```
# Example code to install packages
install.packages("data.table")
install.packages("ggmosaic")

# Load required libraries
library(data.table)
library(ggplot2)
library(ggmosaic)
library(readr)

# Assign the data files to data.tables
filePath <- "C:\\Users\\Bilal\\Documents\\"  # Fill in the path to your working directory
transactionData <- fread(paste0(filePath,"QVI_transaction_data.csv"))
customerData <- fread(paste0(filePath,"QVI_purchase_behaviour.csv"))

# Summarize transaction data
str(transactionData)



transactionData$DATE <- as.Date(transactionData$DATE, origin = "1899-12-30")


transactionData[, .N, PROD_NAME]


productWords <- data.table(unlist(strsplit(unique(transactionData[, PROD_NAME]), " ")))


setnames(productWords, 'words')


summary(transactionData)
transactionData[PROD_QTY == 200, ]
```

```r
transactionData[LYLTY_CARD_NBR == 226000, ]




#### Filter out the customer based on the loyalty card number
transactionData <- transactionData[LYLTY_CARD_NBR != 226000, ]

#### Re-examine transaction data
summary(transactionData)


#### Count the number of transactions by date
transactionData[, .N, by = DATE]


#### Count the number of transactions by date
transactionData[, .N, by = DATE]

#### Create a sequence of dates and join this the count of transactions by date
allDates <- data.table(seq(as.Date("2018/07/01"), as.Date("2019/06/30"), by = "day"))


setnames(allDates, "DATE")
transactions_by_day <- merge(allDates, transactionData[, .N, by = DATE], all.x = TRUE)


theme_set(theme_bw())
theme_update(plot.title = element_text(hjust = 0.5))
```

```
ggplot(transactions_by_day, aes(x = DATE, y = N)) +
  geom_line() +
  labs(x = "Day", y = "Number of transactions", title = "Transactions over
       time") +
  scale_x_date(breaks = "1 month") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

```
# Ensure the transactions_by_day data frame has a 'DATE' column of Date type and an 'N' column for the number of transactions.

# Step 1: Create a sequence of dates
all_dates <- seq(min(transactions_by_day$DATE), max(transactions_by_day$DATE), by="day")

# Step 2: Create a data frame with this sequence
all_dates_df <- data.frame(DATE = all_dates)

# Step 3: Merge to ensure all dates are present
complete_transactions <- merge(all_dates_df, transactions_by_day, by = "DATE", all.x = TRUE)

# Step 4: Replace NA in 'N' with 0 to indicate no transactions
complete_transactions$N[is.na(complete_transactions$N)] <- 0

# Step 5: Plotting
library(ggplot2)
ggplot(complete_transactions, aes(x = DATE, y = N)) +
  geom_line() +
  labs(x = "Day", y = "Number of transactions", title = "Transactions over time") +
```

```r
  scale_x_date(date_breaks = "1 month", date_labels = "%b %Y") +

  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))



#### Filter to December and look at individual days

ggplot(transactions_by_day[month(DATE) == 12, ], aes(x = DATE, y = N)) + geom_line() + labs(x = "Day", y = "Number of transactions", title = "Transactions over time") + scale_x_date(breaks = "1 day") + theme(axis.text.x = element_text(angle = 90, vjust = 0.5))



library(ggplot2)

library(dplyr)

library(lubridate)


# Assuming transactions_by_day is already filtered for December and contains 'DATE' and 'N'


# Generate a complete sequence of dates for December

december_dates <- seq(as.Date("2022-12-01"), as.Date("2022-12-31"), by="day")


# Create a data frame with this sequence

december_df <- data.frame(DATE = december_dates)


# Ensure transactions_by_day has a Date column in Date format

transactions_by_day$DATE <- as.Date(transactions_by_day$DATE)


# Merge to ensure all December dates are present

complete_december_transactions <- merge(december_df, transactions_by_day, by = "DATE", all.x = TRUE)


# Replace NA in 'N' with 0 to indicate no transactions

complete_december_transactions$N[is.na(complete_december_transactions$N)] <- 0
```

```
# Plotting, ensuring daily breaks
ggplot(complete_december_transactions, aes(x = DATE, y = N)) +
  geom_line() +
  labs(x = "Day", y = "Number of transactions", title = "Transactions over time in December") +
  scale_x_date(date_breaks = "1 day", date_labels = "%Y-%m-%d") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))

transactionData[,PACK_SIZE:= parse_number(PROD_NAME)]
transactionData
hist(transactionData[,PACK_SIZE])

transactionData[,.N,PACK_SIZE][order(PACK_SIZE)]

transactionData[,BRAND:= toupper(substr(PROD_NAME,1, regexpr(pattern=' ', PROD_NAME)-1))]
transactionData[,.N,by=BRAND][order(-N)]

#### Clean brand names
transactionData[BRAND == "RED", BRAND := "RRD"]
transactionData[BRAND == "SNBTS", BRAND := "SUNBITES"]
transactionData[BRAND == "INFZNS", BRAND := "INFUZIONS"]
transactionData[BRAND == "WW", BRAND := "WOOLWORTHS"]
transactionData[BRAND == "SMITH", BRAND := "SMITHS"]
transactionData[BRAND == "NCC", BRAND := "NATURAL"]
transactionData[BRAND == "DORITO", BRAND := "DORITOS"]
transactionData[BRAND == "GRAIN", BRAND := "GRNWVES"]

## check again :
transactionData[, .N, by = BRAND][order(BRAND)]

## examing customer data
str(customerData)
```

```
summary(customerData)


#### Examining the values of lifestage and premium_customer

customerData[, .N, by = LIFESTAGE][order(-N)]

customerData[, .N, by = PREMIUM_CUSTOMER][order(-N)]


#### Merge transaction data to customer data

data <- merge(transactionData, customerData, all.x = TRUE)


data[is.null(LIFESTAGE), .N]

data[is.null(PREMIUM_CUSTOMER), .N]

fwrite(data, paste0(filePath,"QVI_data.csv"))

####Totalsales by LIFESTAGEandPREMIUM_CUSTOMER

sales<-data[,.(SALES= sum(TOT_SALES)),.(LIFESTAGE,PREMIUM_CUSTOMER)]


####Create plot

p<-ggplot(data=sales) + geom_mosaic(aes(weight=SALES,x=
product(PREMIUM_CUSTOMER,LIFESTAGE), fill=PREMIUM_CUSTOMER))+ labs(x= "Lifestage",y=
"Premiumcustomerflag",title="Proportionof sales") + theme(axis.text.x=
element_text(angle=90,vjust=0.5))


####Plot andlabel withproportionof sales

p + geom_text(data= ggplot_build(p)$data[[1]], aes(x= (xmin + xmax)/2,y= (ymin + ymax)/2,label=
as.character(paste(round(.wt/sum(.wt),3)*100, '%'))))
```