Homework 2

Yuhao Mao 3170102264

2020/7/7

Problem 1: Calif_penn_2011 data set.

1. Reading and Cleaning.

a. Load data into data frame *ca_pa*.

```
ca_pa <- read.csv("data/calif_penn_2011.csv")
```

b. Get number of rows and columns.

```
dim(ca_pa)
```

```
## [1] 11275    34
```

There are 11275 rows and 34 columns in total.

c. Column sum and apply command.

```
colSums(apply(ca_pa,c(1,2),is.na))
```

```
##                            X                     GEO.id2
##                            0                           0
##                      STATEFP                    COUNTYFP
##                            0                           0
##                      TRACTCE                  POPULATION
##                            0                           0
##                     LATITUDE                   LONGITUDE
##                            0                           0
##            GEO.display.label          Median_house_value
##                            0                         599
##                  Total_units                Vacant_units
##                            0                           0
##                 Median_rooms  Mean_household_size_owners
##                          157                         215
## Mean_household_size_renters         Built_2005_or_later
##                          152                          98
##            Built_2000_to_2004                 Built_1990s
##                           98                          98
##                  Built_1980s                 Built_1970s
##                           98                          98
##                  Built_1960s                 Built_1950s
##                           98                          98
##                  Built_1940s        Built_1939_or_earlier
```

```
##                       98                        98
##                Bedrooms_0                Bedrooms_1
##                       98                        98
##                Bedrooms_2                Bedrooms_3
##                       98                        98
##                Bedrooms_4        Bedrooms_5_or_more
##                       98                        98
##                    Owners                   Renters
##                      100                       100
##   Median_household_income     Mean_household_income
##                      115                       126
```

The `apply()` function applies `is.na()` function on every entry of ca_pa data frame. After that, `colSums()` function sums all columns. As a whole this command calculates the number of NA entries in each column.

d. Filter out NA values.

```
ca_pa.clean <- na.omit(ca_pa)
```

e. Number of rows that are filtered out:

```
dim(ca_pa)[1]-dim(ca_pa.clean)[1]
```

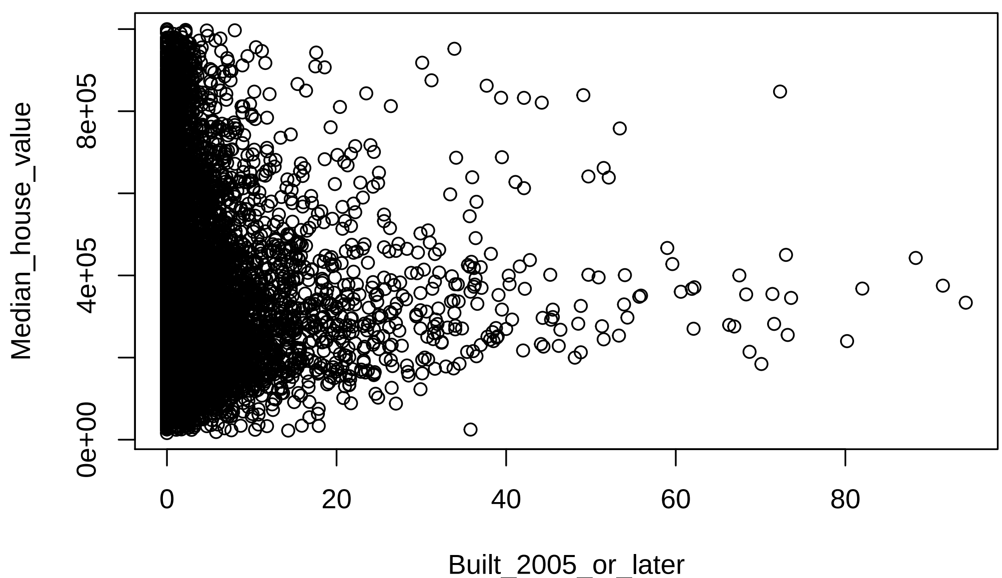
```
## [1] 670
```

f. In (c) we calculate the number of NAs in each column and in (e) we calculate the number of rows that contain NAs. Since all column sums in (c) is smaller than our result in (e), it is compatible.

