Market Basket Analysis

Data processing

Load the data

Load the data downloaded from http://archive.ics.uci.edu/ml/datasets/online+retail

```
online_retail_data <- read_excel('./data/online-retail.xlsx')
online_retail_data <- online_retail_data[complete.cases(online_retail_data), ]
str(online_retail_data)
```

```
## tibble [406,829 x 8] (S3: tbl_df/tbl/data.frame)
##  $ InvoiceNo  : chr [1:406829] "536365" "536365" "536365" "536365" ...
##  $ StockCode  : chr [1:406829] "85123A" "71053" "84406B" "84029G" ...
##  $ Description: chr [1:406829] "WHITE HANGING HEART T-LIGHT HOLDER" "WHITE METAL LANTERN" "CREAM CUPI
##  $ Quantity   : num [1:406829] 6 6 8 6 6 2 6 6 6 32 ...
##  $ InvoiceDate: POSIXct[1:406829], format: "2010-12-01 08:26:00" "2010-12-01 08:26:00" ...
##  $ UnitPrice  : num [1:406829] 2.55 3.39 2.75 3.39 3.39 7.65 4.25 1.85 1.85 1.69 ...
##  $ CustomerID : num [1:406829] 17850 17850 17850 17850 17850 ...
##  $ Country    : chr [1:406829] "United Kingdom" "United Kingdom" "United Kingdom" "United Kingdom" .
```

Data cleaning:

We have negative quantities, we can remove them:

```
online_retail_data <- filter(online_retail_data, Quantity > 0)
```

We have prices as 0, we can remove them as well:

```
online_retail_data <- filter(online_retail_data, UnitPrice > 0)
```

Based on the below code we can safely say that we don't have any NA values in the dataset:

```
which(is.na(online_retail_data))
```

```
## integer(0)
```

Convert the data types and formats of the fields:

Here the columns CustomerID, Country, StockCode can be changed to factor.

```
online_retail_data$Country <- as.factor(online_retail_data$Country)
online_retail_data$CustomerID <- as.factor(online_retail_data$CustomerID)
online_retail_data$StockCode <- as.factor(online_retail_data$StockCode)
```

Similarly we can construct the PurchaseTime and PurchaseTimeInHours of the goods from invoice date and convert the InvoiceDate to Date datatype.

```
online_retail_data$PurchaseTime <- format(online_retail_data$InvoiceDate,"%H:%M:%S")
online_retail_data$PurchaseTimeInHours <- format(online_retail_data$InvoiceDate,"%H")
online_retail_data$InvoiceDate <- as.Date(online_retail_data$InvoiceDate)
```

We can construct total cose of the oder

```
online_retail_data$TotalCost <- online_retail_data$Quantity * online_retail_data$UnitPrice
```

