1. Create & load the given dataset in CSV format and answer the following questions.

Order_ID	Product	Quantity	Price	Date	Category
1.	Laptop	2	800	05-01-2022	Electronics
2.	Smartphone	1	600	10-01-2022	Electronics
3.	Headphones	3	50	15-01-2022	Electronics
4.	T-shirt	5	20	20-01-2022	Apparel
5.	Jeans	2	50	01-02-2022	Apparel
6.	Sneakers	1	80	10-02-2022	Apparel
7.	Tablet	1	300	25-02-2022	Electronics
8.	Backpack	2	40	20-02-2022	Accessories
9.	Sunglasses	1	25	05-03-2022	Accessories
10.	Watch	1	100	10-03-2022	Accessories

a) Determine the dimensions of the dataset.
b) Find the most expensive product sold.
c) Identify products sold in the Electronics category after January 15, 2022.
d) Calculate the total revenue generated in each month.
e) Visualize Total Sales by category using a bar chart.

```
data<- read.csv("D://data.csv")

a)
dimensions<- dim(data)
print(dimensions)

b)
most_expensive<- data[which.max(data$Price), "Product"]
print(most_expensive)

c)
afterjan15 <- data[data$Category == "Electronics" & as.Date(data$Date)
>as.Date("15-01-2022"), "Product"]
print(afterjan15)

d)
# Convert Date column to Date format
data$Date<- as.Date(data$Date, format = "%d-%m-%Y")

# Extract the month and year from the Date column
data$Month<- format(data$Date, "%Y-%m")

# Calculate total revenue for each month
total_revenue<- tapply(data$Quantity * data$Price, data$Month, sum)

# Print the result
print(total_revenue)

e)
# Calculate total sales by category using tapply
total_sales<- tapply(data$Quantity * data$Price, data$Category, sum)

# Create a bar plot
barplot(total_sales,
main = "Total Sales by Category",
xlab = "Category",
ylab = "Total Sales",
col = "skyblue")
```

2. Create & load the given dataset in CSV format and answer the following questions.

Order_ID	Product	Quantity	Price ($)	Date	Category
1.	Apples	3	5	05-01-2022	Fruits
2.	Bread	2	3	10-01-2022	Bakery
3.	Milk	1	2	15-01-2022	Dairy
4.	Chicken	2	8	20-01-2022	Meat
5.	Pasta	1	4	01-02-2022	Grains
6.	Spinach	3	2	10-02-2022	Vegetables
7.	Yogurt	2	3	15-02-2022	Dairy
8.	Orange Juice	1	4	20-02-2022	Beverages
9.	Eggs	1	3	05-03-2022	Dairy
10.	Potato Chips	2	2	10-03-2022	Snacks

a) Which product had the least quantity sold?
b) On which date was the highest revenue generated?
c) What is the average quantity of products sold in theDairy category?
d) How many different categories of products are there?
e) Summarize sales by category and visualize with a bar chart.

```
data<- read_csv("your_dataset.csv")
```

```
data<- read.csv("D://data.csv")
```

a)

```
# Find the minimum quantity
min_quantity<- min(data$Quantity)

# Find products with the least quantity sold
products_least_quantity<- data$Product[data$Quantity == min_quantity]

# Print the result
print(products_least_quantity)
```

b)

```
highest_revenue_date<- data$Date[which.max(data$Quantity * data$Price)]
print(highest_revenue_date)
```

c)

```
average_dairy_qty<- mean(data$Quantity[data$Category == "Dairy"])
print(average_dairy_qty)
```

d)

```
num_categories<- length(unique(data$Category))
print(num_categories)
```

e)

```
# Aggregate sales data by category
sales_by_category<- aggregate(Quantity ~ Category, data, sum)

# Create bar chart
barplot(sales_by_category$Quantity, names.arg = sales_by_category$Category,
main = "Sales by Category",
xlab = "Category",
ylab = "Total Sales")
```

3. Create & load the given dataset in CSV format and answer the following questions.

Menu_Item	Category	Price ($)	Calories
Burger	Main Dish	12	600
Salad	Appetizer	8	350
Pizza	Main Dish	15	800
Pasta	Main Dish	10	700
Soup	Appetizer	6	200
Sandwich	Main Dish	9	550
Fries	Side Dish	5	400
Smoothie	Beverage	7	250
Nachos	Appetizer	11	900
Ice Cream	Dessert	4	300

a) Determine the total number of menu items in each category.
b) Find the menu item with the highest calorie count, and what are its details?
c) Identify the category with the highest average calorie count.
d) Visualize Total Sales by category using a pie chart.
e) Calculate the total revenue generated by each category.