2. The very new house.

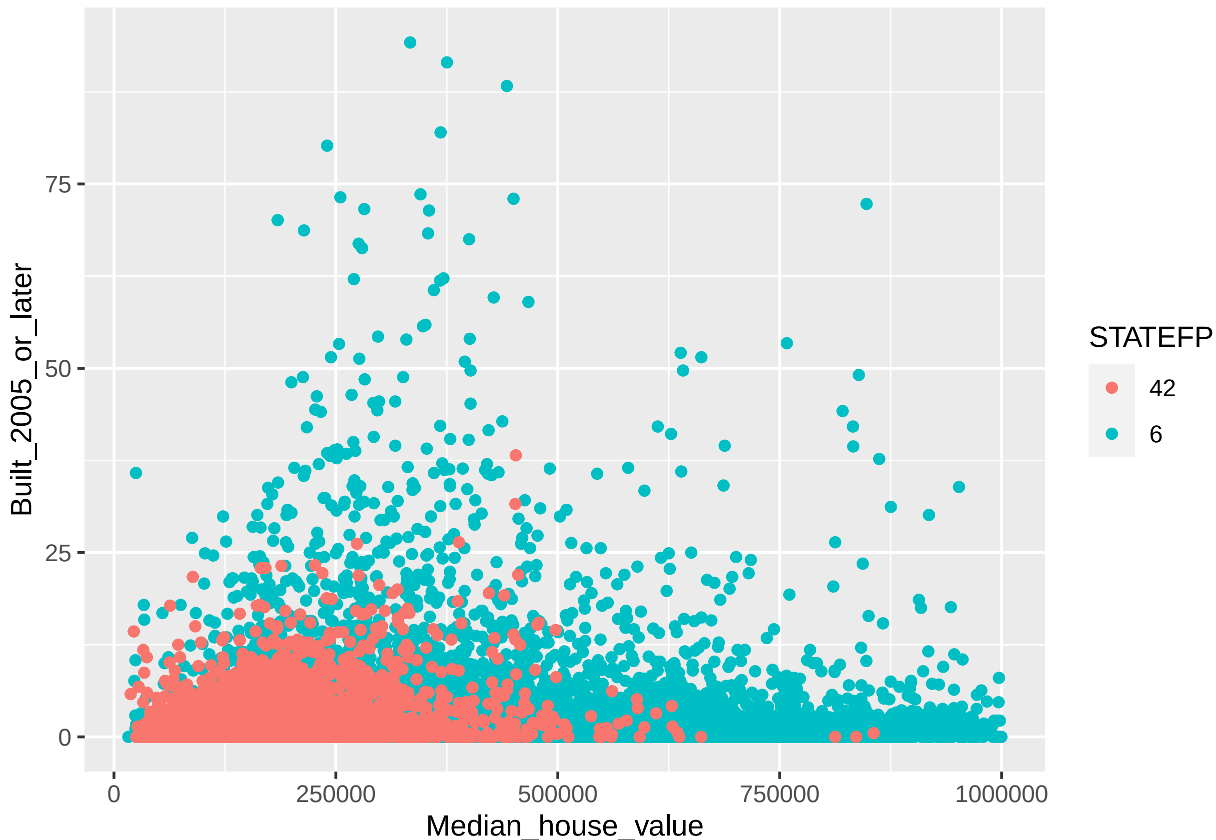
a. Plot median house price against percentage of houses.

```
plot(Median_house_value~Built_2005_or_later, data=ca_pa.clean)
```



b. Plot according to STATEFP.

```
library(ggplot2)
ca_pa.clean$STATEFP <- as.character(ca_pa.clean$STATEFP)
ggplot(ca_pa.clean, aes(Median_house_value, Built_2005_or_later, color=STATEFP)) + geom_point()
```




```
ca_pa.clean$STATEFP <- as.numeric(ca_pa.clean$STATEFP)
```

3. Nobody home.

a. The vacancy rate.

```
ca_pa.clean["Vacancy_rate"]<- ca_pa.clean$Vacant_units/ca_pa.clean$Total_units
```

