1 Consider the following data frame given below

| Subject | Class | Marks |
|---|---|---|
| 1 | 1 | 56 |
| 2 | 2 | 75 |
| 3 | 1 | 48 |
| 4 | 2 | 69 |
| 5 | 1 | 84 |
| 6 | 2 | 53 |

i. Create a subset of subjects less than 4 by using subset() function and demonstrate the output.

```
# Create a sample data frame with the given marks values
data <- data.frame(
  subject = c(1, 2, 3, 4, 5, 6),
  class = c(1, 2, 1, 2, 1, 2),
  marks = c(56, 75, 48, 69, 84, 53)
)

# Display the original data frame
print("Original Data Frame:")
print(data)

# Create a subset where subject is less than 4
subset_data <- subset(data, subject < 4)

# Display the subset data frame
print("Subset Data Frame (subject < 4):")
print(subset_data)
```

ii. Create a subset where the subject column is less than 3 and the class equals to 2 by using [ ] brackets and demonstrate the output.

```
# Create a sample data frame with the given marks values
data <- data.frame(
  subject = c(1, 2, 3, 4, 5, 6),
  class = c(1, 2, 1, 2, 1, 2),
  marks = c(56, 75, 48, 69, 84, 53)
)

# Display the original data frame
print("Original Data Frame:")
print(data)
```

```
# Create a subset where subject is less than 3 and class equals to 2
subset_data <- data[data$subject < 3 & data$class == 2, ]

# Display the subset data frame
print("Subset Data Frame (subject < 3 and class == 2):")
print(subset_data)
```

2 The data analyst of Argon technology Mr. John needs to enter the salaries of 10 employees in R. The Salaries of the employees are given in the following table: **[Dec 2024, 10 marks]**

| Sr. No | Name Of Employee | Salaries |
|---|---|---|
| 1 | Vivek | 21000 |
| 2 | Karan | 55000 |
| 3 | James | 67000 |
| 4 | Soham | 50000 |
| 5 | Renu | 54000 |
| 6 | Farah | 40000 |
| 7 | Hetal | 30000 |
| 8 | Mary | 70000 |
| 9 | Ganesh | 20000 |
| 10 | Krish2 | 15000 |

i. Which R commands will Mr. John use to enter these values? Demonstrate the output.

```
# Create a data frame with the given records
employee_data <- data.frame(
 sr_number = 1:10,
 name = c("Vivek", "Karan", "James", "Soham", "Renu", "Farah", "Hetal", "Mary",
"Ganesh", "Krish")
)

print("Employee Dataset:")
print(employee_data)

salary <- c(21000, 55000, 67000, 50000, 54000, 40000, 30000, 70000, 20000, 15000)

employee_data$salary <- salary

# Display the dataset
print("Employee Dataset:")
print(employee_data)
```

ii. Now Mr. John wants to add the salaries of 5 new employees in the existing table, which commands he will use to join datasets with new values in R. Demonstrate the output.

```
# Create a data frame with the salaries of 5 new employees
new_employees <- data.frame(
  sr_number = 11:15,
  name = c("Amit", "Neha", "Rahul", "Sara", "Rohit"),
  salary = c(60000, 45000, 58000, 52000, 48000)
)

# Join the new salaries with the existing dataset
combined_data <- rbind(employee_data, new_employees)

# Display the combined dataset
print("Combined Employee Dataset:")
print(combined_data)
```

3 i. Write the script to sort the values contained in the following vector in ascending order and descending order: (23, 45, 10, 34, 89, 20, 67, 99). Demonstrate the output.

ii. Name and explain the operators used to form data subsets in R.

```
# Define the vector
vector <- c(23, 45, 10, 34, 89, 20, 67, 99)

# Sort in ascending order
ascending_order <- sort(vector)

# Sort in descending order
descending_order <- sort(vector, decreasing = TRUE)

# Display the results
ascending_order
descending_order
```

ii) Name and explain the operators used to form data subsets in R

subset() function used to filter data frames or matrices based on conditions.

Example: subset(dataframe, column_name > 10)
Above code will filter rows where the column_name is greater than 10.