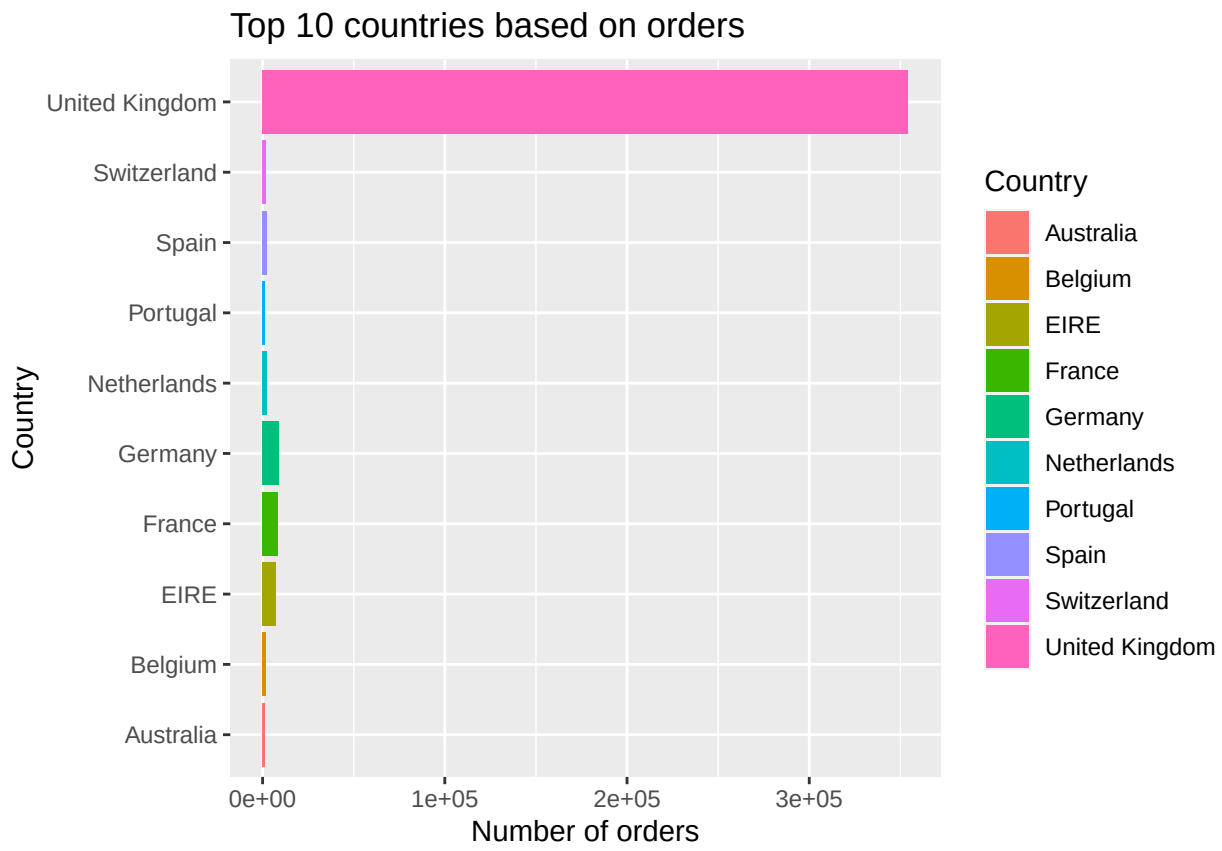
Save the data into csv format:

```
write_csv(online_retail_data, file="./data/online-retail.csv")
```

Exploratory Analysis:

Top 10 countries based on orders

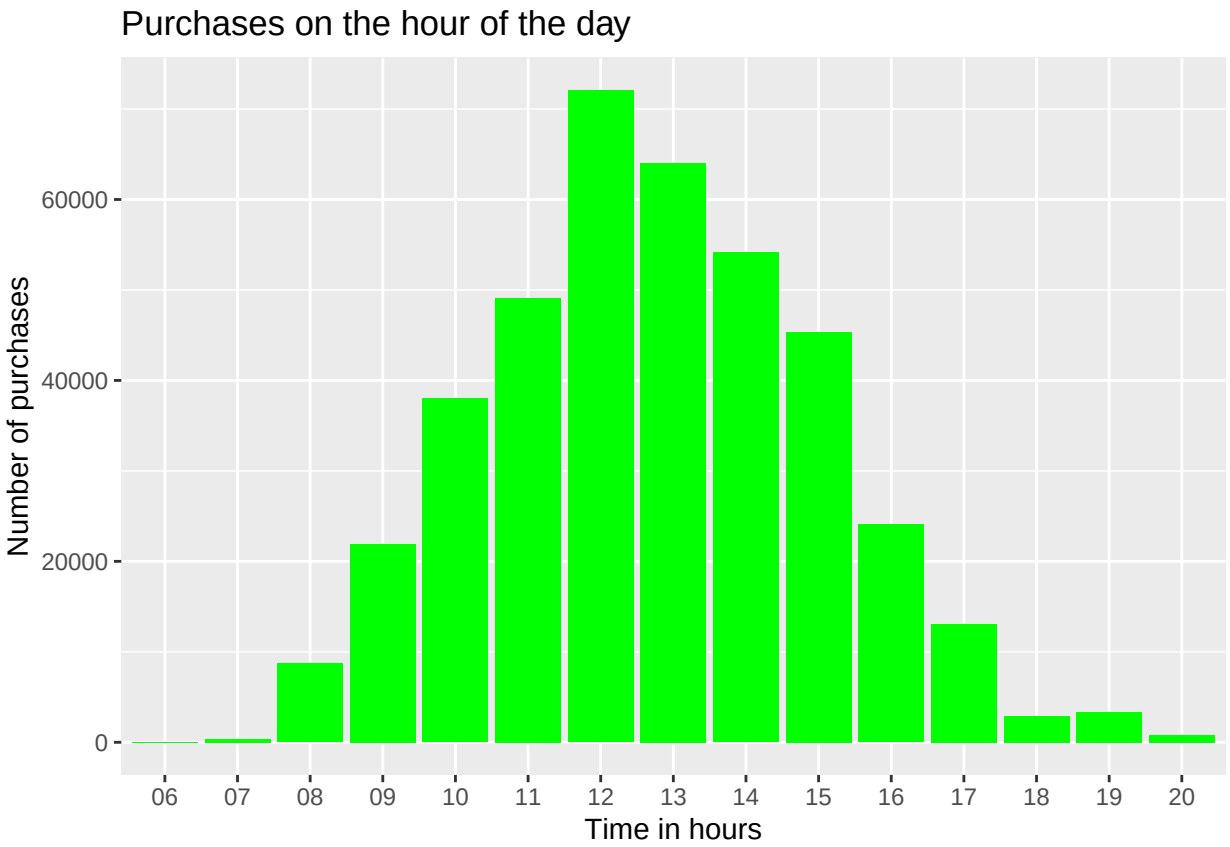
```
bar_graph_data <- online_retail_data %>%
  group_by(Country) %>%
  summarize(count = n()) %>%
  arrange(desc(count))
bar_graph_data <- head(bar_graph_data, n=10)

ggplot(bar_graph_data, aes(x=count, y=Country, fill=Country)) +
  geom_bar(stat = "identity") +
  ggtitle("Top 10 countries based on orders") +
  xlab("Number of orders")
```