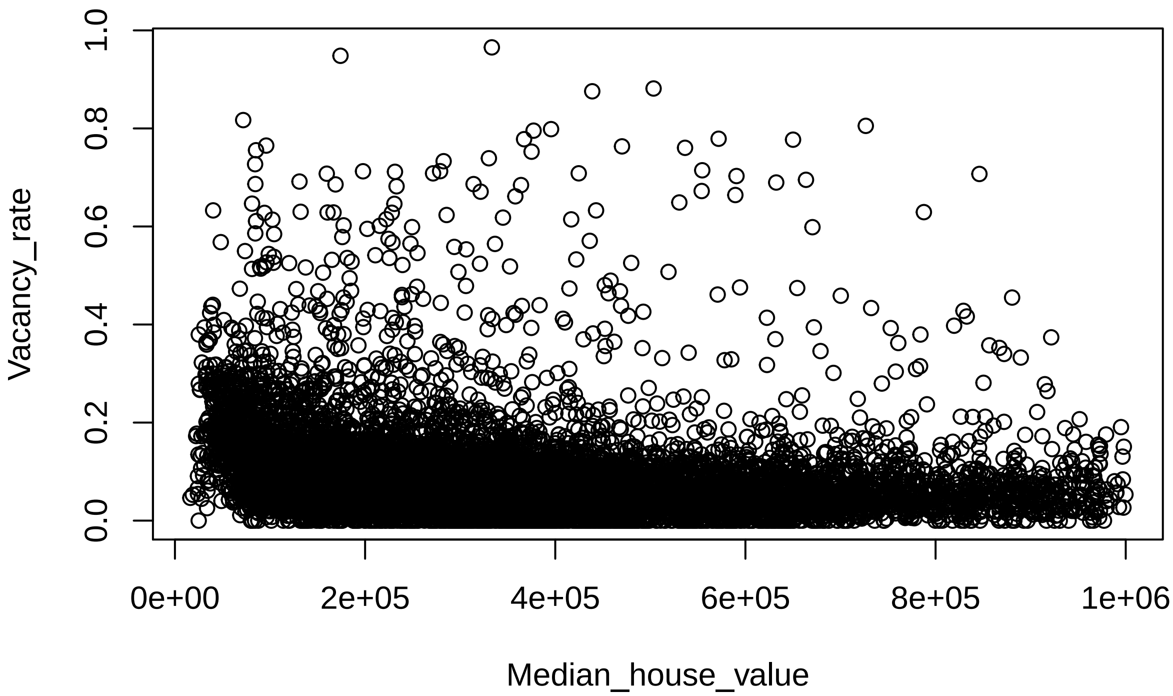
The maximum:

```
summary(ca_pa.clean$Vacancy_rate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.03846 0.06767 0.08889 0.10921 0.96531
```

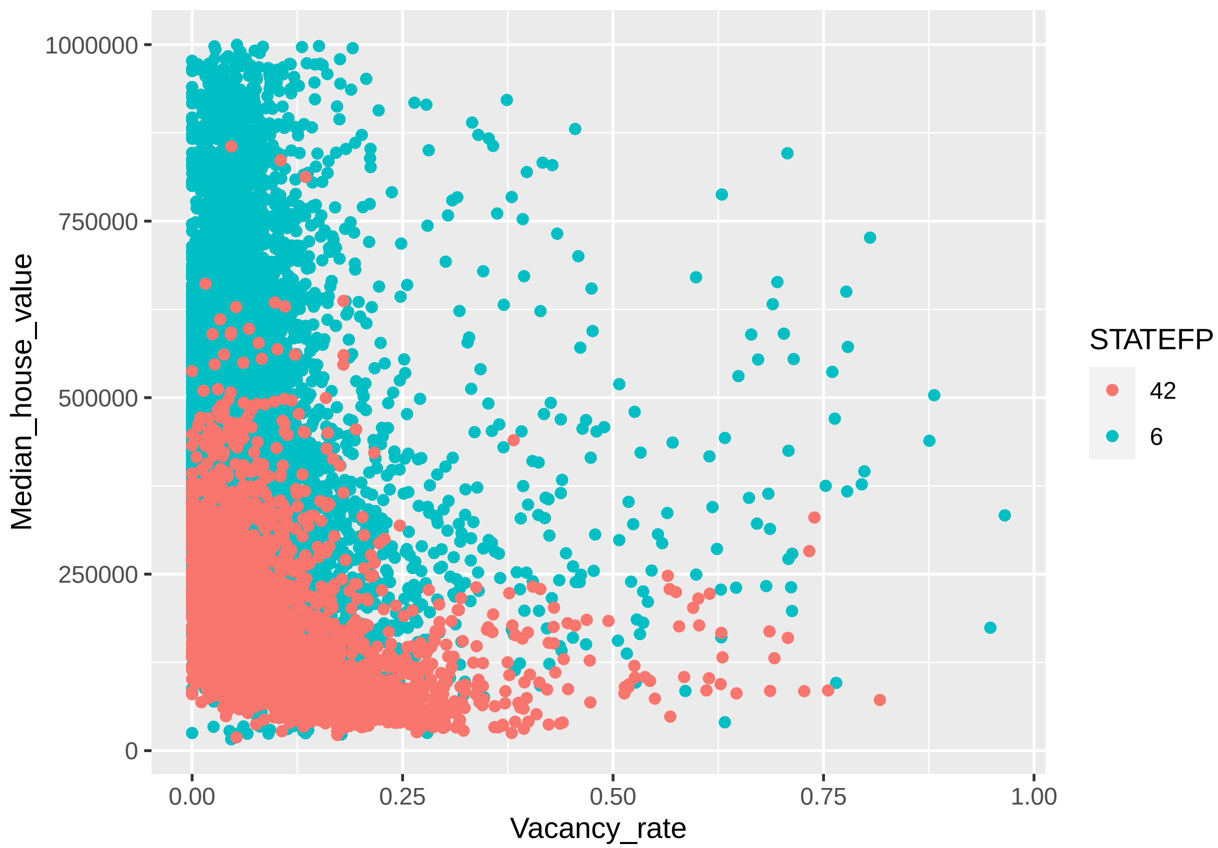
b. Plot vacancy rate against median house value.

```
plot(Vacancy_rate~Median_house_value, data=ca_pa.clean)
```



