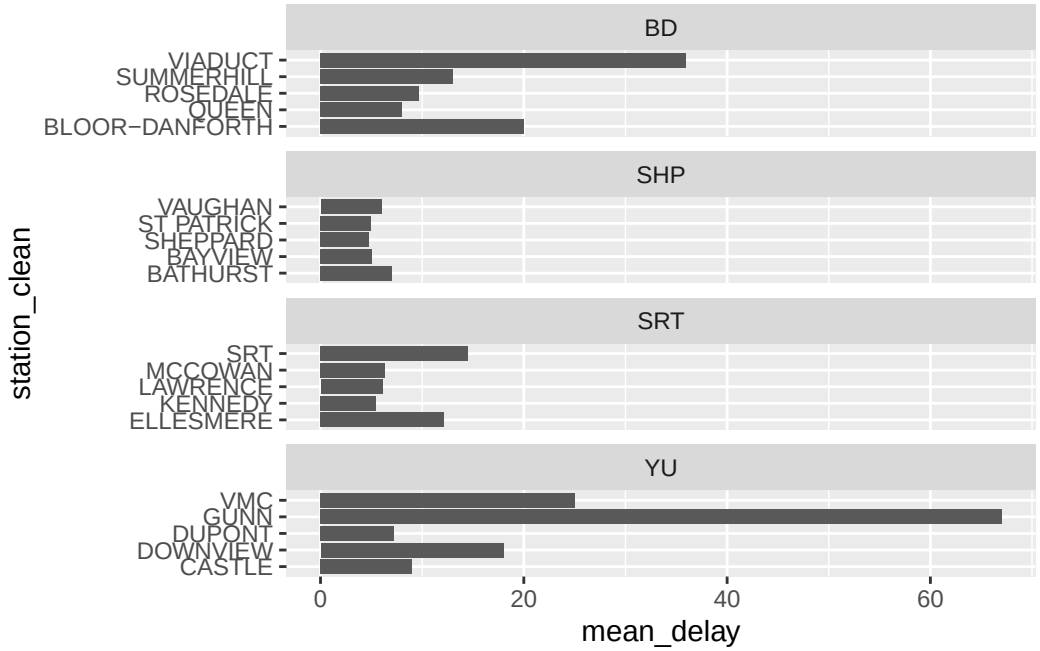
STA2201S_lab2

1. Using the `delay_2022` data, plot the five stations with the highest mean delays. Facet the graph by `line`



2. Using the `opendatatoronto` package, download the data on mayoral campaign contributions for 2014. Hints:

```
+ find the ID code you need for the package you need by searching for 'campaign' in the `all_
+ you will then need to `list_package_resources` to get ID for the data file
+ note: the 2014 file you will get from `get_resource` has a bunch of different campaign con
```

```
res2 <- list_package_resources("f6651a40-2f52-46fc-9e04-b760c16edd5c")
camp_2014 <- get_resource("5b230e92-0a22-4a15-9572-0b19cc222985")
mcamp_2014 <- camp_2014[["2_Mayor_Contributions_2014_election.xls"]]
```

3. Clean up the data format (fixing the parsing issue and standardizing the column names using `janitor`)

```
mcamp_2014 <- mcamp_2014|>
  row_to_names(row_number = 1)
```

```
mcamp_2014 <- clean_names(mcamp_2014)
```

```
head(mcamp_2014)
```

```
# A tibble: 6 x 13
  contributors_name  contributors_address contributors_postal_~ contribution_am~
  <chr>              <chr>                <chr>                 <chr>
1 A D'Angelo, Tullio <NA>                 M6A 1P5               300
2 A Strazar, Martin  <NA>                 M2M 3B8               300
3 A'Court, K Susan   <NA>                 M4M 2J8               36
4 A'Court, K Susan   <NA>                 M4M 2J8               100
5 A'Court, K Susan   <NA>                 M4M 2J8               100
6 Aaron, Robert B    <NA>                 M6B 1H7               250
# ... with 9 more variables: contribution_type_desc <chr>,
#   goods_or_service_desc <chr>, contributor_type_desc <chr>,
#   relationship_to_candidate <chr>, president_business_manager <chr>,
#   authorized_representative <chr>, candidate <chr>, office <chr>, ward <chr>
```

4. Summarize the variables in the dataset. Are there missing values, and if so, should we be worried about them? Is every variable in the format it should be? If not, create new variable(s) that are in the right format.

```
skim(mcamp_2014)
```

