

EDA: CUSTOMER BEHAVIOR ANALYSIS

1. Understanding the data

- MONTH : Month when the transaction occurred
- STORECODE : A unique identifier for each store
- DAY : Day when the transaction occurred
- BILL_ID : A unique identifier for each transaction or bill
- BILL_AMT : Represents total amount of the purchase
- QTY : Represents number of units of a product
- VALUE : The total value of the item(s) purchased, calculated as quantity multiplied by price
- PRICE : Price of a single unit
- GRP : Group or Category of the products
- SGRP : Subgroup or subcategory of product groups
- SSGRP : Subgroup of Subgroup of product groups
- CMP : Represents Company name
- MBRD : Represents Mother Brand
- BRD : Represents Brand name

2.Data preprocessing

##Load Necessary libraies

library(dplyr)

library(tidyverse)

library(lubridate)

#Step 1: Data Loading and Cleaning

View(Hackathon_Ideal_Data)

View(Hackathon_Working_Data)

View(Hackathon_Validation_Data)

View structure of datasets

```
str(Hackathon_Ideal_Data)
```

```
str(Hackathon_Working_Data)
```

```
str(Hackathon_Validation_Data)
```

Check for null values

```
> # Check for null values
> colSums(is.na(Hackathon_Ideal_Data))
  MONTH STORECODE   QTY   VALUE   GRP   SGRP   SSGRP   CMP   MBRD   BRD
      0         0      0      0      0      0      0      0      0      0
> colSums(is.na(Hackathon_Working_Data))
  MONTH STORECODE   DAY  BILL_ID  BILL_AMT   QTY   VALUE   PRICE   GRP   SGRP   SSGRP
      0         0      0      0      0      0      0      0      0      0      0
  CMP   MBRD   BRD
      0      0      0
> colSums(is.na(Hackathon_Validation_Data))
  ID STORECODE   MONTH   GRP
      0         0      0      0
```