c. Repeat (b) according to states.

```
ca_pa.clean$STATEFP <- as.character(ca_pa.clean$STATEFP)
ggplot(ca_pa.clean, aes(Vacancy_rate, Median_house_value, color=STATEFP)) + geom_point()
```




```
ca_pa.clean$STATEFP <- as.numeric(ca_pa.clean$STATEFP)
```

There is a difference across these two states. In California (STATEFP 6), there are more valuable houses with low vacancy rate. In addition, in Pennsylvania (STATEFP 42), highly vacant houses are all of low median house value.

4. Counties.

a. Explain following codes.

```
## Extract indexs of all Alameda County rows which is in California.
acca <- c()
for (tract in 1:nrow(ca_pa.clean)) {
  if (ca_pa.clean$STATEFP[tract] == 6) {
    if (ca_pa.clean$COUNTYFP[tract] == 1) {
      acca <- c(acca, tract)
    }
  }
}
## Extract all median house values of Alameda County.
accamhv <- c()
for (tract in acca) {
  accamhv <- c(accamhv, ca_pa.clean[tract,10])
}
## Get median of Median_house_value of Alameda county.
median(accamhv)
```

```
## [1] 474050
```

b. Replace the code chunk by single line.

```r
median(ca_pa.clean$Median_house_value[ca_pa.clean$COUNTYFP==1 & ca_pa.clean$STATEFP==6])
```

```
## [1] 474050
```

c. For Alameda, Santa Clara and Allegheny Counties, what were the average percentages of housing built since 2005?

```r
## Alameda
Alameda.index <- ca_pa.clean$COUNTYFP==1 & ca_pa.clean$STATEFP==6
mean(ca_pa.clean$Built_2005_or_later[Alameda.index])
```

```
## [1] 2.820468
```

```r
## Santa Clara
Santa_Clara.index <- ca_pa.clean$COUNTYFP==85 & ca_pa.clean$STATEFP==6
mean(ca_pa.clean$Built_2005_or_later[Santa_Clara.index])
```

```
## [1] 3.200319
```

```r
## Allegheny
Allegheny.index <- ca_pa.clean$COUNTYFP==3 & ca_pa.clean$STATEFP==42
mean(ca_pa.clean$Built_2005_or_later[Allegheny.index])
```

```
## [1] 1.474219
```

