

```
library(tidyverse, MASS, readxl)
```

```
data = read.csv(file.choose())
```

1.

Code:

```
summary(data)
```

Output:

```
Inspection.ID   DBA.Name     AKA.Name     License..   Facility.Type
Risk
Min.   : 44247 Length:196825 Length:196825 Min.   : 0 Length:196825
Length:196825
1st Qu.:1150792 Class :character Class :character 1st Qu.:1225118 Class
:character Class:character
Median :1493299 Mode :character Mode :character Median :1979973
Mode :character Mode :character
Mean   :1449379                      Mean :1601134
3rd Qu.:2009233                      3rd Qu.:2240843
Max.   :2352738                      Max.   :9999999
NA's :17
Address        City        State        Zip
Inspection.Date Inspection.Type
Length:196825 Length:196825 Length:196825 Min. :10014
Length:196825 Length:196825
Class :character Class :character Class :character 1st Qu.:60614 Class
:character Class:character
Mode :character Mode :character Mode :character Median :60625 Mode
:character Mode :character
Mean :60629
3rd
Qu.:60643
Max. :60827
NA's :50
Results        Violations    Latitude    Longitude
Location.Historical.Wards.2003.2015
Length:196825 Length:196825 Min. :41.64 Min. : -87.91 Length:196825
Mode:logical
Class :character Class :character 1st Qu.:41.83 1st Qu.: -87.71 Class
:character NA's:196825
Mode :character Mode :character Median :41.89 Median : -87.67 Mode
:character
```

Mean :41.88 Mean :-87.68
3rd Qu.:41.94 3rd Qu.: -87.64
Max. :42.02 Max. :-87.53

NA's :690 NA's :690
Zip.Codes Community.Areas Census.Tracts Wards
Mode:logical Mode:logical Mode:logical Mode:logical
NA's:196825 NA's:196825 NA's:196825 NA's:196825

2.

Code:

```
View(head(data, 6))
```

Output:

Shows 1st 6 entries of dataframe

3.

Code:

```
View(tail(data, 10))
```

Output:

Shows last 10 entries of dataframe.

4.

Code:

```
dim(data)
```

Output:

```
196825  22
```

5.

Code:

```
class(data)
```

Output:

```
"data.frame"
```

6.

Code:

```
colnames(data)
```

Output:

```
[1] "Inspection.ID"  "DBA.Name"      "AKA.Name"      "License.."
[5] "Facility.Type"  "Risk"          "Address"        "City"
[9] "State"          "Zip"           "Inspection.Date" "Inspection.Type"
[13] "Results"        "Violations"    "Latitude"       "Longitude"
[17] "Location"       "Historical.Wards.2003.2015" "Zip.Codes"
"Community.Areas"
[21] "Wards"
"Census.Tracts"
```

7.

Code:

```
numeric_data = data[unlist(lapply(data, FUN =  
is.numeric)))] for (col in colnames(numeric_data)) {  
  print(col)  
  print(fivenum(unlist(numeric_data[col])))  
}
```

Output:

```
[1] "Inspection.ID"  
Inspection.ID196811 Inspection.ID145757  
Inspection.ID108177 Inspection.ID50095      Inspection.ID6  
      44247      1150792      1493299      2009233      2352738  
[1] "License.."  
License..820 License..173540 License..54181 License..44589  
License..195886 0 1225115      1979973      2240843      9999999  
[1] "Zip"  
Zip6676 Zip49478 Zip164472 Zip107493 Zip196696  
10014 60614 60625 60643 60827  
[1] "Latitude"  
Latitude17868 Latitude185616 Latitude46770 Latitude52279  
Latitude190811 41.64467      41.83490 41.89198 41.93981  
42.02106  
[1] "Longitude"  
Longitude2 Longitude47534 Longitude23559  
Longitude9143 Longitude184393  
-87.91443 -87.70774 -87.66681 -87.63496 -87.52509
```

8.

Code:

```
data[3]
```

Output:

AKA.Name

```
1      CHIPOTLE MEXICAN GRILL
2      CHILI'S (T1-B14)
3 NICK'S FIRE GRILL STEAK & LEMONADE INC.
4
5      MOSAIC EARLY CHILDHOOD ACADEMY
6      AMARIT RESTAURANT
7      CHIPOTLE MEXICAN GRILL
8      NAIBOA LATIN STORE
9      JAMIESON PUBLIC SCHOOL
10     GATEWAY MONTESSORI
[ reached 'max' / getOption("max.print") -- omitted 196815 rows ]
```

9.

Code:

```
data_lis = data[data$License.. > 2000000 & data$License.. < 3000000,]
```

Output:

NA

10.

Code:

```
blue_line = data[data$DBA.Name == "BLUE LINE",]
```

Output:

NA

11.

