Delete this section once you've followed these instructions

1. Change 'Your name, your ID' in line 9 above to be your name and ID. No quotes needed.
2. Run the `setup` and `getdata` chunks below. (You can click the green play button at the top right of these chunks.)
3. Click Knit to test that you can run correctly knit this file.
4. Delete this section, up to the first code chunk. I.e. delete the header, "Delete this section once you've followed these instructions", and points 1 through 4. *Don't* delete the `setup` code chunk.

Code last run 2021-02-16.
Daily: Data as of January 29, 2021.
Neighbourhood: Data as of January 28, 2021.

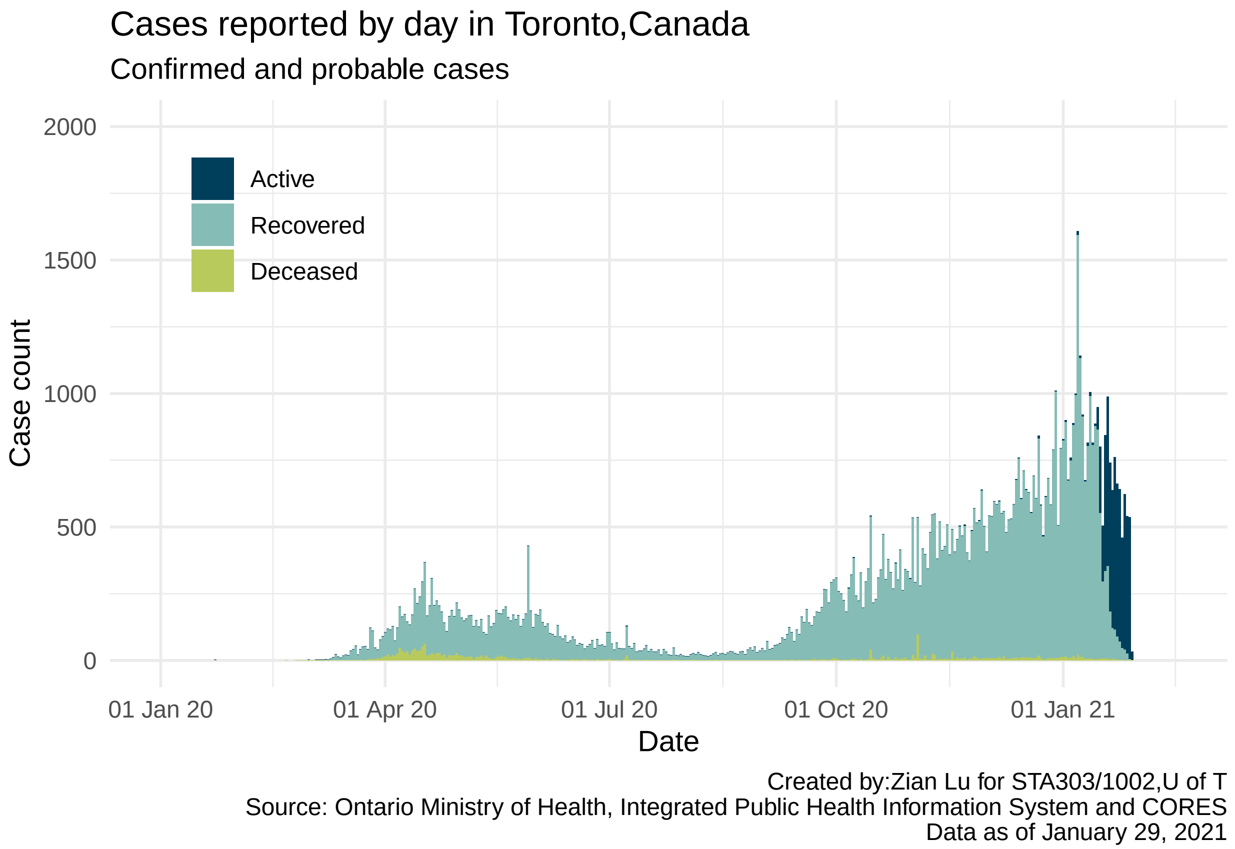
Task 1: Daily cases

Data wrangling

```
reported <- reported_raw %>%
  mutate_if(is.numeric, replace_na, replace = 0) %>%
  mutate (reported_date = date(reported_date))%>%
  pivot_longer(-c(reported_date),names_to= "types",values_to="cases") %>%
  mutate(types=str_to_sentence(types)) %>%
  mutate(types = fct_relevel(types, "Active","Recovered","Deceased",after =0))
```

Data visualization

```r
reported %>%
  ggplot(aes(x=reported_date,y=cases,fill=types))+
  geom_bar(stat="identity")+
  scale_x_date(labels = scales::date_format("%d %b %y"),
               limits = c(date("2020-01-01"), Sys.Date()))+
  theme_minimal()+
  labs(title = "Cases reported by day in Toronto,Canada",
  subtitle="Confirmed and probable cases",
  x = "Date",
  y = "Case count",
  caption = str_c("Created by:Zian Lu for STA303/1002,U of T\n",
  "Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
                  date_daily[1,1]))+
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8))+
  scale_y_continuous(limits = c(0, 2000), breaks = seq(0, 2000, by = 500)) +
  scale_fill_manual(values=c("#003F5C", "#86BCB6","#B9CA5D"))
```



Cases reported by day in Toronto,Canada

Confirmed and probable cases

Created by:Zian Lu for STA303/1002,U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 29, 2021