##Summary Statistics

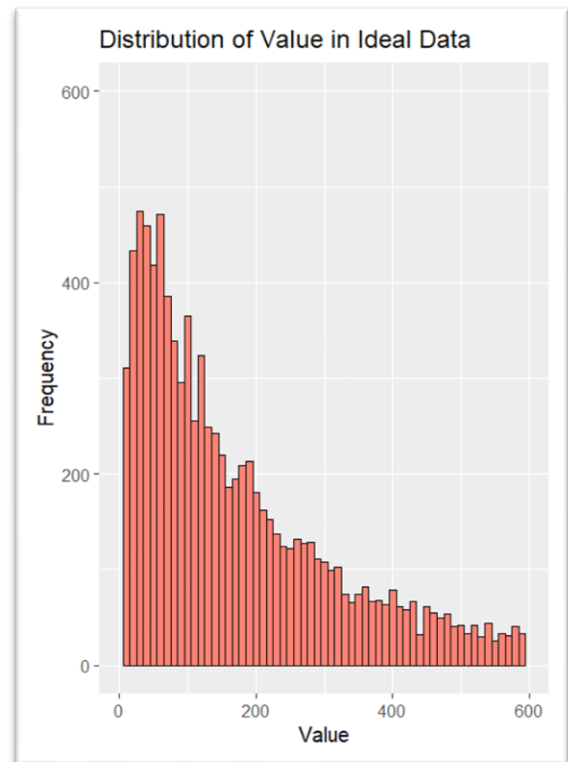
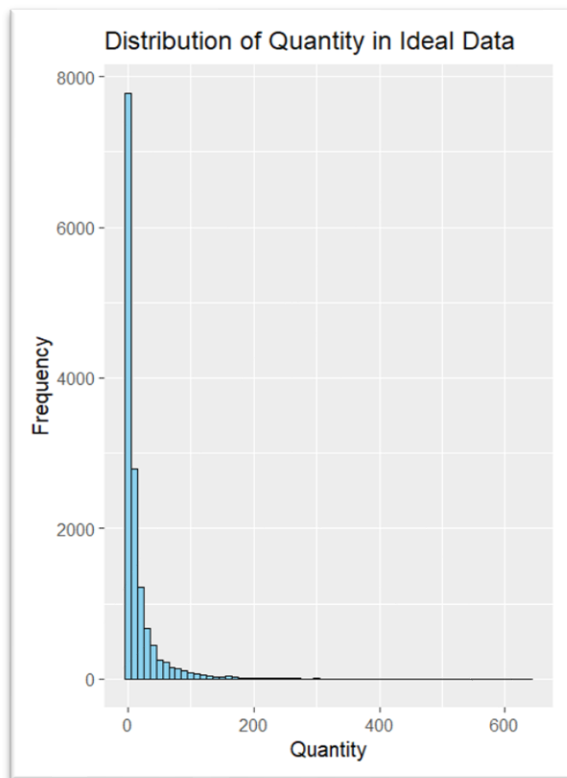
```
> ##Summary Statistics
>
> summary(Hackathon_Ideal_Data)
  MONTH      STORECODE      QTY      VALUE      GRP      SGRP
Length:14260 Length:14260 Min.   : 0.00 Min.   : 0.0 Length:14260 Length:14260
Class :character Class :character 1st Qu.: 1.00 1st Qu.: 10.0 Class :character Class :character
Mode  :character Mode  :character Median : 4.00 Median : 99.0 Mode  :character Mode  :character
      Mean : 16.35 Mean : 294.5
      3rd Qu.: 16.00 3rd Qu.: 283.0
      Max. :641.00 Max. :24185.0
  SSGRP      CMP      MBRD      BRD
Length:14260 Length:14260 Length:14260 Length:14260
Class :character Class :character Class :character Class :character
Mode  :character Mode  :character Mode  :character Mode  :character
```

```
> summary(Hackathon_Working_Data)
  MONTH      STORECODE      DAY      BILL_ID      BILL_AMT      QTY
M1:8363 Length:26985 Min.   : 1.00 Length:26985 Min.   : 0.0 Min.   : 0.500
M2:9192 Class :character 1st Qu.: 7.00 Class :character 1st Qu.: 40.0 1st Qu.: 1.000
M3:9430 Mode  :character Median :14.00 Mode  :character Median : 111.0 Median : 1.000
      Mean :15.17 Mean : 278.8 Mean : 4.105
      3rd Qu.:23.00 3rd Qu.: 280.0 3rd Qu.: 2.000
      Max. :31.00 Max. :7292.0 Max. :12000.000
  VALUE      PRICE      GRP      SGRP      SSGRP      CMP
Min.   : 0.00 Min.   : 0.00 Length:26985 Length:26985 Length:26985 Length:26985
1st Qu.: 10.00 1st Qu.: 10.00 Class :character Class :character Class :character Class :character
Median : 30.00 Median : 22.00 Mode  :character Mode  :character Mode  :character Mode  :character
Mean : 67.81 Mean : 52.81
3rd Qu.: 80.00 3rd Qu.: 64.00
Max. :3150.00 Max. :3150.00
  MBRD      BRD
Length:26985 Length:26985
Class :character Class :character
Mode  :character Mode  :character

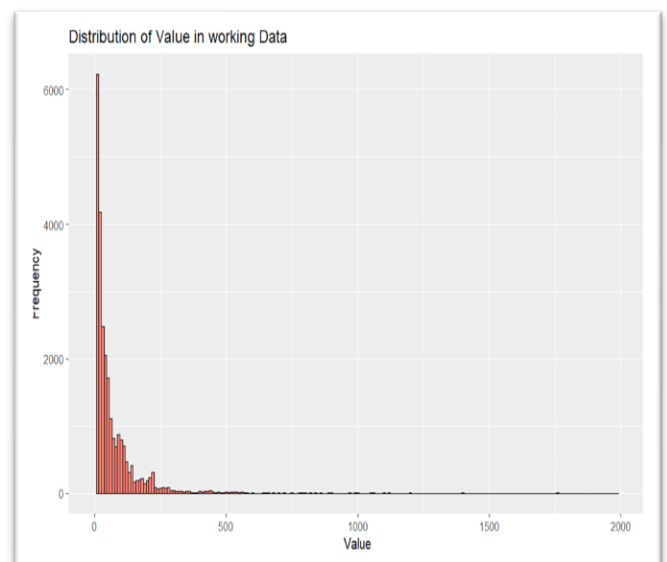
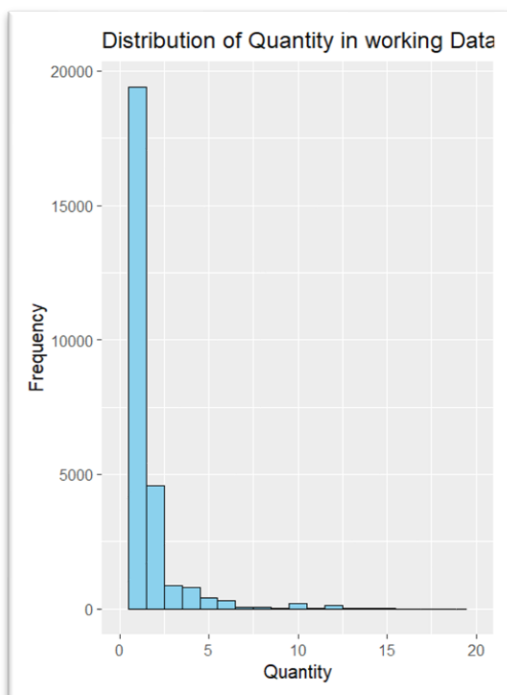
> summary(Hackathon_Validation_Data)
  ID      STORECODE      MONTH      GRP
Min.   :1112535 Length:2430 Length:2430 Length:2430
1st Qu.:1114964 Class :character Class :character Class :character
Median :1117393 Mode  :character Mode  :character Mode  :character
Mean :1117393
3rd Qu.:1119822
Max. :1122251
```

