

lcbo

August 4, 2024

1 LCBO Web Scraping

1.0.1 Imports

```
[182]: from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.chrome.service import Service
from selenium.common.exceptions import NoSuchElementException, \
↳ ElementClickInterceptedException
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from bs4 import BeautifulSoup
import time
import re
import requests
import pandas as pd
```

1.0.2 Setting up web page and downloading HTML

```
[20]: service = Service(r'C:\Users\Matthew\Documents\chromedriver-win64\chromedriver.
↳ exe')
driver = webdriver.Chrome(service = service)

url = "https://www.lcbo.com/en/products/beer-cider/"
pages = ["lager", "ale", "cider"]

for page in pages:
    driver.get(url+page)
    while True:
        time.sleep(4)
        driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
        load_more_button = driver.find_element(By.ID, "loadMore")
        if load_more_button.is_displayed():
            load_more_button.click()
        else:
            break
```

```

html_content = driver.page_source
with open(f"page_content_{page}.html", "w", encoding="utf-8") as file:
    file.write(html_content)

driver.quit()

```

1.0.3 Parsing HTML for links

```

[26]: html_contents = []

for page in pages:
    with open(f"page_content_{page}.html", "r", encoding="utf-8") as file:
        html_contents.append(file.read())

```

```

[30]: soups = []

for content in html_contents:
    soups.append(BeautifulSoup(content, "html.parser"))

```

```

[52]: elements_list = []

for soup in soups:
    elements_list.append(soup.find_all(class_ = "coveo-result-list-container"))

```

```

[82]: url_prefix = "https://www.lcbo.com/en/"
hrefs = []

for element_list in elements_list:
    for element in element_list:
        links = element.find_all("a", href = re.compile(f"^{url_prefix}(?!.\
↵*#)"), class_ = "CoveoResultLink")
        hrefs.append([a["href"] for a in links if "href" in a.attrs])

hrefs.pop(4)
hrefs.pop(2)
hrefs.pop(0);

```

```

[123]: lagers = hrefs[0][1::2]
ales = hrefs[1][1::2]
cidars = hrefs[2][1::2]

```

1.0.4 Web scraping from individual product pages

```
[144]: def getData(beer_type, hrefs):
    data = []
    for href in hrefs:
        response = requests.get(href)
        response_string = response.text
        soup = BeautifulSoup(response_string, "html.parser")

        try:
            name = soup.find("span", itemprop = "name").text
        except:
            name = None

        try:
            price = soup.find("meta", itemprop = "price").get("content")
        except:
            price = None

        try:
            info = soup.find("div", id = "moredetail").find_all("div", class_ = "
↪"value")
            abv = info[0].text
            if len(info) > 1:
                made_in = info[1].text
            else:
                made_in = None
        except:
            abv = None
            made_in = None

        try:
            quantity_ml_container = soup.find("div", class_ = "
↪"lcbo-product-size").find("span").text.split()
            if len(quantity_ml_container) == 5:
                quantity = quantity_ml_container[0]
                ml = quantity_ml_container[2]
                container = quantity_ml_container[4]
            elif len(quantity_ml_container) == 3:
                quantity = 1
                ml = quantity_ml_container[0]
                container = quantity_ml_container[2]
        except:
            quantity = None
            ml = None
            container = None
```

```

        data.append([name, price, quantity, ml, abv, container, beer_type,
↪made_in])

    return data

```

```
[307]: lager_data = getData("Lager", lagers)
```

```
[308]: ale_data = getData("Ale", ales)
```

```
[309]: cider_data = getData("Cider", ciders)
```

1.0.5 Cleaning data and handling missing information

Manual data entry in this case is questionable, but given the somewhat low amount of missing data and the time it would require to automate data collection from another source, entering the data manually is overall faster.

```
[312]: def missingInformationReport(data):
        for index, row in enumerate(data):
            if None in row:
                print(f"Row {index}: {row}\n")

```

```
[313]: missingInformationReport(lager_data)
```

```
Row 56: ['Old Style Pilsner', '2.35', None, None, '  Canada ', None, 'Lager',
None]
```

```
Row 355: ['Muskoka Brewery Stein Sized Fest Bier', '3.95', None, None, '  Canada
', None, 'Lager', None]
```

```
Row 364: ['Grolsch Premium Pilsner', None, 1, '500', '5%', 'can', 'Lager', '
Netherlands ']
```

```
Row 516: ['Railway City Merry & Bright Cranberry Lager', '3.45', None, None, '
Canada ', None, 'Lager', 'Railway City']
```

```
[314]: lager_data[56] = ['Old Style Pilsner', '2.35', 1, '473', "5%", "can", 'Lager',
↪None]
lager_data[355] = ['Muskoka Brewery Stein Sized Fest Bier', '3.95', 1, 568, "5.
↪5%", "can", 'Lager', None]
lager_data[364] = ['Grolsch Premium Pilsner', '2.95', 1, '500', '5%', 'can',
↪'Lager', '  Netherlands ']
lager_data[516] = ['Railway City Merry & Bright Cranberry Lager', '3.45', 1,
↪'473', '5.5%', 'can', 'Lager', 'Railway City']