a)

```
menu_items_per_category<- table(data$Category)
print(menu_items_per_category)
```

b)

```
highest_calorie_item<- data[which.max(data$Calories), ]
print(highest_calorie_item)
```

c)

```
average_calories_by_category<- tapply(data$Calories, data$Category, mean)
highest_avg_calorie_category<- names(which.max(average_calories_by_category))
print(highest_avg_calorie_category)
```

d)

```
sales_by_category<- tapply(data$Price, data$Category, sum)
pie(sales_by_category, labels = names(sales_by_category), main = "Total Sales by Category")
```

e)

```
revenue_by_category<- tapply(data$Price, data$Category, sum)
print(revenue_by_category)
```

4. Create a data frame using the given dataset and answer the following questions.

S.No	NAME	REG_NO	SUBJECT1	SUBJECT2
1	Tarun	2000	34	86
2	Karthik	2001	67	72
3	Raj	2002	75	77
4	Vishnu	2003	83	98
5	Shankar	2004	84	100

For the given data, write code using R to

a) Find the total score details of all the students.

b) Find the maximum score obtained by each student among the 2 subjects.

c) Find the average mark scored by Tarun and Raj.

d) What is the difference between the highest and lowest total scores obtained by the students?

e) Display the table details as a bar chart with the titleScore Details.

```
# Create the dataframe for the given dataset
data<- data.frame(
  S.NO = 1:5,
  NAME = c("TARUN", "KARTHIK", "RAJ", "VISHNU", "SHANKAR"),
  REG_NO = 2000:2004,
  SUBJECT1 = c(34, 67, 75, 83, 84),
  SUBJECT2 = c(86, 72, 77, 98, 100)
)
```

a)

```
data$total_score<- rowSums(data[, c("SUBJECT1", "SUBJECT2")])
print(data)
```

b)

```
data$max_score<- apply(data[, c("SUBJECT1", "SUBJECT2")], 1, max)
print(data[, c("NAME", "max_score")])
```

c)

```
# Subset data for Tarun and Raj
tarun_data <- data[data$NAME == "TARUN", ]
raj_data <- data[data$NAME == "RAJ", ]

# Calculate average marks for Tarun and Raj
tarun_average <- mean(c(tarun_data$SUBJECT1, tarun_data$SUBJECT2))
raj_average <- mean(c(raj_data$SUBJECT1, raj_data$SUBJECT2))

# Print the results
cat("Average marks scored by Tarun:", tarun_average, "\n")
cat("Average marks scored by Raj:", raj_average, "\n")
```

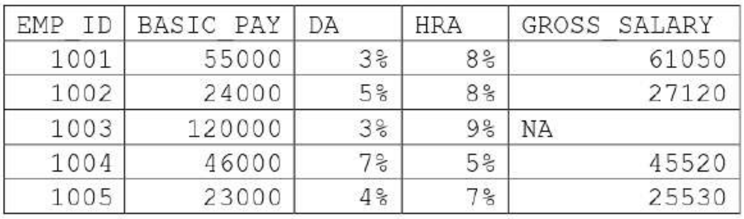
d)

```
score_range<- diff(range(data$total_score))
print(score_range)
```

e)

```
barplot(data$total_score, names.arg = data$NAME,
xlab = "Students", ylab = "Total Score",
main = "Score Details")
```

5. Create a data frame for the given dataset answer the following questions.



EMP ID	BASIC PAY	DA	HRA	GROSS_SALARY
1001	55000	3%	8%	61050
1002	24000	5%	8%	27120
1003	120000	3%	9%	NA
1004	46000	7%	5%	45520
1005	23000	4%	7%	25530