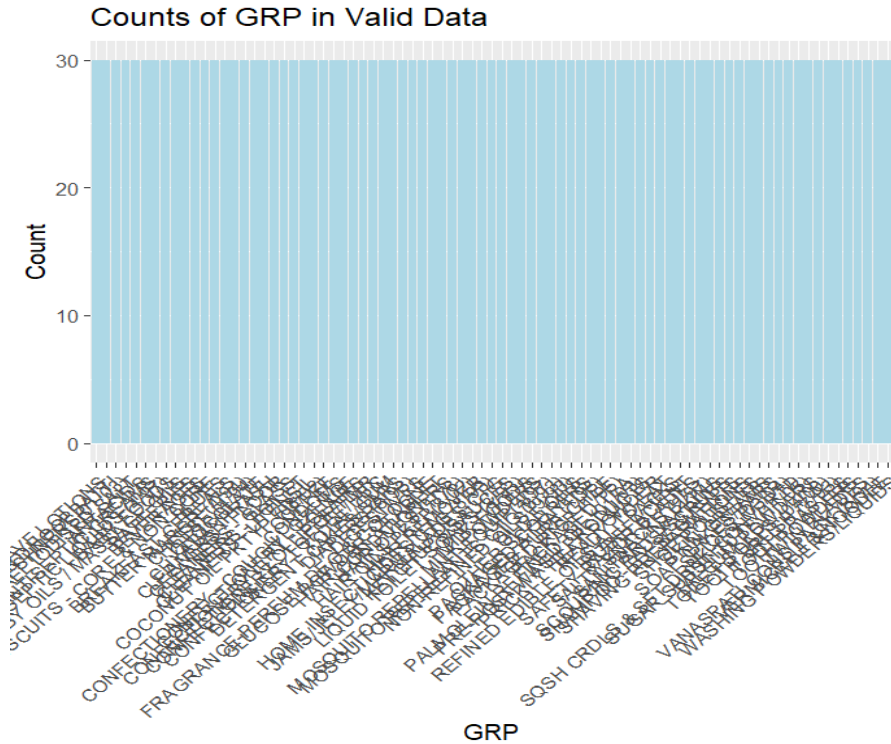
Step2: Visualize distributions of key variables

Ideal Data: Distribution of QTY and VALUE



Working Data: Distribution of QTY and VALUE





```
#step3: Analyze customer behavior in working_data
```

Total spend per customer

```
> # Total spend per customer
> total_spend <- Hackathon_Working_Data %>%
+   group_by(STORECODE) %>%
+   summarise(TotalSpend=sum(BILL_AMT));total_spend
# A tibble: 10 x 2
  STORECODE TotalSpend
  <chr>         <dbl>
1 N1          626465.
2 N10         550107
3 N2          794062
4 N3          494536.
5 N4          872216.
6 N5          695755
7 N6          366530
8 N7         1036822.
9 N8          492817.
10 N9         1592872.
```

