R Notebook

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
# Read in data: row: city, column: date

#getwd()
#setwd('DATA 599 Capstone/RT-LBCI')

data <- read_csv('RTLBCI/RTLBCI_Cleaned_To_2023-04-28.csv')
```

```
## Rows: 30 Columns: 142
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (2): City_Provience, March 21, 2022
## dbl (140): August 10, 2020, August 17, 2020, August 24, 2020, August 31, 202...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
date_cols <- colnames(data)[2:ncol(data)] # Get the column names of the date columns
data[date_cols] <- lapply(data[date_cols], as.character) # Convert all date columns to character

data_long <- data %>%
  pivot_longer(cols = -1,    # Select all columns except the first one
               names_to = "Date",    # Name the new 'Date' column
               values_to = "Value") # Name the new 'Value' column

data_wide <- data_long %>%

  # Fill the new city columns with the 'Value' column
  pivot_wider(names_from = City_Provience,
              values_from = Value)

# Adjust the format according to your date format
```

```
data_wide$Date <- as.Date(data_wide$Date, format = "%B %d, %Y")

data_wide <- data_wide %>%
  mutate_at(vars(-1), as.numeric)
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Moncton, New Brunswick (0539) 1 =
##   .Primitive("as.double")(`Moncton, New Brunswick (0539) 1`)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
# write_csv(data_wide, "RTLBCI_reshaped_data.csv")

# Rename data_wide to data
data <- data_wide

data
```

```
## # A tibble: 141 x 31
##    Date       `St. John's, Newfoundland (792) 1` `Halifax, Nova Scotia (0348) 1`
##    <date>                                  <dbl>                           <dbl>
##  1 2020-08-10                               100                            100
##  2 2020-08-17                               112.                           103.
##  3 2020-08-24                               123.                           104.
##  4 2020-08-31                               116.                            92.1
##  5 2020-09-07                               118.                           111.
##  6 2020-09-14                               149.                           108.
##  7 2020-09-21                               140.                           101.
##  8 2020-09-28                               195.                           111.
##  9 2020-10-05                               223.                           126.
## 10 2020-10-12                               220.                           124.
## # i 131 more rows
## # i 28 more variables: `Moncton, New Brunswick (0539) 1` <dbl>,
## #   `Quebec, Quebec (0685) 1` <dbl>, `Trois-Rivieres, Quebec (0953) 1` <dbl>,
## #   `Sherbrooke, Quebec (0758) 1` <dbl>, `Montreal, Quebec (0547) 1` <dbl>,
## #   `Ottawa, Ontario (0616) 1` <dbl>, `Hamilton, Ontario (0349) 1` <dbl>,
## #   `Kitchener, Ontario (0419) 1` <dbl>,
## #   `St. Catharines - Niagara Falls, Ontario (0788) 1` <dbl>, ...
```

```
# DATA: row: date, column: city

# read in data
df <- read_csv('RTLBCI/RTLBCI_row_date.csv')
```

```
## Rows: 141 Columns: 31
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): Geography, Moncton, New Brunswick (0539) 1
## dbl (29): St. John's, Newfoundland (792) 1, Halifax, Nova Scotia (0348) 1, Q...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df$Geography <- as.Date(df$Geography, format = "%d-%b-%y")

# Rename Geography to Date
colnames(df)[1] <- "Date"

df
```

```
## # A tibble: 141 x 31
##    Date       `St. John's, Newfoundland (792) 1` `Halifax, Nova Scotia (0348) 1`
##    <date>                                  <dbl>                           <dbl>
##  1 2020-08-10                               100                            100
##  2 2020-08-17                               112.                           103.
##  3 2020-08-24                               123.                           104.
##  4 2020-08-31                               116.                            92.1
##  5 2020-09-07                               118.                           111.
##  6 2020-09-14                               149.                           108.
##  7 2020-09-21                               140.                           101.
##  8 2020-09-28                               195.                           111.
##  9 2020-10-05                               223.                           126.
## 10 2020-10-12                               220.                           124.
## # i 131 more rows
## # i 28 more variables: `Moncton, New Brunswick (0539) 1` <chr>,
## #   `Quebec, Quebec (0685) 1` <dbl>, `Trois-Rivieres, Quebec (0953) 1` <dbl>,
## #   `Sherbrooke, Quebec (0758) 1` <dbl>, `Montreal, Quebec (0547) 1` <dbl>,
## #   `Ottawa, Ontario (0616) 1` <dbl>, `Hamilton, Ontario (0349) 1` <dbl>,
## #   `Kitchener, Ontario (0419) 1` <dbl>,
## #   `St. Catharines - Niagara Falls, Ontario (0788) 1` <dbl>, ...
```