Code:

```
data[, c("Inspection.ID", "Facility.Type")]
```

Output:

	Inspection.ID	Facility.Type
1	2352683	Restaurant
2	2352734	
	Restaurant 3	2352713
4	2352701	Restaurant
5	2352727	Children's Services Facility
6	2352738	Restaurant
7	2352684	Restaurant
8	2352702	Grocery
Store 9		2352718
	School	
10	2352696	Children's Services Facility

12.

Code:

```
data[grep("High", data$Risk), "DBA.Name"]
```

Output:

```
[1] "CHIPOTLE MEXICAN GRILL"      "CHILI'S T-I"
[3] "NICK'S FIRE GRILL STEAK & LEMONADE INC."  "TAQUERIA BLUE LINE
INC."
[5] "PORTAGE PARK DAY NURSERY"      "AMARIT RESTAURANT"
[7] "CHIPOTLE MEXICAN GRILL"      "JAMIESON"
[9] "GATEWAY MONTESSORI SCHOOL"    "BLUE LINE"
[11] "SWEETGREEN"                  "SAME SAME"
[13] "CHIPOTLE MEXICAN GRILL"      "HARRY CARAY'S SHORTSTOP"
[15] "THE GRAIL CAFE"              "SUBWAY"
[17] "AMARIT RESTAURANT"           "RPM SEAFOOD"
[19] "LITTLE FEET BIG STEPS CENTER"  "KENTUCKY FRIED CHICKEN"
```

13.

Code:

```
View(head(data[data$Results == "Fail",], 2))
```

Output:

Shows 2nd column where results are Fail.

14.

Code:

```
nrow(data)
```

Output:

196825

15.

Code:

```
which(unlist(lapply(data, is.na)))
```

Output:

Check for NA values

16.

Code:

```
df = data %>% select_if(~ !all(is.na(.)))
```

Output:

Removes last 5 columns with null values.

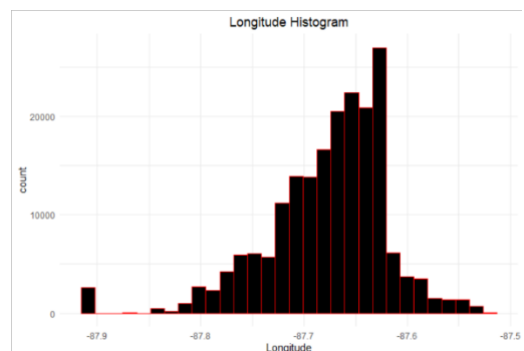
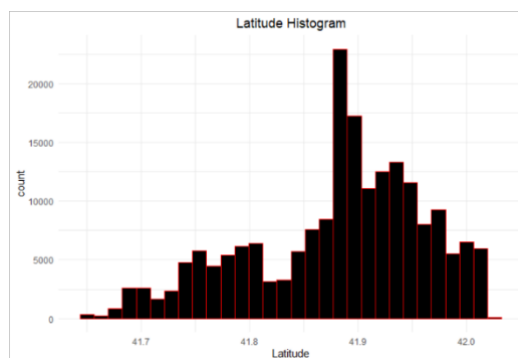
17.

Code:

```
ggplot(data, mapping = aes(x = Latitude)) +  
  geom_histogram(color = "red", fill =  
    "black") + ggtitle("Latitude Histogram")  
  + theme_minimal() +  
  theme(plot.title = element_text(hjust = 0.5))
```

```
ggplot(data, mapping = aes(x =  
  Longitude)) + geom_histogram(color =  
    "red", fill = "black") + ggtitle("Longitude  
  Histogram") + theme_minimal() +  
  theme(plot.title = element_text(hjust =
```

0.5)) Output:



18.

Code:

```
is.na(data$Latitude) =  
mean(data$Latitude)  
is.na(data$Longitude) =  
mean(data$Longitude)
```

Output:

Replaces NA values with the mean.

19.

Code:

```
which(unlist(lapply(data, duplicated)) == T)
```

Output:

Checks for duplicates.

20.

Code:

```
data %>% count(Risk, Results) %>% filter(Results ==
```

"Pass") Output:

	Risk	Results	n
1		Pass	12
2	Risk 1 (High)	Pass	78054
3	Risk 2 (Medium)	Pass	20595
4	Risk 3(Low)	Pass	7405

21.

Code:

```
data$Inspection.Date = as.Date(data$Inspection.Date, format =  
"%m/%d/%Y") data["Year"] = format(data$Inspection.Date, "%Y")  
data["Month"] = format(data$Inspection.Date, "%B")  
data["Day"] = format(data$Inspection.Date, "%d")
```

Output:

Adding in the Year, Month and Day columns.