Purchases on the hour of the day

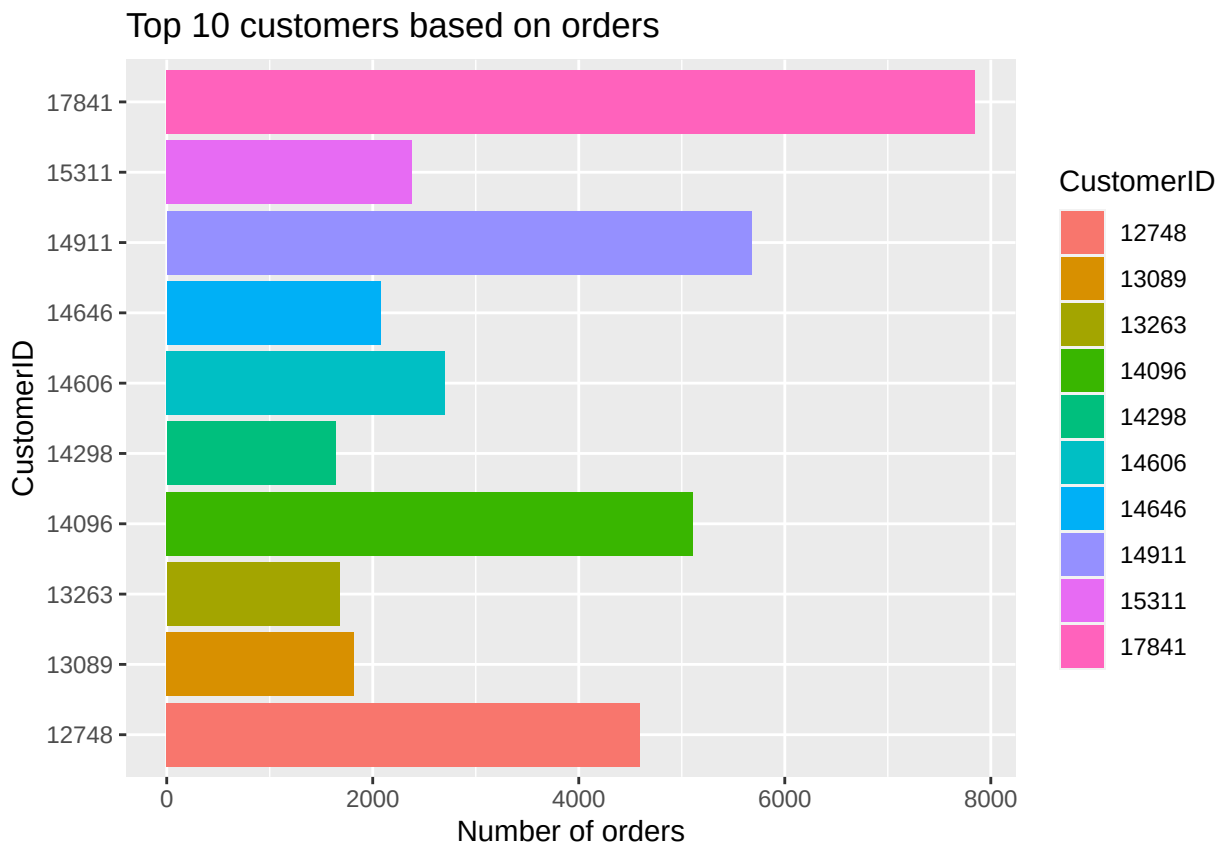
```
ggplot(online_retail_data, aes(x=PurchaseTimeInHours)) +
geom_histogram(stat="count", fill="green") +
ggtitle("Purchases on the hour of the day") +
xlab("Time in hours") +
ylab("Number of purchases")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



Top 10 customers based on number of orders placed