Table 1: Data summary

Name	mcamp_2014

Table 1: Data summary

Number of rows	10199
Number of columns	13
_____	
Column type frequency:	
character	13
_____	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
contributors_name	0	1	4	31	0	7545	0
contributors_address	10197	0	24	26	0	2	0
contributors_postal_code	0	1	7	7	0	5284	0
contribution_amount	0	1	1	18	0	209	0
contribution_type_desc	0	1	8	14	0	2	0
goods_or_service_desc	10188	0	11	40	0	9	0
contributor_type_desc	0	1	10	11	0	2	0
relationship_to_candidate	10166	0	6	9	0	2	0
president_business_manager	10197	0	13	16	0	2	0
authorized_representative	10197	0	13	16	0	2	0
candidate	0	1	9	18	0	27	0
office	0	1	5	5	0	1	0
ward	10199	0	NA	NA	0	0	0

Majorities of values are missing in 'contributors_address', 'goods_or_service_desc', 'relationship_to_candidate', 'president_business_manager', 'authorized_representative', and 'ward'. However, there are no missing values among the rest of the variables.In particular, there are no missing values in 'contribution_amount'. So. we are safe to analyze its distribution.

We find that contribution amounts are in character values. So, we change them into numeric values.

```
mcamp_2014$contribution_amount <- as.numeric(mcamp_2014$contribution_amount)
```

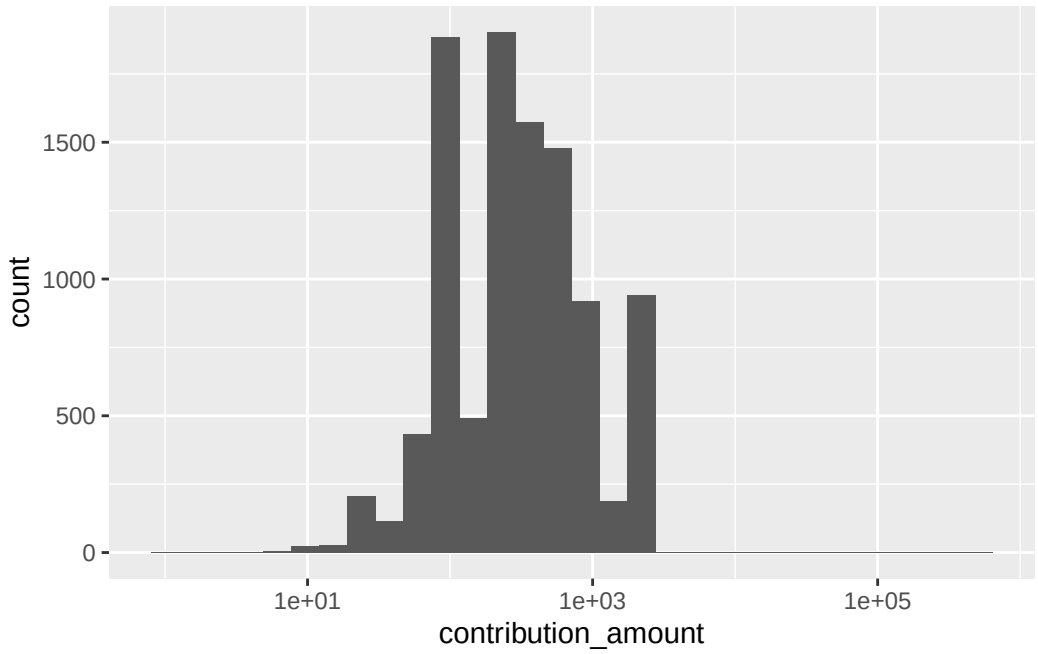
There are some duplicates but it would be also be reasonable to consider that donors make the same amount of contributions multiple times. Since there is no other relevant information such as 'date' that could be used to see if they are duplicates or not, we keep them.

```
get_dupes(mcamp_2014)
```

```
# A tibble: 1,716 x 14
   contributors_name contributors_address contributors_postal_~ contribution_am~
   <chr>             <chr>                <chr>                              <dbl>
 1 A'Court, K Susan  <NA>                 M4M 2J8                              100
 2 A'Court, K Susan  <NA>                 M4M 2J8                              100
 3 Adain, Jacqueline <NA>                 M4C 5N8                              100
 4 Adain, Jacqueline <NA>                 M4C 5N8                              100
 5 Adams, Don        <NA>                 M4L 3A5                               25
 6 Adams, Don        <NA>                 M4L 3A5                               25
 7 Adams, Don        <NA>                 M4L 3A5                               25
 8 Adams, Marion     <NA>                 K0C 2K0                              300
 9 Adams, Marion     <NA>                 K0C 2K0                              300
10 Agnew, Arel       <NA>                 M6G 1V2                              100
# ... with 1,706 more rows, and 10 more variables:
#   contribution_type_desc <chr>, goods_or_service_desc <chr>,
#   contributor_type_desc <chr>, relationship_to_candidate <chr>,
#   president_business_manager <chr>, authorized_representative <chr>,
#   candidate <chr>, office <chr>, ward <chr>, dupe_count <int>
```