Average transaction value

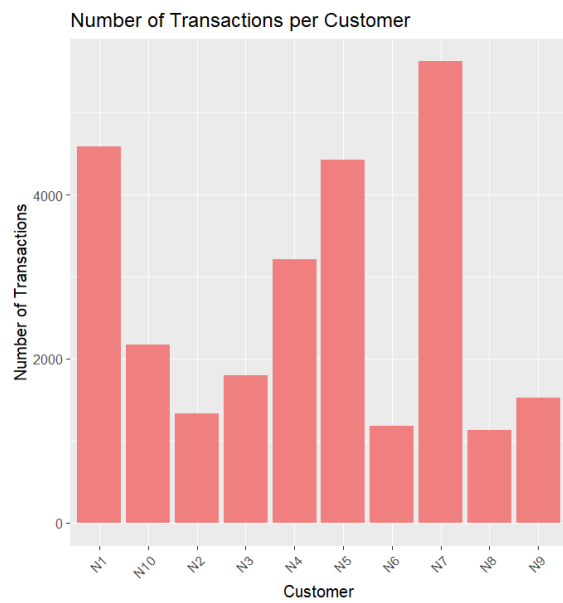
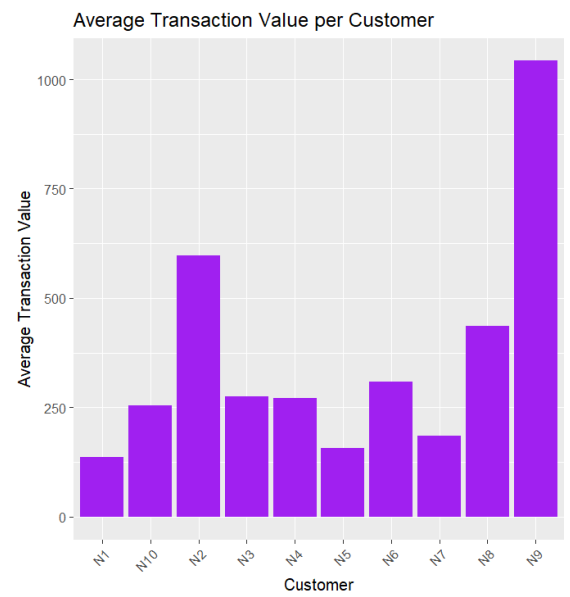
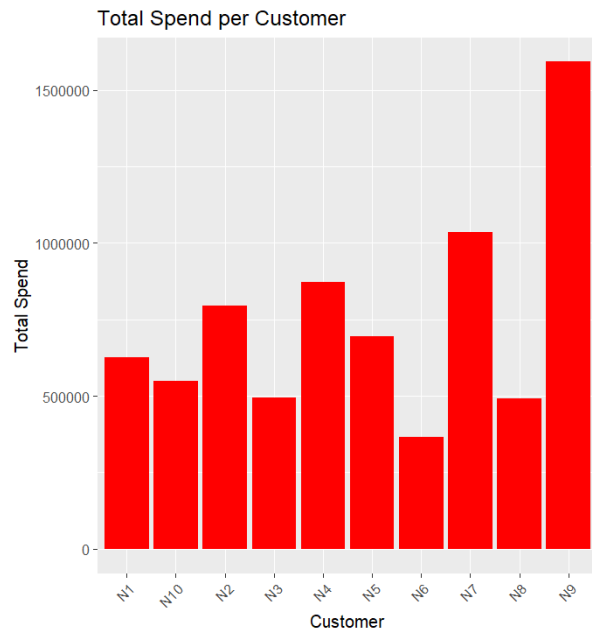
```
> # Average transaction value
> avg_transaction_value <- Hackathon_Working_Data %>%
+   group_by(STORECODE) %>%
+   summarize(AvgTransactionValue = mean(BILL_AMT));avg_transaction_value
# A tibble: 10 × 2
  STORECODE AvgTransactionValue
  <chr>          <dbl>
1 N1             137.
2 N10            254.
3 N2             597.
4 N3             276.
5 N4             272.
6 N5             157.
7 N6             309.
8 N7             184.
9 N8             437.
10 N9            1042.
```