```
bar_graph_data <- online_retail_data %>%
  group_by(CustomerID) %>%
  summarize(Count = n()) %>%
  arrange(desc(Count))
bar_graph_data <- head(bar_graph_data, n=10)

ggplot(bar_graph_data, aes(x=Count, y=CustomerID, fill=CustomerID)) +
  geom_bar(stat = "identity") +
  ggtitle("Top 10 customers based on orders") +
  xlab("Number of orders")
```

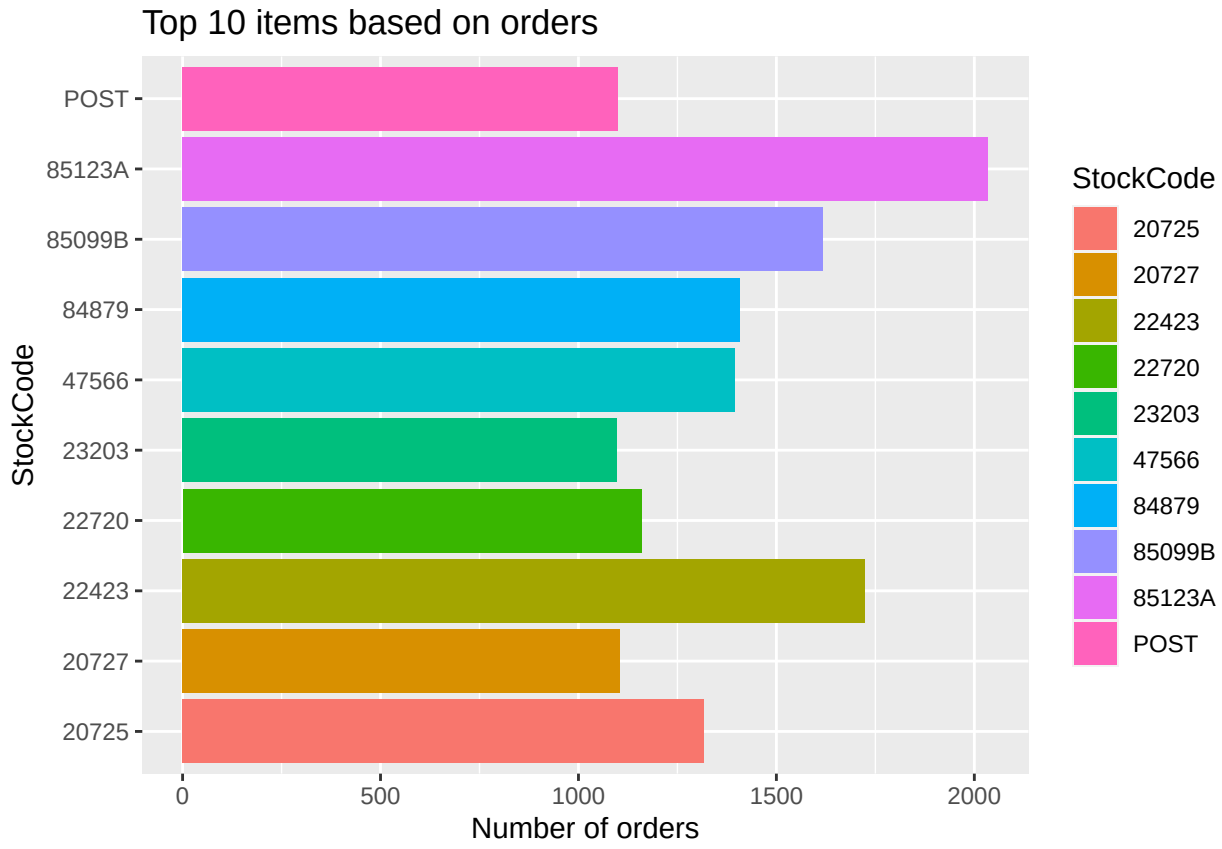


Top 10 items sold