d. Correlations.

```r
## Whole data
cor(ca_pa.clean[c("Median_house_value", "Built_2005_or_later")])
```

```
##                     Median_house_value Built_2005_or_later
## Median_house_value          1.00000000         -0.01893186
## Built_2005_or_later        -0.01893186          1.00000000
```

```r
## All California
california <- ca_pa.clean[ca_pa.clean$STATEFP==6,]
cor(california[c("Median_house_value", "Built_2005_or_later")])
```

```
##                     Median_house_value Built_2005_or_later
## Median_house_value           1.0000000          -0.1153604
## Built_2005_or_later         -0.1153604           1.0000000
```

```r
## All Pennsylvania
penn <- ca_pa.clean[ca_pa.clean$STATEFP==42,]
cor(penn[c("Median_house_value", "Built_2005_or_later")])
```

```
##                     Median_house_value Built_2005_or_later
## Median_house_value           1.0000000           0.2681654
## Built_2005_or_later          0.2681654           1.0000000
```

```r
## Alameda

cor(ca_pa.clean[Alameda.index,][c("Median_house_value", "Built_2005_or_later")])
```

```
##                     Median_house_value Built_2005_or_later
## Median_house_value          1.00000000          0.01303543
## Built_2005_or_later         0.01303543          1.00000000
```

```r
## Santa Clara
cor(ca_pa.clean[Santa_Clara.index,][c("Median_house_value", "Built_2005_or_later")])
```

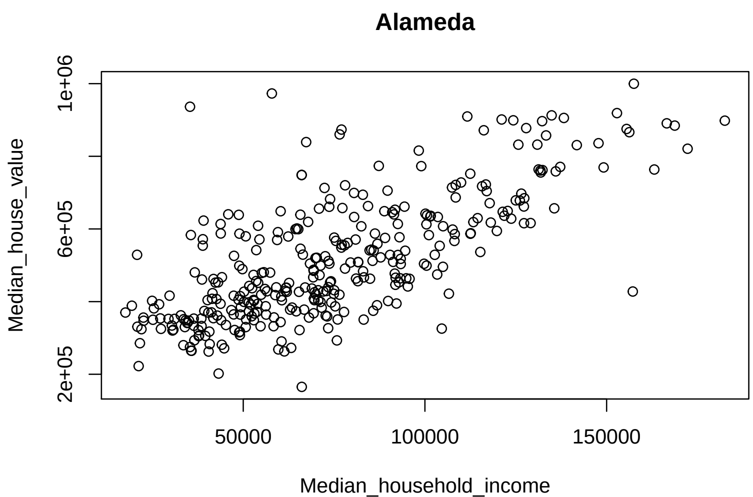
```
##                     Median_house_value Built_2005_or_later
## Median_house_value           1.0000000          -0.1726203
## Built_2005_or_later         -0.1726203           1.0000000
```

```r
## Allegheny
cor(ca_pa.clean[Allegheny.index,][c("Median_house_value", "Built_2005_or_later")])
```

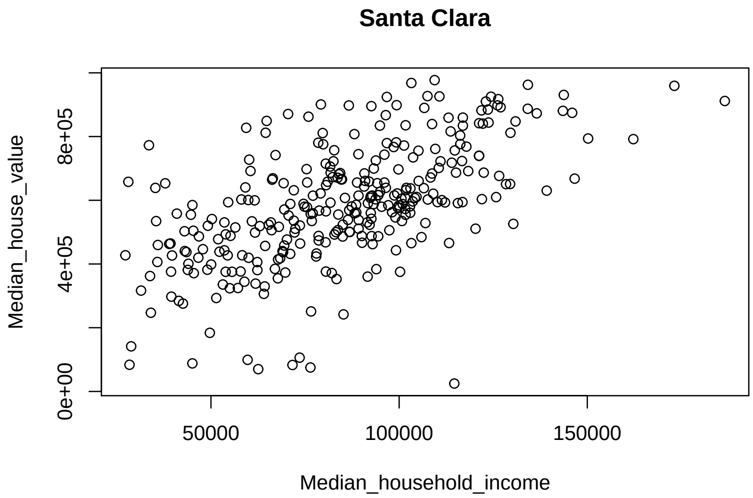
```
##                     Median_house_value Built_2005_or_later
## Median_house_value           1.0000000           0.1939652
## Built_2005_or_later          0.1939652           1.0000000
```

e. Plot median house values against median income for three counties.

