Data Science I

Getting Started with R Problem Set - Answers

1. Given a vector `v` with values `[1, 2, 3, 4, 5]`, what will be the output of the expression `v[v > 3]`?

```r
v = c(1, 2, 3, 4, 5)
v
```

```
[1] 1 2 3 4 5
```

```r
v[v > 3]
```

```
[1] 4 5
```

2. Create a matrix `m` of dimensions 3×3 with elements from 1 to 9. What will be the output of `m[2,]`?

```r
m = matrix(1:9, nrow = 3)
m
```

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

```r
m[2, ]
```

```
[1] 2 5 8
```

3. Given a data frame `df` with two columns: `Name` and `Age`. If Name has values `['Tom', 'Jerry', 'Mickey']` and Age has values `[25, 22, 30]`, what will be the output of the expression `df[df$Age > 24,]`?

```
df = data.frame (Name = c('Tom', 'Jerry', 'Mickey'), Age = c(25, 22, 30))
df
```

```
    Name Age
1    Tom  25
2  Jerry  22
3 Mickey  30
```

```
df[df$Age > 24,]
```

```
    Name Age
1    Tom  25
3 Mickey  30
```

4. What will be the output of the function call `seq(2, 10, by=3)`?

```
seq(2, 10, by=3)
```

```
[1] 2 5 8
```

5. Given a vector `v = c(10, 20, 10, 40, 50)`, what will be the output of `which(v == 10)`?

```
v = c(10, 20, 10, 40, 50)
v
```

```
[1] 10 20 10 40 50
```

```
which(v == 10)
```

```
[1] 1 3
```

6. Given the same vector `v`, what will be the output of `sort(v, decreasing = TRUE)`?

```
sort(v, decreasing = TRUE)
```

```
[1] 50 40 20 10 10
```

7. Using the same vector `v`, what will be the output of `order(v)`?

```
order(v)
```

```
[1] 1 3 2 4 5
```

8. What will be the result of the expression `startsWith("Hello", "He")`?

```
startsWith("Hello", "He")
```

```
[1] TRUE
```

9. Given a matrix `m` of dimensions 2×3 with elements from 1 to 6. What will be the output of the expression `m[1, 2] > m[2, 1]`?

```
m = matrix(1:6, nrow = 2)
m
```

```
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

```
m[1, 2]
```

```
[1] 3
```

```
m[2, 1]
```

```
[1] 2
```

```r
m[1, 2] > m[2, 1]
```

```
[1] TRUE
```

10. Identify the logical error in the following code:

```r
x <- c(1, 2, 3, 4, 5)
y <- x[6]
print(y)
```

```r
x <- c(1, 2, 3, 4, 5)
y <- x[6] # index out of range
print(y)
```

```
[1] NA
```

11. What will be the result of the expression `startsWith("world", "Hello")`?

```r
startsWith("world", "Hello")
```

```
[1] FALSE
```

12. Given a matrix `m` with values:

```
1 2 3
4 5 6
7 8 9
```

What will be the output of `m[2:3, 1:2]`?

```r
m = matrix (1:9, nrow = 3, byrow = TRUE)
m
```

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

```
m[2:3, 1:2]
```

```
     [,1] [,2]
[1,]    4    5
[2,]    7    8
```

13. If `v` is a vector with values `[5, 7, 9, 11, 13]`, what will be the result of `max(v) - min(v)`?

```
v = c(5, 7, 9, 11, 13)
v
```

```
[1]  5  7  9 11 13
```

```
max(v) - min(v)
```

```
[1] 8
```

14. Using the previous vector `v`, what will be the output of `v[v < 10 | v > 12]`?

```
v[v < 10 | v > 12]
```

```
[1]  5  7  9 13
```

15. Consider a data frame `df` with columns `A`, `B`, and `C`. If `A`' has values `[3, 6, 9]`, `B` has values `[5, 10, 15]`, and `C` has values `[7, 14, 21]`. What will be the output of `df[df$A == 6 & df$B == 10,]`?