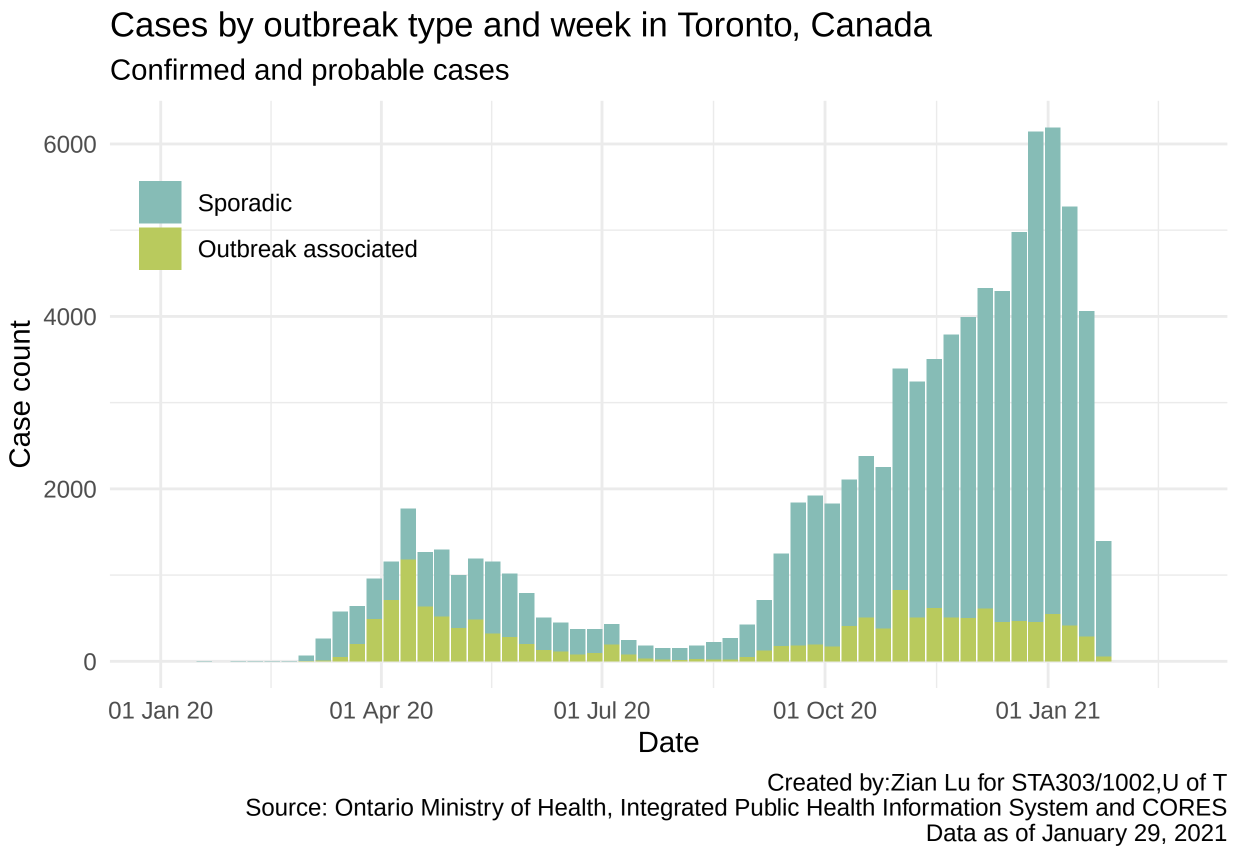
Task 2: Outbreak type

Data wrangling

```
outbreak <- outbreak_raw %>%
  mutate (episode_week = date(episode_week)) %>%
  mutate(outbreak_or_sporadic=str_replace(outbreak_or_sporadic,
                                          "OB Associated","Outbreak associated")) %>%
  mutate(outbreak_or_sporadic = fct_relevel(outbreak_or_sporadic,
                                            "Sporadic","Outbreak associated",after =0)) %>%
  group_by(episode_week) %>%
  mutate(total_cases=sum(cases))
```

Data visualization

```
outbreak %>%
  ggplot(aes(x=episode_week,y=cases,fill=outbreak_or_sporadic))+
  geom_bar(stat="identity")+
  scale_x_date(labels = scales::date_format("%d %b %y"),
               limits = c(date("2020-01-01"), Sys.Date()+7))+
  theme_minimal()+
  labs(title = "Cases by outbreak type and week in Toronto, Canada",
  subtitle="Confirmed and probable cases",
  x = "Date",
  y = "Case count",
  caption = str_c("Created by:Zian Lu for STA303/1002,U of T\n",
  "Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
                  date_daily[1,1]))+
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8))+
  scale_y_continuous(limits = c(0, max(outbreak$total_cases)),
                     breaks = seq(0, max(outbreak$total_cases), by = 2000)) +
  scale_fill_manual(values=c("#86BCB6", "#B9CA5D"))
```



Task 3: Neighbourhoods

Data wrangling: part 1

```
income <- nbhood_profile %>%
  filter(Characteristic ==  "  18 to 64 years (%)") %>%
  filter(`_id` ==1143) %>%
  select(-1:-6) %>%
  pivot_longer(everything(),names_to="neighbourhood_name",values_to="low_income") %>%
  mutate(low_income=parse_number(low_income))
```

