b = c('uno', 'dos', 'tres')  
)
```

Expected Output

```
$b  
[1] "uno" "dos" "tres"  
$a  
[1] 1 2 3
```

54 Complete the code by (df[y] = c(6 :10) or df\$y <- 6 :10 or df) to return the output

```
df <- data.frame( x = 1 :5 )
```

Expected Output

```
x y 1 1 6 2 2 7 3 3 8 4 4 9 5 5 10
```

55 Use a suitable tidyverse operator to calculate the mean of the lengths, removing missing values
Complete the code to return the output

```
lengths <- c(7, 4, 4, 1, 10, 8)
.....(lengths)
```

Expected Output

```
[1] 5.666667
```

56 Complete the code by (df[x] = c(1,2,3) or df\$x <- 1 :3 or df) to return the output

```
df <- data.frame(
  x = c(4, 5, 6),
  y = c('one', 'two', 'three')
)  
df[x] = c(1,2,3).
```

Expected Output

```
x y
1 1 one
2 2 two
3 3 three
```

57 The food data frame, a preview of which is shown below, contains three columns that are all numeric. Calculate the variance (var) for each column, ensuring that you return a list with one list element per column.

A tibble : 3 x 3

energy protein carbohydrate

```
1 200 4.29 35.7
2 180 10.9 23.4
3 188 11.8 10.6
```

Complete the code by (var(food, var) or map(food, var)) to return the output

Expected Output

```
$energy
[1] 7408.167

$protein
[1] 13.11455

$carbohydrate
[1] 383.5999
```

58 Complete the code by (for(i in c(1 :1) or for(i in 1 :10)) to return the output

```
for(i in 1:10) { print(i) }
```

Expected Output

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


- 59 Consider the loop below, that should print the squared value of x , unless the squared value becomes greater than 100, in which case the loop should stop. Complete the code by (if ($x^2 > 100$) {print(x)} or if ($x^2 > 100$) break) to return the output

```
x <- 5
while(x < 100){
  print(x^2)
  x <- x + 1
  if (x^2 > 100) break
}
```

Expected Output

```
[1] 25
[1] 36
[1] 49
[1] 64
[1] 81
[1] 100
```

- 60 Complete the code to return the output

```
head(airquality, 5)
Ozone Solar.R Wind Temp Month Day
1 41 190 7.4 67 5 1
2 36 118 8 72 5 2
3 12 149 12.6 74 5 3
4 NA 313 11.5 62 5 4
5 28 NA 14.3 56 5 5
dplyr : ..... do (airquality, Day == 5)
```

Expected Output

```
Ozone Solar.R Wind Temp Month Day
1 NA NA 14.3 56 5 5
2 NA 220 8.6 85 6 5
3 64 175 4.6 83 7 5
4 35 NA 7.4 85 8 5
5 47 95 7.4 87 9 5
```