```
df = data.frame (A = c(3, 6, 9), B = c(5, 10, 15), C = c(7, 14, 21))
df
```

```
  A  B  C
1 3  5  7
2 6 10 14
3 9 15 21
```

```
df[df$A == 6 & df$B == 10,]
```

```
  A  B  C
2 6 10 14
```

16. What will be the output of the expression `seq(3, 30, length.out = 10)`?

```
seq(3, 30, length.out = 10)
```

```
 [1]  3  6  9 12 15 18 21 24 27 30
```

17. Given a vector `w` with values `[45, 22, 89, 67, 34]`, what will be the output of `sort(w)[which(w > 50)]`?

```
w = c(45, 22, 89, 67, 34)
w
```

```
[1] 45 22 89 67 34
```

```
w > 50
```

```
[1] FALSE FALSE  TRUE  TRUE FALSE
```

```r
which(w > 50)
```

```
[1] 3 4
```

```r
sort(w)
```

```
[1] 22 34 45 67 89
```

```r
sort(w)[c(3, 4)]
```

```
[1] 45 67
```

```r
sort(w)[which(w > 50)]
```

```
[1] 45 67
```

18. Identify the logical error in the following code:

```r
x <- c(1, 2, 3, 4, 5)
y <- x[6]
print(y)
```

```r
x <- c(1, 2, 3, 4, 5)
y <- x[6] # index out of range
print(y)
```

```
[1] NA
```

19. If z is a vector with values `[10, 20, 30, 40, 50]`, what will be the result of `z[1] + z[5]`?

```r
z = c(10, 20, 30, 40, 50)
z
```

```
[1] 10 20 30 40 50
```

```r
z[1] + z[5]
```

```
[1] 60
```

20. Given the previous vector `z`, what will be the output of the expression `z[z > 15 & z < 45]`?

```r
z[z > 15 & z < 45]
```

```
[1] 20 30 40
```

21. Using the vector `v` with values [5, 7, 9, 11, 13], what will be the output of `order(v, decreasing = TRUE)[1]`?

```r
v = c(5, 7, 9, 11, 13)
v
```

```
[1]  5  7  9 11 13
```

```r
order(v, decreasing = TRUE)
```

```
[1] 5 4 3 2 1
```

```r
order(v, decreasing = TRUE)[1]
```

```
[1] 5
```

22. Consider a matrix `n` with values:

```
3 6 9
12 15 18
21 24 27
```

What will be the output of `n[1,] + n[3,]`?

```r
n = matrix(seq(3,27,3), nrow = 3, byrow = TRUE)
n
```

```
     [,1] [,2] [,3]
[1,]    3    6    9
[2,]   12   15   18
[3,]   21   24   27
```

```r
n[1,]
```

```
[1] 3 6 9
```

```r
n[3,]
```

```
[1] 21 24 27
```

```r
n[1,] + n[3,]
```

```
[1] 24 30 36
```

23. If `u` is a vector with values [2, 4, 8, 16, 32], what will be the result of `u[2] * u[4]`?

```r
u = c(2, 4, 8, 16, 32)
u
```

```
[1]  2  4  8 16 32
```

```r
u[2] * u[4]
```

```
[1] 64
```

24. Given a matrix `p` with dimensions 4×4 containing values from 1 to 16, what will be the output of `p[2:3, 2:3]`?

```r
p = matrix(1:16, nrow = 4)
p
```

```
     [,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16
```

```r
p[2:3, 2:3]
```

```
     [,1] [,2]
[1,]    6   10
[2,]    7   11
```

25. What will be the output of the function call `seq(1, 1.9, by=0.3)`?

```r
seq(1, 1.9, by=0.3)
```

```
[1] 1.0 1.3 1.6 1.9
```

26. Given a vector `q` with values `[40, 10, 30, 20, 50]`, what will be the output of the expression `sort(q)`?