Data wrangling: part 2

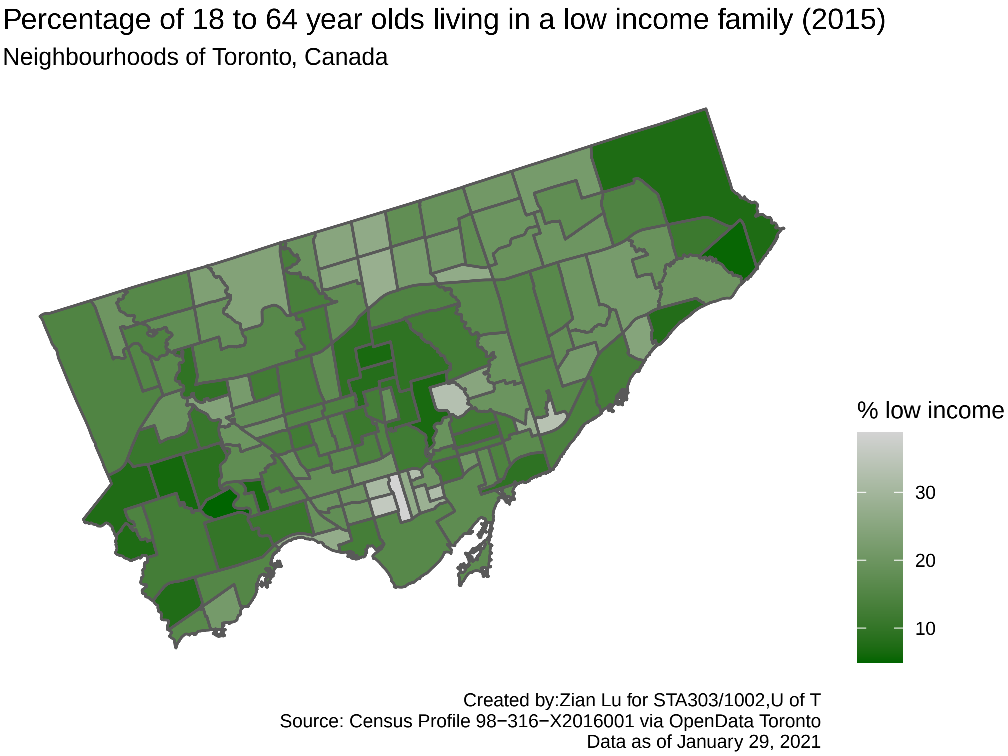
```
nbhoods_all <- nbhoods_shape_raw %>%
  mutate(neighbourhood_name = str_remove(AREA_NAME,"\\s\\(\\d+\\)$")) %>%
  mutate(neighbourhood_name=str_replace(neighbourhood_name,
                                        "Weston-Pellam Park","Weston-Pelham Park")) %>%
  mutate(neighbourhood_name=str_replace(neighbourhood_name,
                                        "St.James Town","St. James Town")) %>%
  left_join(nbhood_raw,by = "neighbourhood_name") %>%
  left_join(income,by = "neighbourhood_name") %>%
  mutate(rate_per_100000=rate_per_100_000_people) %>%
  select(-21)
```