5. Visually explore the distribution of values of the contributions. What contributions are notable outliers? Do they share a similar characteristic(s)? It may be useful to plot the distribution of contributions without these outliers to get a better sense of the majority of the data.

```
ggplot(data = mcamp_2014)+
  geom_histogram(aes(x=contribution_amount))+
  scale_x_log10()
```



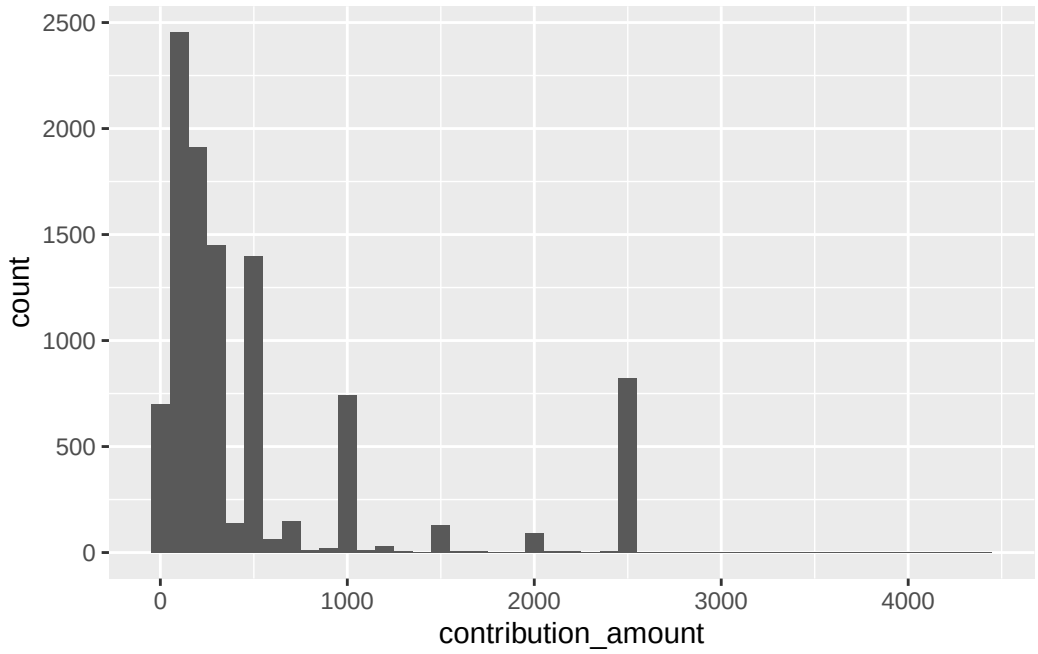
We find there are a few outliers. Indeed there are only 9 contributions out of total 10199 contributions which are greater than 2500.

```
sum(mcamp_2014$contribution_amount>5000)
```

```
[1] 9
```

Shown below is the distribution of the amount of contribution without those outliers:

```
mcamp_2014 |>
  filter(contribution_amount<=5000)|>
  ggplot(aes(x=contribution_amount))+
  geom_histogram(binwidth=100)
```



6. List the top five candidates in each of these categories:

+ total contributions

```
mcamp_2014|>
  group_by(candidate)|>
  summarize(tot_cont = sum(contribution_amount))|>
  arrange(-tot_cont)|>
  slice(1:5)
```

```
# A tibble: 5 x 2
  candidate     tot_cont
  <chr>            <dbl>
1 Tory, John    2767869.
2 Chow, Olivia  1638266.
3 Ford, Doug     889897.
4 Ford, Rob      387648.
5 Stintz, Karen  242805
```

+ mean contribution

```r
mcamp_2014|>
  group_by(candidate)|>
  summarize(mean_cont = mean(contribution_amount))|>
  arrange(-mean_cont)|>
  slice(1:5)
```

```
# A tibble: 5 x 2
  candidate        mean_cont
  <chr>                <dbl>
1 Sniedzins, Erwin     2025
2 Syed, Hïmy           2018
3 Ritch, Carlie        1887.
4 Ford, Doug           1456.
5 Clarke, Kevin        1200
```