Number of transactions per customer

```
> # Number of transactions per customer
> num_transactions <- Hackathon_Working_Data %>%
+   group_by(STORECODE) %>%
+   summarize(NumTransactions = n());num_transactions
# A tibble: 10 × 2
  STORECODE NumTransactions
  <chr>          <int>
1 N1             4583
2 N10            2169
3 N2             1331
4 N3             1793
5 N4             3212
6 N5             4431
7 N6             1185
8 N7             5625
9 N8             1128
10 N9            1528
```

```
> # Merge these metrics
> customer_metrics <- total_spend %>%
+   inner_join(avg_transaction_value, by = "STORECODE") %>%
+   inner_join(num_transactions, by = "STORECODE");customer_metrics
# A tibble: 10 × 4
  STORECODE TotalSpend AvgTransactionValue NumTransactions
  <chr>          <dbl>          <dbl>          <int>
1 N1             626465.             137.             4583
2 N10            550107.             254.             2169
3 N2             794062.             597.             1331
4 N3             494536.             276.             1793
5 N4             872216.             272.             3212
6 N5             695755.             157.             4431
7 N6             366530.             309.             1185
8 N7            1036822.             184.             5625
9 N8             492817.             437.             1128
10 N9            1592872.            1042.             1528
```

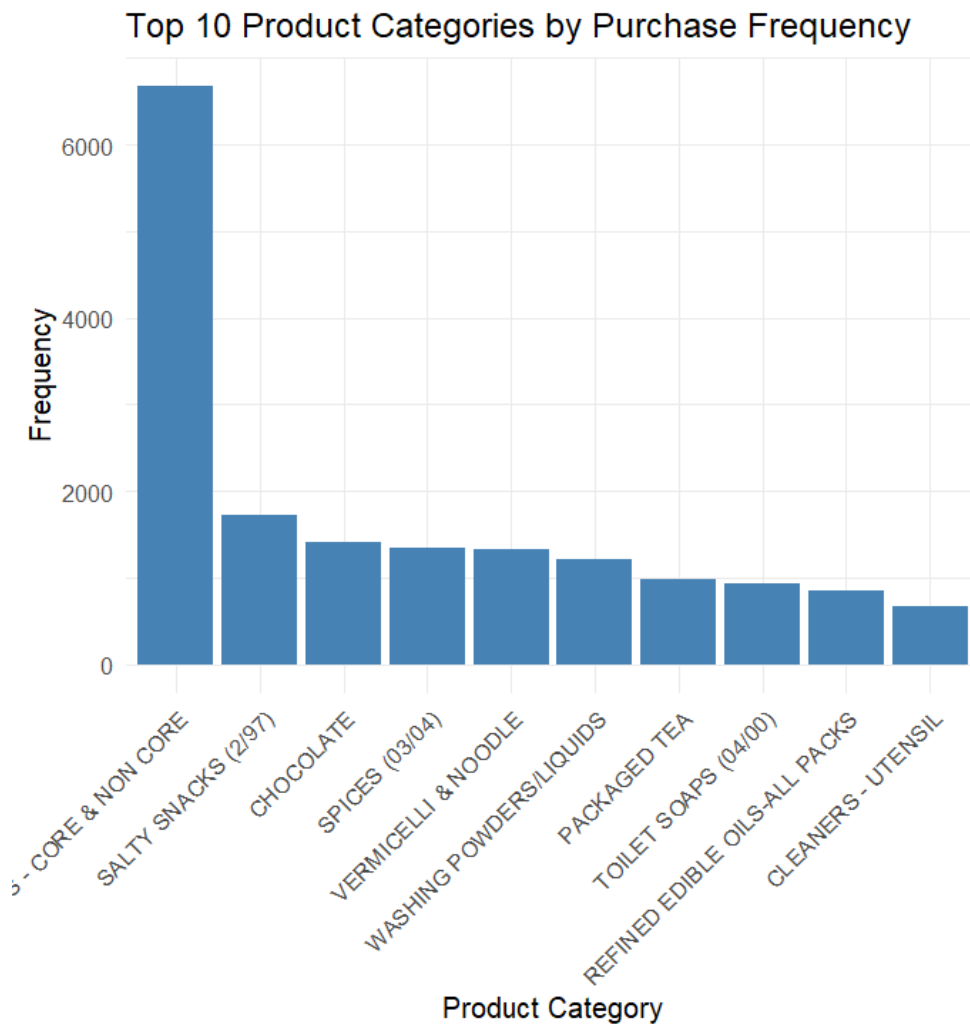
Visualize customer metrics



##Step 4: Analyzing Purchasing Patterns

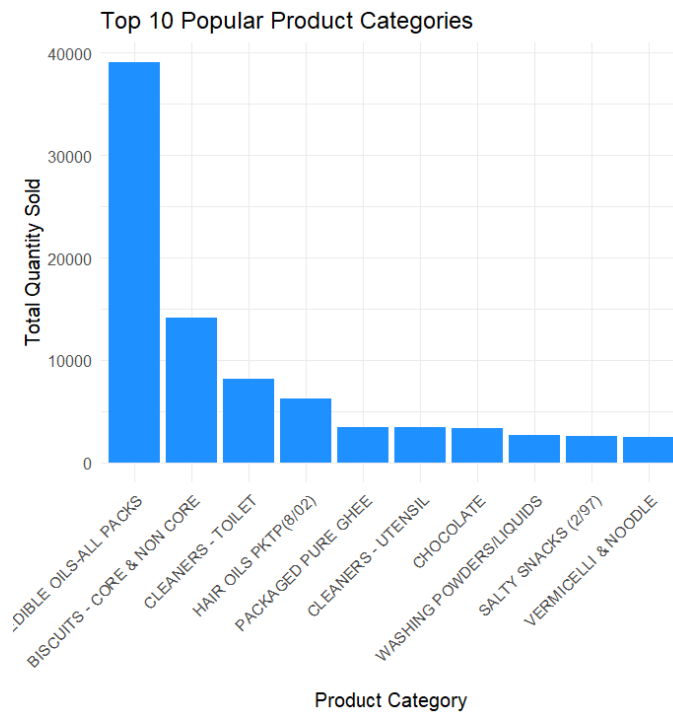
Analyze the frequency of purchases for different product categories

#Frequency of Purchases for Different Product Categories

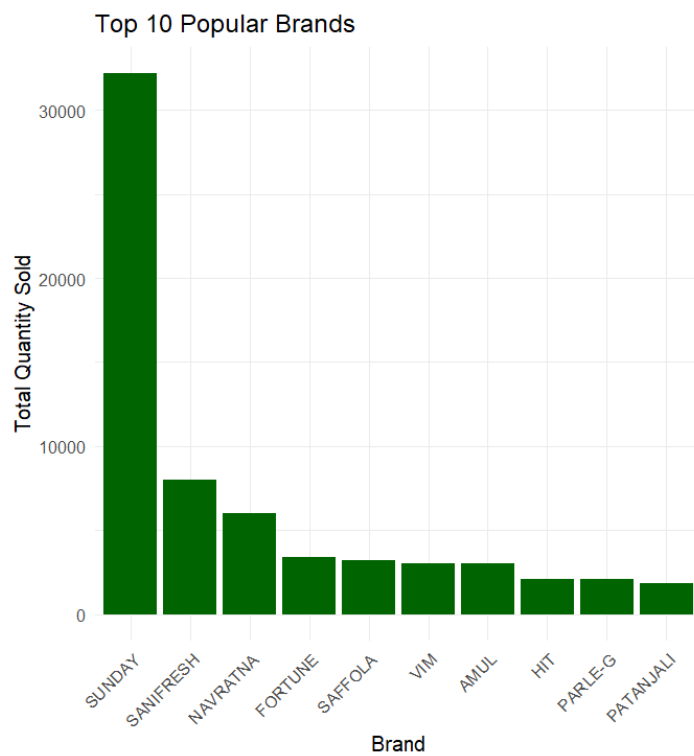


#Step 5: Analyzing Customer Preferences

#Most Popular Product Categories and Brands



Determine the most popular brands



#Step 6: Analyze Monthly Purchasing Patterns

##Aggregate the data by month to analyze the frequency of purchases and total spend.

```
> # Analyze peak purchasing times by month
> monthly_purchases <- Hackathon_Working_Data %>%
+   group_by(MONTH) %>%
+   summarize(Frequency = n(), TotalSpend = sum(BILL_AMT, na.rm = TRUE));monthly_purchases
# A tibble: 3 × 3
  MONTH Frequency TotalSpend
  <fct>    <int>    <dbl>
1 M1       8363  2301409.
2 M2       9192  2908232.
3 M3       9430  2312541.
```