Data wrangling: part 3

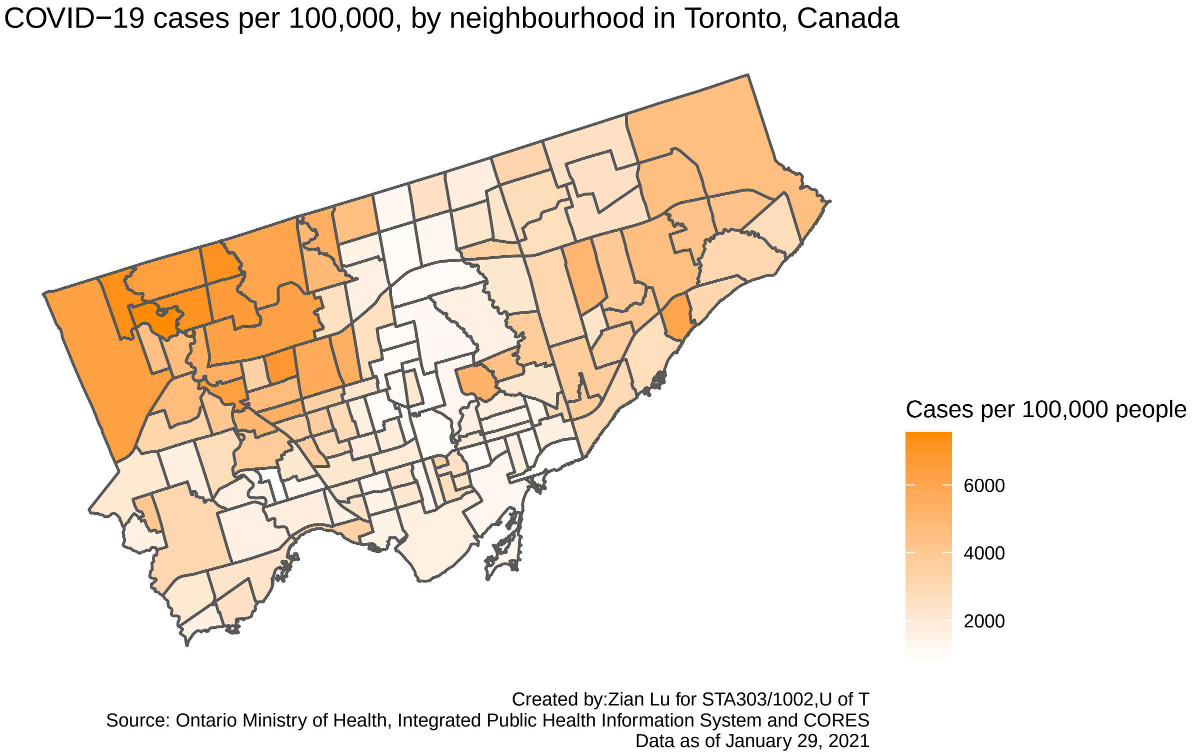
```
nbhoods_final<- nbhoods_all %>%
  mutate(med_inc=median(low_income)) %>%
  mutate(med_rate=median(rate_per_100000)) %>%
  mutate(nbhoods_type=case_when(
    low_income >= med_inc & rate_per_100000 >=med_rate ~"Higher low income rate, higher case rate",
    low_income >= med_inc & rate_per_100000 <med_rate ~"Higher low income rate, lower case rate",
    low_income < med_inc & rate_per_100000 >=med_rate ~"Lower low income rate, higher case rate",
    low_income < med_inc & rate_per_100000 <med_rate ~"Lower low income rate, lower case rate"))
```

Data visualization

```
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = low_income)) +
theme_map() +
theme(legend.position ="right",)+
labs(title = "Percentage of 18 to 64 year olds living in a low income family (2015)",
subtitle="Neighbourhoods of Toronto, Canada",
caption = str_c("Created by:Zian Lu for STA303/1002,U of T\n",
                "Source: Census Profile 98-316-X2016001 via OpenData Toronto\n",
                date_daily[1,1]))+
scale_fill_gradient(name="% low income", low = "darkgreen", high = "lightgrey")
```


Percentage of 18 to 64 year olds living in a low income family (2015)
Neighbourhoods of Toronto, Canada
% low income
30
20
10
Created by:Zian Lu for STA303/1002,U of T
Source: Census Profile 98−316−X2016001 via OpenData Toronto
Data as of January 29, 2021

```r
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = rate_per_100000)) +
theme_map() +
theme(legend.position ="right",)+
labs(title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",
caption = str_c("Created by:Zian Lu for STA303/1002,U of T\n",
"Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
date_daily[1,1]))+
scale_fill_gradient(name="Cases per 100,000 people", low = "white", high = "darkorange")
```