```
# Gas retail prices

# read in data
df_gas <- read_csv('other_datasets/gas_retail_price.csv')
```

```
## Rows: 32 Columns: 20
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): Geography
## dbl (19): Canada, St. John's, Newfoundland and Labrador, Charlottetown and S...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df_gas$Geography <- paste0("01-", df_gas$Geography)
df_gas$Geography <- as.Date(df_gas$Geography, format = "%d-%b-%y")
colnames(df_gas)[1] <- "Date"


df_gas
```

```
## # A tibble: 32 x 20
##    Date       Canada St. John's, Newfoundland and Labra~1 Charlottetown and Su~2
##    <date>      <dbl>                                <dbl>                  <dbl>
```

```
##  1 2020-08-01   105.                                  107.                   99.5
##  2 2020-09-01   104.                                  107.                   98.8
##  3 2020-10-01   103.                                  110.                   99.6
##  4 2020-11-01   101.                                  112                    95.8
##  5 2020-12-01   104.                                  117                   102.
##  6 2021-01-01   111                                   125.                  109.
##  7 2021-02-01   118.                                  131.                  115.
##  8 2021-03-01   125                                   142.                  125.
##  9 2021-04-01   128.                                  141.                  126.
## 10 2021-05-01   132.                                  144.                  130.
## # i 22 more rows
## # i abbreviated names: 1: `St. John's, Newfoundland and Labrador`,
## #   2: `Charlottetown and Summerside, Prince Edward Island`
## # i 16 more variables: `Halifax, Nova Scotia` <dbl>,
## #   `Saint John, New Brunswick` <dbl>, `Québec, Quebec` <dbl>,
## #   `Montréal, Quebec` <dbl>,
## #   `Ottawa-Gatineau, Ontario part, Ontario/Quebec 2` <dbl>, ...
```

```
# CPI all items

# read in data
df_cpi <- read_csv('other_datasets/CPI_all_items.csv')
```

```
## Rows: 64 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): Geography, Vancouver, British Columbia 5, Victoria, British Columb...
## dbl (13): Canada, Newfoundland and Labrador, Prince Edward Island, Nova Scot...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df_cpi$Geography <- paste0("01-", df_cpi$Geography)
df_cpi$Geography <- as.Date(df_cpi$Geography, format = "%d-%b-%y")
colnames(df_cpi)[1] <- "Date"

df_cpi <- df_cpi %>%
  mutate_at(vars(-1), as.numeric)
```

```
## Warning: There were 3 warnings in `mutate()`.
## The first warning was:
## i In argument: `Vancouver, British Columbia 5 =
##   .Primitive("as.double")(`Vancouver, British Columbia 5`)`.
## Caused by warning:
## ! NAs introduced by coercion
## i Run `dplyr::last_dplyr_warnings()` to see the 2 remaining warnings.
```

```
df_cpi <- na.omit(df_cpi)
df_cpi
```

```
## # A tibble: 32 x 17
##    Date       Canada Newfoundland and Lab~1 `Prince Edward Island` `Nova Scotia`
```