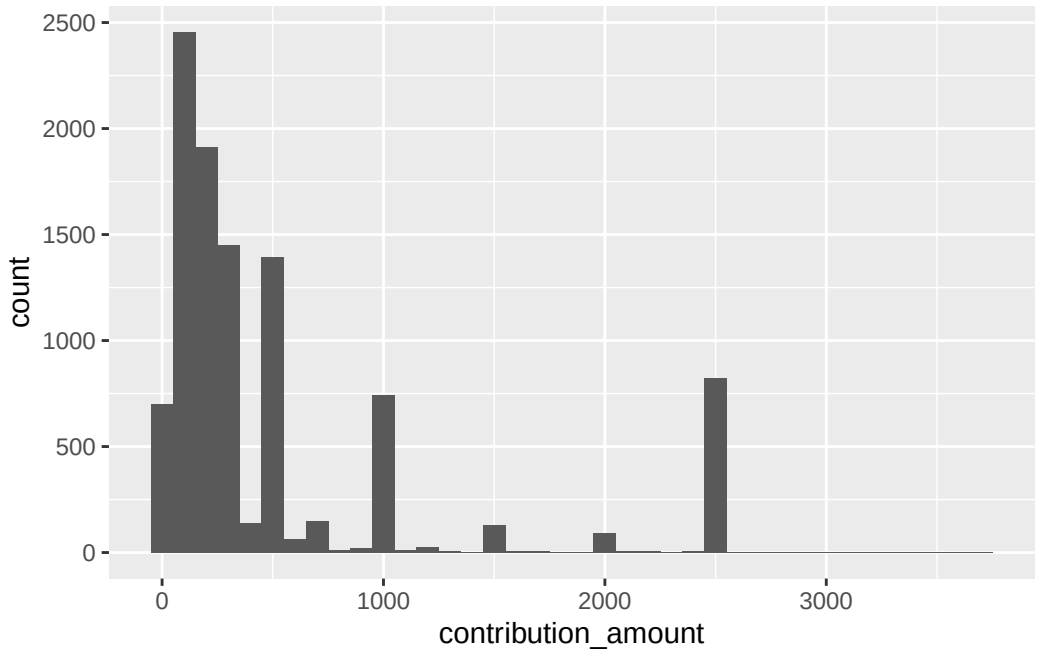
+ number of contributions

```r
mcamp_2014|>
  group_by(candidate)|>
  summarize(num_cont = sum(contribution_amount>0))|>
  arrange(-num_cont)|>
  slice(1:5)
```

```
# A tibble: 5 x 2
  candidate       num_cont
  <chr>              <int>
1 Chow, Olivia        5708
2 Tory, John          2602
3 Ford, Doug           611
4 Ford, Rob            538
5 Soknacki, David      314
```

7. Repeat 5 but without contributions from the candidates themselves.

```r
mcamp_2014 |>
  filter(contributors_name!=candidate)|>
  ggplot(aes(x=contribution_amount))+
  geom_histogram(binwidth=100)
```



From the histogram, there seems to be almost no outliers with contribution amount > 5000. Indeed, there is none contribution over 5000.

```
mcamp_2014 |>
  filter(contribution_amount>5000)|>
  filter(contributors_name!=candidate)
```

```
# A tibble: 0 x 13
# ... with 13 variables: contributors_name <chr>, contributors_address <chr>,
#   contributors_postal_code <chr>, contribution_amount <dbl>,
#   contribution_type_desc <chr>, goods_or_service_desc <chr>,
#   contributor_type_desc <chr>, relationship_to_candidate <chr>,
#   president_business_manager <chr>, authorized_representative <chr>,
#   candidate <chr>, office <chr>, ward <chr>
```

From these observations, we can see that those outlier contributions are made by candidates themselves.

In addition, there are only 31 comtributions out of 11100 contributions made by candidates themselves. Therefore, we can conclude that there are little difference between the distributions of contributions less than 5000 with or without contributions by candidates themselves.

```
sum(mcamp_2014$contributors_name==mcamp_2014$candidate)
```

```
[1] 31
```

8. How many contributors gave money to more than one candidate?

```
n_distinct(
  mcamp_2014|>
  group_by(contributors_name)|>
  summarize(num_cand = n_distinct(candidate))|>
  filter(num_cand>1)|>
  select(contributors_name)
)
```

```
[1] 184
```