a) Find the Range of Gross Salary.
b) Find the average of the basic pay of all the employees without using the mean() function.
c) Display the EMP_ID whose gross salary is not available.
d) Find the total Dearness Allowance (DA) paid to all employees.
e) Display the table details as a pie chart with the BASIC_PAY and use rainbow() function in the color attribute.

```
# Create the dataframe for the given dataset
data<- data.frame(
  EMP_ID = c(1001, 1002, 1003, 1004, 1005),
  BASIC_PAY = c(55000, 24000, 120000, 46000, 23000),
  DA = c("3%", "5%", "3%", "7%", "4%"),
  HRA = c("8%", "8%", "9%", "5%", "7%"),
  GROSS_SALARY = c(61050, 27120, NA, 45520, 25530)
)
```

```
 a) gross_salary_range<- range(data$GROSS_SALARY, na.rm = TRUE)
print(gross_salary_range)
```

```
b) average_basic_pay<- sum(data$BASIC_PAY) / length(data$BASIC_PAY)
print(average_basic_pay)
```

```
c) emp_id_missing_salary<- data$EMP_ID[is.na(data$GROSS_SALARY)]
print(emp_id_missing_salary)
```

```
d)
 # Convert DA to numeric by removing the '%' sign and converting to percentage
data$DA<- as.numeric(gsub("%", "", data$DA))
total_da_paid<- sum(data$BASIC_PAY * data$DA / 100)
print(total_da_paid)
```

```
 e)
# Plotting pie chart
pie(data$BASIC_PAY, labels = data$EMP_ID, col = rainbow(length(data$BASIC_PAY)))
```

6. Web scrape the Covid19 pandemic dataset from the given url and save it as CSV format.
URL : https://en.wikipedia.org/wiki/Template:COVID-19_pandemic_data

```
install.packages("rvest")
library(rvest)
url <- "https://en.wikipedia.org/wiki/Template:COVID-19_pandemic_data"
page <- read_html(url)
covid19_data <- html_element(page, "table.sortable") %>% html_table()
head(covid19_data)
covid19_data <- covid19_data[c(2,3,4)] # select variables
covid19_data <- covid19_data[-nrow(covid19_data), ] # remove last row
write.csv(data, "D:\\covid_data.csv")
```

7. Web scrape the Books published per year per country dataset from the given url and save it as CSV format.
URL : https://en.wikipedia.org/wiki/Books_published_per_country_per_year

```
install.packages("rvest")
library(rvest)
url <- "https://en.wikipedia.org/wiki/Books_published_per_country_per_year"
page <- read_html(url)
data <- html_element(page, "table.sortable") %>% html_table()
head(data)
data <- data[c(2,3,4)] # select variables
data <- data[-nrow(data), ] # remove last row
write.csv(data, "D:\\data.csv")
```

8)From the dataset women, answer the following questions.
1. What is the average height of the women in the dataset?
2. What is the median weight of the women in the dataset?
3. What is the range of weights among the women in the dataset?
4. What is the tallest height recorded in the dataset?
5. Create a scatter plot showing the relationship between height and weight for the women in the dataset.

1.
```
average_height<- mean(women$height)
print(average_height)
```

2.
```
median_weight<- median(women$weight)
print(median_weight)
```

3.
```
weight_range<- range(women$weight)
print(weight_range)
```

4.
```
tallest_height<- max(women$height)
print(tallest_height)
```

5.
```
plot(women$height, women$weight,
xlab = "Height", ylab = "Weight",
main = "Relationship between Height and Weight",
col = "blue", pch = 16)
```

----

9)From the dataset chickwts, answer the following questions.

1. What is the average weight of the chicks for each type of feed?
2. Which type of feed has the highest median weight for the chicks?
3. How many chicks were fed with the feed type "horsebean"?
4. What is the range of weights among the chicks fed with the feed type "soybean"?
5. Create a bar plot showing the average weight of the chicks for each type of feed.

```
# Load the chickwts dataset
data(chickwts)
```

1.

```
avg_weight_by_feed<- tapply(chickwts$weight, chickwts$feed, mean)
print(avg_weight_by_feed)
```

2.

```
median_weight_by_feed<- tapply(chickwts$weight, chickwts$feed, median)
max_median_feed<- names(which.max(median_weight_by_feed))
print(max_median_feed)
```

3.

```
num_horsebean_chicks<- sum(chickwts$feed == "horsebean")
print(num_horsebean_chicks)
```

4.

```
range_soybean_weights<- range(chickwts$weight[chickwts$feed == "soybean"])
print(range_soybean_weights)
```

5.

```
barplot(avg_weight_by_feed,
xlab = "Feed Type", ylab = "Average Weight",
main = "Average Weight of Chicks by Feed Type")
```