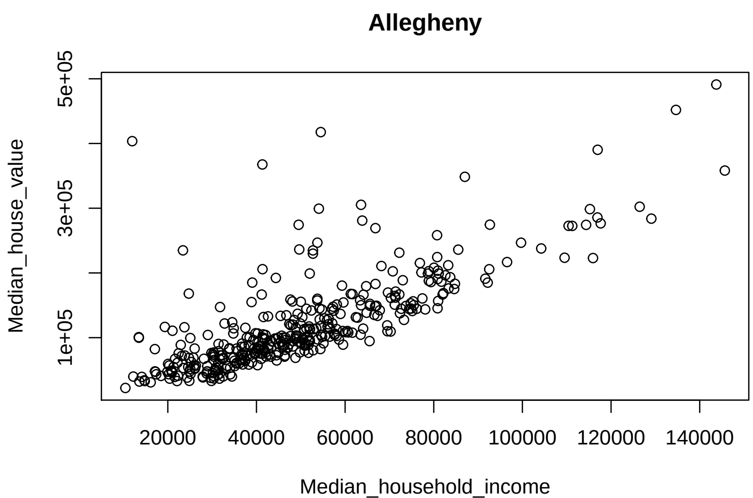
```r
## Alameda
plot(Median_house_value~Median_household_income, data=ca_pa.clean[Alameda.index,])
title("Alameda")
```




```r
## Santa Clara
plot(Median_house_value~Median_household_income, data=ca_pa.clean[Santa_Clara.index,])
title("Santa Clara")
```




```r
## Allegheny
plot(Median_house_value~Median_household_income, data=ca_pa.clean[Allegheny.index,])
title("Allegheny")
```



Problem 2: Explain following codes about `table()`.

```r
## table uses the cross-classifying factors to build a contingency table of the counts at each combinat
## There are 91 female and 92 male.
gender <- factor(c(rep("female", 91), rep("male", 92)))
table(gender)
```

```
## gender
## female   male
##     91     92
```

```r
## Specifying levels gives ordered factors.
## There are 92 male and 91 female.
gender <- factor(gender, levels=c("male", "female"))
table(gender)
```

```
## gender
##   male female
##     92     91
```

```r
## If some entry in the specified levels do not exist, NA will be used in the place.
gender <- factor(gender, levels=c("Male", "female"))
# Note the mistake: "Male" should be "male"
table(gender)
```

```
## gender
##   Male female
##      0     91
```

```r
## By default, NA and NaN are excluded.
## If exclude is set to NULL, then NULL will also be included.
table(gender, exclude=NULL)
```

```
## gender
##   Male female   <NA>
##      0     91     92
```

Problem 3: Function to calculate percentile

```r
percentile <- function(v, cutoff){
  count <- 0
  for(x in v){
    if(x>cutoff){
      count <- count+1
    }
  }
  return(count/length(v))
}
```

```r
## Tests
v <- seq(1,100)
percentile(v, 50)
```

```
## [1] 0.5
```

Problem 4: Rabbit data set.

```r
library(MASS)
data(Rabbit)
Dose <- unstack(Rabbit, Dose ~ Animal)[,1]
Treatment <- unstack(Rabbit, Treatment ~ Animal)[,1]
```

```r
BPchange <- unstack(Rabbit, BPchange ~ Animal)
Rabbit.change <- data.frame(Treatment, Dose, BPchange)
Rabbit.change
```

```
##    Treatment   Dose    R1    R2    R3    R4   R5
## 1    Control   6.25  0.50  1.00  0.75  1.25  1.5
## 2    Control  12.50  4.50  1.25  3.00  1.50  1.5
## 3    Control  25.00 10.00  4.00  3.00  6.00  5.0
## 4    Control  50.00 26.00 12.00 14.00 19.00 16.0
## 5    Control 100.00 37.00 27.00 22.00 33.00 20.0
## 6    Control 200.00 32.00 29.00 24.00 33.00 18.0
## 7        MDL   6.25  1.25  1.40  0.75  2.60  2.4
## 8        MDL  12.50  0.75  1.70  2.30  1.20  2.5
## 9        MDL  25.00  4.00  1.00  3.00  2.00  1.5
## 10       MDL  50.00  9.00  2.00  5.00  3.00  2.0
## 11       MDL 100.00 25.00 15.00 26.00 11.00  9.0
## 12       MDL 200.00 37.00 28.00 25.00 22.00 19.0
```