```
bar_graph_data <- online_retail_data %>%
  group_by(StockCode) %>%
  summarize(Count = n()) %>%
  arrange(desc(Count))
bar_graph_data <- head(bar_graph_data, n=10)
bar_graph_data
```

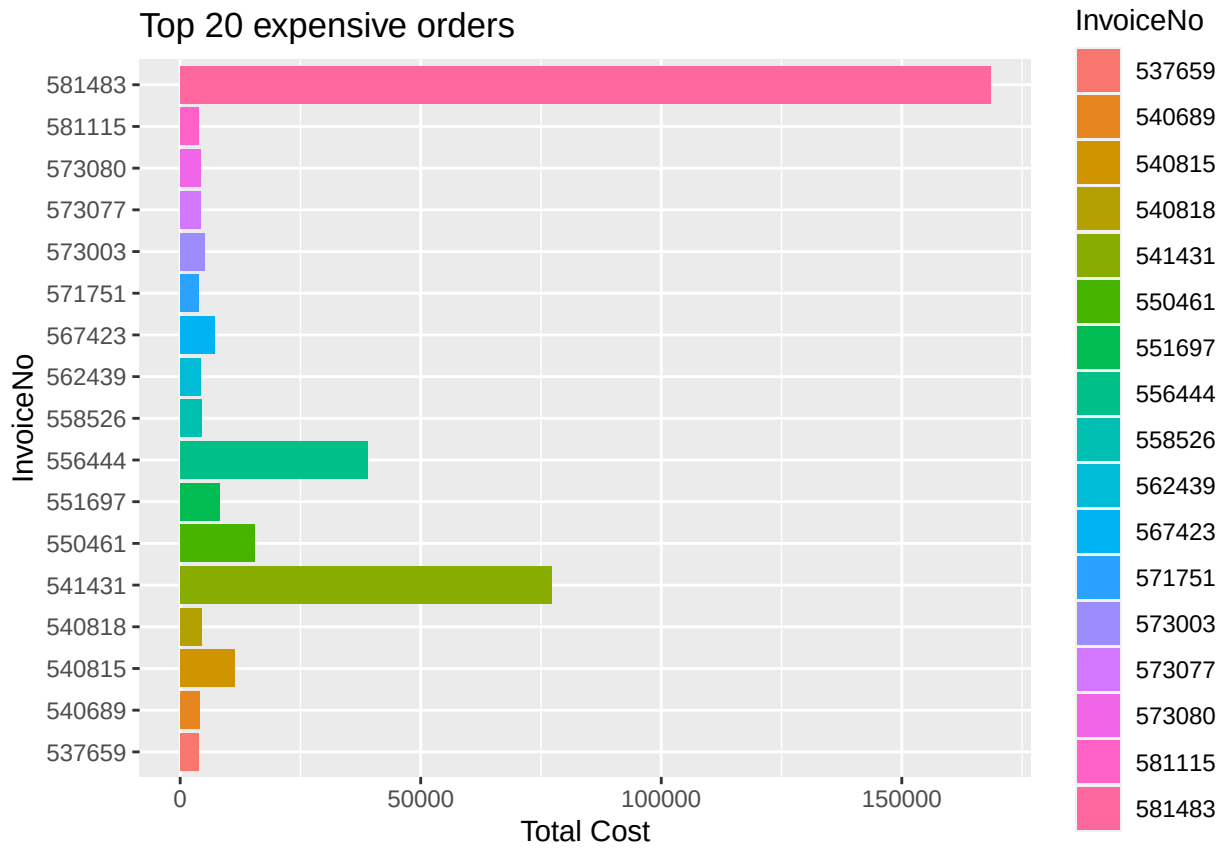
```
## # A tibble: 10 x 2
##    StockCode Count
##    <fct>     <int>
##  1 85123A     2035
##  2 22423      1723
##  3 85099B     1618
##  4 84879      1408
##  5 47566      1396
##  6 20725      1317
##  7 22720      1159
##  8 20727      1105
##  9 POST       1099
## 10 23203      1098
```