V <- c(1,2,3,4,5,6)

subset(V, V<4)

Sample Output
[1] 1 2 3

[ ] brackets can also be used to create subset of the data

Name <- c("N1","N2","N3", "N4")
Marks <- c (50, 40 , 35, 20)

Df= data.frame(Name, Marks)

Df [Df$Marks < 40,]

4 Consider the following data frame given below:

| Course | Id | Class | Marks |
|---|---|---|---|
| 1 | 11 | 1 | 56 |
| 2 | 12 | 2 | 75 |
| 3 | 13 | 1 | 48 |
| 4 | 14 | 2 | 69 |
| 5 | 15 | 1 | 84 |
| 6 | 16 | 2 | 53 |

i. Create a subset of course less than 3 by using [ ] brackets and demonstrate the output.

```
# Creating the data frame with the given information
course_data <- data.frame(
 course = c(1, 2, 3, 4, 5, 6),
 id = c(11, 12, 13, 14, 15, 16),
 class = c(1, 2, 1, 2, 1, 2),
 marks = c(56, 75, 48, 69, 84, 53)
)

# Displaying the data frame
print("Course Data Frame:")
print(course_data)

# Subset using []
subset_course_less_than_3 <- course_data [ course_data$course < 3, ]

# Display the subset
print("Subset of Course less than 3 using [] brackets:")
print(subset_course_less_than_3)
```

ii. Create a subset where the course column is less than 3 or the class equals to 2 by using subset() function and demonstrate the output.

```
# Subset using subset()
subset_course_class_condition <- subset(course_data, course < 3 | class == 2)

# Display the subset
print("Subset where course < 3 or class == 2 using subset():")
print(subset_course_class_condition)
```

5 i. The following table shows the number of units of different products sold on different days:

| Product | Monday | Tuesday | Wednesday | Thursday | Friday |
|---|---|---|---|---|---|
| Bread | 12 | 3 | 5 | 11 | 9 |
| Milk | 21 | 27 | 18 | 20 | 15 |
| Cola Cans | 10 | 1 | 33 | 6 | 12 |
| Chocolate Bars | 6 | 7 | 4 | 13 | 12 |
| Detergent | 5 | 8 | 12 | 20 | 23 |

Create five sample numeric vectors from this data.

```
# Create a data frame for the given sales data
sales_data <- data.frame(
  product = c("bread", "milk", "cola cans", "chocolate bars", "detergent"),
  monday = c(12, 21, 10, 6, 5),
  tuesday = c(3, 27, 1, 7, 8),
  wednesday = c(5, 18, 33, 4, 12),
  thursday = c(11, 20, 6, 13, 20),
  friday = c(9, 15, 12, 12, 23)
)

# Display the sales data table
print("Sales Data Table:")
print(sales_data)

# Create five sample numeric vectors
sample_vector1 <- sales_data[sample(1:nrow(sales_data)), "monday"]
sample_vector2 <- sales_data[sample(1:nrow(sales_data)), "tuesday"]
sample_vector3 <- sales_data[sample(1:nrow(sales_data)), "wednesday"]
sample_vector4 <- sales_data[sample(1:nrow(sales_data)), "thursday"]
```

```
sample_vector5 <- sales_data[sample(1:nrow(sales_data)), "friday"]

# Display the sample vectors
print("Sample Numeric Vectors:")
print(sample_vector1)
print(sample_vector2)
print(sample_vector3)
print(sample_vector4)
print(sample_vector5)
```

6 **i. Create a data frame from the following 4 vectors and demonstrate the output:**
**emp_id = c(1:5 )**
**emp_name = c("Rick", "Dan", "Michelle", "Ryan", "Gary")**
**start_date = c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11", "2015-03-27")**
**salary = c(60000, 45000,75000, 84000, 20000)**

**ii. Display structure and summary of the above data frame**
**iii. Extract the emp_name and salary columns from the above data frame.**
**iv. Extract the employee details whose salary is less than or equal to 60000.**

**# Step i: Create a data frame from the given vectors**

```
emp_id <- c(1:5)
emp_name <- c("Rick", "Dan", "Michelle", "Ryan", "Gary")
start_date <- as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11", "2015-03-27"))
salary <- c(60000, 45000, 75000, 84000, 20000)

# Create the data frame
employee_data <- data.frame(emp_id, emp_name, start_date, salary)
```