```
##    <date>      <dbl>                 <dbl>                 <dbl>         <dbl>
##  1 2020-08-01   137                   139.                  138.          138.
##  2 2020-09-01   137.                  139.                  138.          138.
##  3 2020-10-01   138.                  140.                  139.          138.
##  4 2020-11-01   138.                  141                   138.          139.
##  5 2020-12-01   137.                  140.                  138.          139.
##  6 2021-01-01   138.                  142.                  140.          140.
##  7 2021-02-01   139.                  142                   141.          141.
##  8 2021-03-01   140.                  143.                  142.          142.
##  9 2021-04-01   140.                  144.                  143.          142.
## 10 2021-05-01   141                   144.                  144.          143.
## # i 22 more rows
## # i abbreviated name: 1: `Newfoundland and Labrador`
## # i 12 more variables: `New Brunswick` <dbl>, Quebec <dbl>, Ontario <dbl>,
## #   Manitoba <dbl>, Saskatchewan <dbl>, Alberta <dbl>,
## #   `British Columbia` <dbl>, `Vancouver, British Columbia 5` <dbl>,
## #   `Victoria, British Columbia 5` <dbl>, `Whitehorse, Yukon 6` <dbl>,
## #   `Yellowknife, Northwest Territories 6` <dbl>, ...
```

```
# Housing completions

# read in data
df_housing <- read_csv('other_datasets/housing_completion_van.csv')
```

```
## Rows: 29 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Type of unit
## num (1): Total units
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df_housing$`Type of unit` <- paste0("01-", df_housing$`Type of unit`)
df_housing$`Type of unit` <- as.Date(df_housing$`Type of unit`, format = "%d-%b-%y")
colnames(df_housing)[1] <- "Date"

df_housing
```

```
## # A tibble: 29 x 2
##    Date       `Total units`
##    <date>             <dbl>
##  1 2020-08-01          2069
##  2 2020-09-01          1894
##  3 2020-10-01          1797
##  4 2020-11-01          1952
##  5 2020-12-01          1816
##  6 2021-01-01          1553
##  7 2021-02-01           964
##  8 2021-03-01          1905
##  9 2021-04-01          2200
## 10 2021-05-01          1256
## # i 19 more rows
```

```r
# New housing price index

# read in data
df_housing_price <- read_csv('other_datasets/New housing price index_van.csv')
```

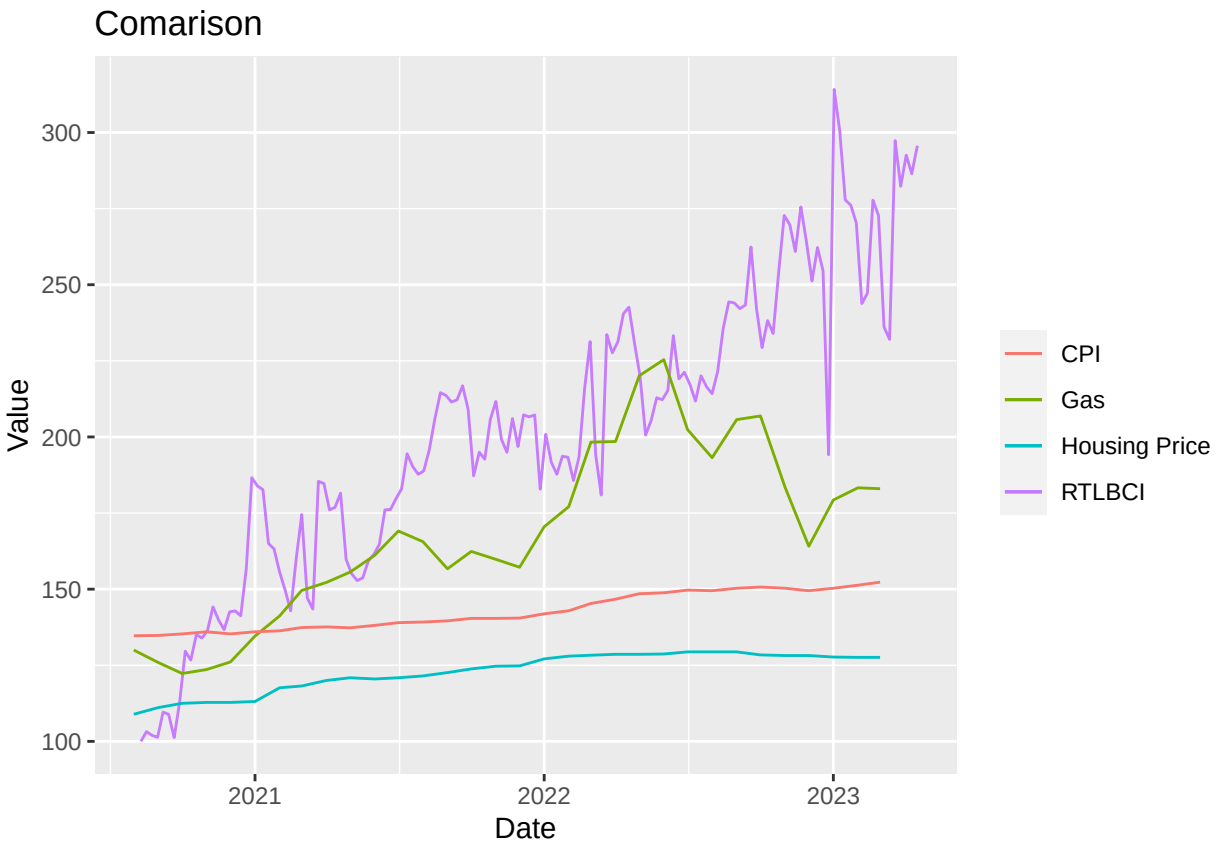
```
## Rows: 42 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): New housing price indexes
## dbl (1): Total (house and land)
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
df_housing_price$`New housing price indexes` <- paste0("01-", df_housing_price$`New housing price index
df_housing_price$`New housing price indexes` <- as.Date(df_housing_price$`New housing price indexes`, f
colnames(df_housing_price)[1] <- "Date"