```
ggplot(bar_graph_data, aes(x=Count, y=StockCode, fill=StockCode)) +
  geom_bar(stat = "identity") +
  ggtitle("Top 10 items based on orders") +
  xlab("Number of orders")
```



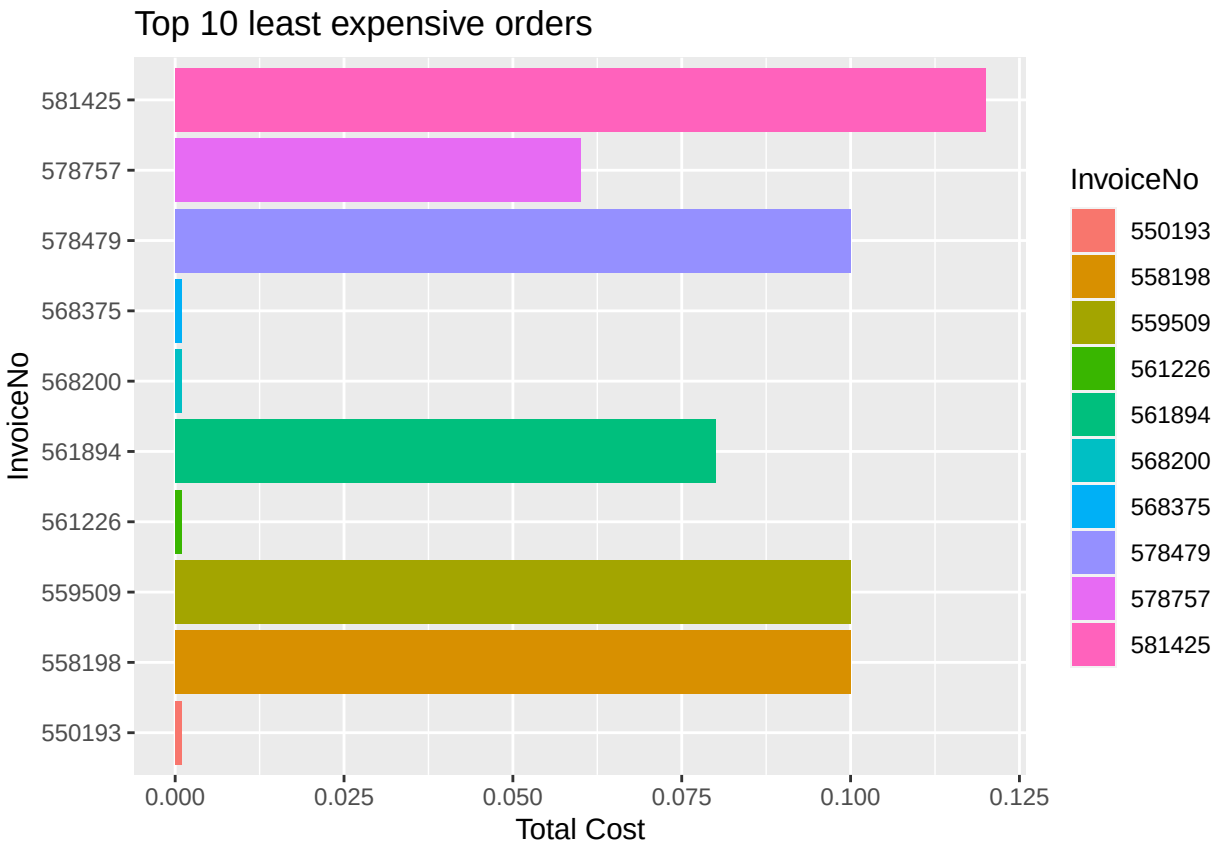
Top 20 expensive orders

```
bar_graph_data <- online_retail_data %>%
  arrange(desc(TotalCost))
bar_graph_data <- head(bar_graph_data, n=20)

ggplot(bar_graph_data, aes(x=TotalCost, y=InvoiceNo, fill=InvoiceNo)) +
  geom_bar(stat = "identity") +
  ggtitle("Top 20 expensive orders") +
  xlab("Total Cost")
```



Top 10 least expensive items

```
bar_graph_data <- online_retail_data %>%
  arrange(desc(TotalCost))
bar_graph_data <- tail(bar_graph_data, n=10)

ggplot(bar_graph_data, aes(x=TotalCost, y=InvoiceNo, fill=InvoiceNo)) +
  geom_bar(stat = "identity") +
  ggtitle("Top 10 least expensive orders") +
  xlab("Total Cost")
```


Top 10 least expensive orders
InvoiceNo
581425
578757
578479
568375
568200
561894
561226
559509
558198
550193
0.000
0.025
0.050
0.075
0.100
0.125
Total Cost
InvoiceNo
550193
558198
559509
561226
561894
568200
568375
578479
578757
581425