**Step ii: Display structure and summary of the data frame**

```
str(employee_data)
summary(employee_data)
```

```
> # Step ii: Display structure and summary of the data frame
> str(employee_data)
'data.frame':   5 obs. of  4 variables:
 $ emp_id    : int  1 2 3 4 5
 $ emp_name  : chr  "Rick" "Dan" "Michelle" "Ryan" ...
 $ start_date: Date, format: "2012-01-01" "2013-09-23" ...
 $ salary    : num  60000 45000 75000 84000 20000
```

**# Step iii: Extract the emp_name and salary columns**

```
employee_data[ , c("emp_name", "salary")]
```

```
> # Step iii: Extract the emp_name and salary columns
> employee_data[, c("emp_name", "salary")]
  emp_name salary
1     Rick  60000
2      Dan  45000
3 Michelle  75000
4     Ryan  84000
5     Gary  20000
```

**v. Extract the employee details whose salary is less than or equal to 60000.**

# Step iv: Extract employee details whose salary is less than or equal to 60000

employee_data[ employee_data$salary <= 60000, ]

```
> # Step iv: Extract employee details whose salary is less than or equal to 60000
> employee_data[employee_data$salary <= 60000, ]
  emp_id emp_name start_date salary
1      1     Rick 2012-01-01  60000
2      2      Dan 2013-09-23  45000
5      5     Gary 2015-03-27  20000
```

7 Suppose you have two datasets A and B,
Dataset A has the following data 6 7 8 9
Dataset B has the following data 1 2 4 5

```
# Dataset A
A <- c(6, 7, 8, 9)

# Dataset B
B <- c(1, 2, 4, 5)

# Combine A and B
C <- c(A, B)
print(C)
```

Output:

[1] 6 7 8 9 1 2 4 5

To combine two datasets (vectors, matrices, or data frames) into one in R, you can use functions like `c()` for vectors or `rbind()` and `cbind()` for matrices and data frames.

8

**Consider the following data frame given below**:

| Course | Id | Class | Marks |
|---|---|---|---|
| 1 | 11 | 1 | 56 |
| 2 | 12 | 2 | 75 |
| 3 | 13 | 1 | 48 |
| 4 | 14 | 2 | 69 |
| 5 | 15 | 1 | 84 |
| 6 | 16 | 2 | 53 |

i. Create a subset of course less than 5 by using [ ] brackets and demonstrate the output.

```
# Creating the data frame with the given information
course_data <- data.frame(
 course = c(1, 2, 3, 4, 5, 6),
 id = c(11, 12, 13, 14, 15, 16),
 class = c(1, 2, 1, 2, 1, 2),
 marks = c(56, 75, 48, 69, 84, 53)
)

 # Displaying the data frame

print("Course Data Frame:")
print(course_data)

# Subset using []
subset_course_less_than_5 <- course_data[course_data$course < 5, ]

# Display the subset
print("Subset of Course less than 5 using [] brackets:")
print(subset_course_less_than_5)
```

ii. Create a subset where the course column is less than 4 or the class equals to 1 by using subset() function and demonstrate the output.

```
 # Subset using subset()
 subset_course_class_condition <- subset(course_data, course < 4 | class == 1)

# Display the subset
print("Subset where course < 4 or class == 1 using subset():")
print(subset_course_class_condition)
```

i. Write a script to create a dataset named data1 in R containing following text.

Text: 2, 3, 4, 5, 6.7, 7, 8.1, 9

Solution

Script to create dataset named data1

```
# Create the dataset as text
data1 <- c("2", "3", "4", "5", "6.7", "7", "8.1", "9")

# View the dataset
print(data1)
```

output

```
[1] "2"   "3"   "4"   "5"   "6.7" "7"   "8.1" "9"
```

If numeric data is passed, the output will look like

```
data1 = c(2, 3, 4, 5, 6.7, 7, 8.1, 9)
data1
```

```
[1] 2.0 3.0 4.0 5.0 6.7 7.0 8.1 9.0
```

9 Write a R Script to create a Employee vector for 10 employee names. Create a salary vector to represent salary of 10 Employees. Create a data frame EMP from these two vectors.

10. Write 4 commands to demonstrate Hadoop file system operation.