df_housing_price <- na.omit(df_housing_price)
df_housing_price
```

```
## # A tibble: 32 x 2
##    Date       `Total (house and land)`
##    <date>                        <dbl>
##  1 2020-08-01                     109.
##  2 2020-09-01                     111.
##  3 2020-10-01                     112.
##  4 2020-11-01                     113.
##  5 2020-12-01                     113.
##  6 2021-01-01                     113.
##  7 2021-02-01                     118.
##  8 2021-03-01                     118.
##  9 2021-04-01                     120
## 10 2021-05-01                     121.
## # i 22 more rows
```

```r
# Visualize the data

# RTLBCI
#colnames(data)

# Rename the column names
#colnames(data)
colnames(data)[30] <- "Vancouver"

#colnames(df_gas)
colnames(df_gas)[17] <- "Vancouver"

#colnames(df_cpi)
colnames(df_cpi)[13] <- "Vancouver"

#colnames(df_housing)
```

```r
colnames(df_housing)[2] <- "Vancouver"

#colnames(df_housing_price)
colnames(df_housing_price)[2] <- "Vancouver"



ggplot() +
  geom_line(data = data, aes(x = Date, y = Vancouver, color = "RTLBCI")) +
  geom_line(data = df_gas, aes(x = Date, y = Vancouver, color = "Gas")) +
  geom_line(data = df_cpi, aes(x = Date, y = Vancouver, color = "CPI")) +
  geom_line(data = df_housing_price, aes(x = Date, y = Vancouver, color = "Housing Price")) +
  labs(title = "Comarison",
       x = "Date",
       y = "Value") +
  theme(legend.title = element_blank())
```




```r
ggplot() +
  geom_line(data = data, aes(x = Date, y = Vancouver, color = "RTLBCI")) +
  geom_line(data = df_gas, aes(x = Date, y = Vancouver, color = "Gas")) +
  geom_line(data = df_cpi, aes(x = Date, y = Vancouver, color = "CPI")) +
  geom_line(data = df_housing_price, aes(x = Date, y = Vancouver, color = "Housing Price")) +
  geom_line(data = df_housing, aes(x = Date, y = Vancouver, color = "housing completiion")) +
  labs(title = "Comarison",
       x = "Date",
```

```
      y = "Value") +
theme(legend.title = element_blank())
```