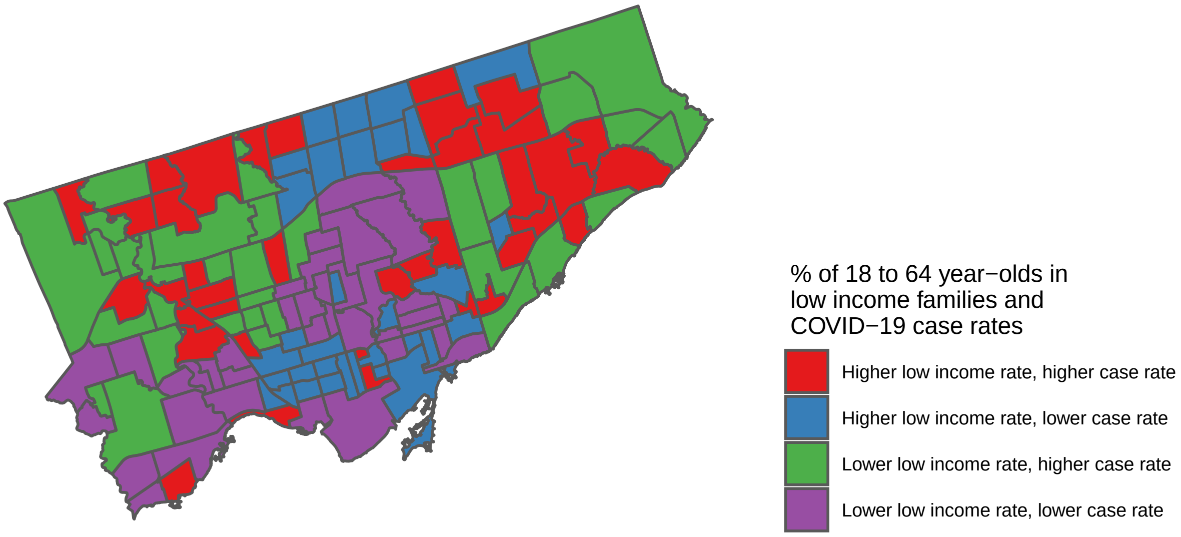


COVID−19 cases per 100,000, by neighbourhood in Toronto, Canada

Created by:Zian Lu for STA303/1002,U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 29, 2021

```r
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = nbhoods_type)) +
theme_map() +
theme(legend.position ="right")+
labs(title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",
caption = str_c("Created by:Zian Lu for STA303/1002,U of T\n",
"Income data source: Census Profile 98-316-X2016001 via OpenData Toronto\n",
"COVID data source: Ontario Ministry of Health, Integrated Public\n",
                "Health Information System and CORES\n",date_daily[1,1]))+
scale_fill_brewer(name=" % of 18 to 64 year-olds in\n low income families and\n COVID-19 case rates",
                  palette = "Set1")
```

COVID−19 cases per 100,000, by neighbourhood in Toronto, Canada


% of 18 to 64 year−olds in
low income families and
COVID−19 case rates
🟥 Higher low income rate, higher case rate
🟦 Higher low income rate, lower case rate
🟩 Lower low income rate, higher case rate
🟪 Lower low income rate, lower case rate

Created by:Zian Lu for STA303/1002,U of T
Income data source: Census Profile 98−316−X2016001 via OpenData Toronto
COVID data source: Ontario Ministry of Health, Integrated Public
Health Information System and CORES
Data as of January 29, 2021

```r
# This chunk of code helps you prepare your assessment for submission on Crowdmark
# This is optional. If it isn't working, you can do it manually/take another approach.

# Run this chunk by hand after knitting your final version of your pdf for submission.
# A new file called 'to_submit' will appear in your working directory with each page of your assignment

# Install the required packages
if(!match("staplr", installed.packages()[,1], nomatch = FALSE))
  {install.packages("staplr")}

# Don't edit anything in this function
prep_for_crowdmark <- function(pdf=NULL){
  # Get the name of the file you're currently in.
  this_file <- rstudioapi::getSourceEditorContext()$path
  pdf_name <- sub(".Rmd", ".pdf", sub('.*/', '', this_file))

  # Create a file called to_submit to put the individual files in
  # This will be in the same folder as this file is saved
  if(!match("to_submit", list.files(), nomatch = FALSE))
    {dir.create("to_submit")}

  # Split the files
  if(is.null(pdf)){
  staplr::split_pdf(pdf_name, output_directory = "to_submit", prefix = "page_")} else {
    staplr::split_pdf(pdf, output_directory = "to_submit", prefix = "page_")
  }
}

prep_for_crowdmark()
```