```r
q = c(40, 10, 30, 20, 50)
q
```

```
[1] 40 10 30 20 50
```

```r
sort(q)
```

```
[1] 10 20 30 40 50
```

27. Using the same vector `q`, what will be the output of the expression `order(q)`?

```r
order(q)
```

```
[1] 2 4 3 1 5
```

28. Given a data frame `df` with a single column `Value` and values `[50, 20, 40, 10, 30]`, if you want to rearrange the rows of the data frame in ascending order based on the `Value` column, which of the following codes should you use?

a. `df[sort(df$Value), ]`
b. `df[order(df$Value), ]`
c. `sort(df$Value)`
d. `order(df$Value)`

```r
df = data.frame (Value = c(50, 20, 40, 10, 30))
df
```

```
  Value
1    50
2    20
3    40
4    10
5    30
```

```r
df[order(df$Value), ]
```

```
[1] 10 20 30 40 50
```

29. If `r` is a vector with values `[-5, -3, 0, 3, 5]`, what will be the result of `sort(r, decreasing = TRUE)`?

```r
r = c(-5, -3, 0, 3, 5)
r
```

```
[1] -5 -3  0  3  5
```

```r
sort(r, decreasing = TRUE)
```

```
[1]  5  3  0 -3 -5
```

30. Using the same vector `r`, what will be the output of `order(r)`?

```
order(r)
```

```
[1] 1 2 3 4 5
```

31. Given a vector `s` with values `["apple", "banana", "cherry"]`, what will be the result of `sort(s)`?

```
s = c("apple", "banana", "cherry")
s
```

```
[1] "apple"  "banana" "cherry"
```

```
sort(s)
```

```
[1] "apple"  "banana" "cherry"
```

32. Using the same vector `s`, what will be the output of `order(s)`?

```
order(s)
```

```
[1] 1 2 3
```

33. Consider a vector `t` with values `[5, 3, 4, 1, 2]`. If you want to get the second smallest value, which of the following codes will give the correct result?

a. `sort(t)[2]`
b. `order(t)[2]`
c. `t[order(t)[2]]`
d. `t[sort(t)[2]]`

```
t = c(5, 3, 4, 1, 2)
t
```

```
[1] 5 3 4 1 2
```

```r
sort(t)[2]
```

```
[1] 2
```

34. Given a vector `u` with values `[2.5, 2.7, 2.3, 2.6]`, what will be the output of `order(u)`?

```r
u = c(2.5, 2.7, 2.3, 2.6)
u
```

```
[1] 2.5 2.7 2.3 2.6
```

```r
order(u)
```

```
[1] 3 1 4 2
```

35. Using the same vector `u`, what will be the result of `sort(u, decreasing = TRUE)`?

```r
sort(u, decreasing = TRUE)
```

```
[1] 2.7 2.6 2.5 2.3
```

36. Identify the error in the following code:

```r
vector <- c(1, 2, 3, 4, 5)
sum(vector = vector)
```

```
[1] 15
```

```r
vector <- c(1, 2, 3, 4, 5)
sum(vector = vector)
```

```
[1] 15
```

Nothing is wrong

37. Spot the mistake in the code below:

```
df <- data.frame(Name=c("Anna", "Bob", "Charlie"), Age=c(25, 30, 35))
df$Name[2]
```

```
[1] "Bob"
```

```
df <- data.frame(Name=c("Anna", "Bob", "Charlie"), Age=c(25, 30, 35))
df$Name[2]
```

```
[1] "Bob"
```

Nothing is wrong

38. What's wrong with the following code?

```
x <- c(5, 10, 15, 20)
mean(x, trim = 2)
```

```
[1] 12.5
```

```
x <- c(5, 10, 15, 20)
mean(x, trim = 2)
```

```
[1] 12.5
```

`trim` must be between 0 and 1.

39. Identify the issue in the following function call:

```
seq(from = 1, to = 10, by = 0)
```

`by` cannot be zero

40. Spot the mistake in this code:

```
matrix <- matrix(1:12, nrow = 3)
matrix[4, 2]
```

There are only 3 rows. So, 4 is a wrong row index