```

```
[315]: missingInformationReport(ale_data)
```

Row 14: ['Sweetwater Ipa', '3.8', None, None, ' United States ', None, 'Ale', 'Sweetwater Brewing']

Row 37: ['Side Launch Passionate Sour', '3.95', None, None, ' Canada ', None, 'Ale', 'Side Launch']

Row 73: ['Lambiek Fabriek Juicy & Wild', '11.6', None, None, None, None, 'Ale', None]

Row 89: ['Henderson's Brewing Co. Pearson Express IPA', '3.65', None, None, ' Canada ', None, 'Ale', 'Henderson Brewing']

Row 171: ['Mill Street Brewery 1 in 12 Phantasm NEIPA', None, 1, '473', '7%', 'can', 'Ale', ' Ontario, Canada ']

Row 221: ['New Ontario Brewing Big & Juicy IPA', '3.55', None, None, ' Canada ', None, 'Ale', 'New Ontario Brewing']

Row 415: [None, None, None, None, None, None, 'Ale', None]

Row 419: ['Village Brewery Binge Watch New Zealand Style Pale Ale', None, 1, '473', '4.5%', 'can', 'Ale', ' Alberta, Canada ']

Row 423: ['Jart-Elle Oude Kriek', None, 1, '375', '6%', 'bottle', 'Ale', ' Belgium ']

Row 424: ['Thornbridge Brewery Jaipur IPA', None, 1, '440', '5.9%', 'can', 'Ale', ' England, United Kingdom ']

Row 425: ['Omnipollo Fatamorgana Folk', None, 1, '473', '3.5%', 'can', 'Ale', ' Ontario, Canada ']

Row 426: ['Rush X Henderson Xanabrew - Belgian Strong Pale Ale', None, 1, '500', '10.5%', 'bottle', 'Ale', ' Ontario, Canada ']

Row 429: ['Siren Caribbean Chocolate Mole Cake Stout', None, 1, '330', '8.8%', 'can', 'Ale', ' England, United Kingdom ']

Row 431: ['Windswept Brewing Wolf Dark & Strong Scottish Ale', None, 1, '330', '6%', 'bottle', 'Ale', ' Scotland, United Kingdom ']

Row 435: ['Clavie Smoked Porter', None, 1, '330', '5.5%', 'bottle', 'Ale', ' Scotland, United Kingdom ']

Row 444: ['Collective Arts Daily Forecast', '3.95', None, None, ' Canada ', None, 'Ale', 'Collective Arts']

Row 482: ['Russell Brewing A Wee Angry Scotch Ale', '3.95', None, None, ' ']

Canada ', None, 'Ale', 'Russell Brewing Company Ltd']

Row 506: ['Highlander Blacksmith Smoked Porter', '3.4', None, None, ' Canada ', None, 'Ale', 'Highlander Brewing']

Row 538: ['Collective Arts Stranger Than Fiction, Porter', '3.45', None, None, ' Canada ', None, 'Ale', 'Collective Arts']

Row 561: ['Dandy Brewing Julia Peach Sour', None, 1, '473', '6.5%', 'can', 'Ale', ' Alberta, Canada ']

Row 597: ['Perth Brewery Mocha Stout', '3.5', None, None, ' Canada ', None, 'Ale', 'Perth Brewery']

Row 609: ['High Park Pumpkin Spice Latte', None, 1, '473', '6%', 'can', 'Ale', ' Ontario, Canada ']

Row 617: ['Lambiek Fabriek - Brett-Elle', None, 1, '375', '6.3%', 'bottle', 'Ale', ' Belgium ']

Row 624: ['Mill Street Tea Time Lemon Wheat', None, 1, '473', '5%', 'can', 'Ale', ' Ontario, Canada ']

Row 628: ['Lake of the Woods Dead Mans Switch NEIPA', '3.45', None, None, ' Canada ', None, 'Ale', 'Lake Of The Woods']

Row 638: ['Brouwerij The Musketeers Troubadour - Obscura', None, 1, '330', '8.2%', 'bottle', 'Ale', ' Belgium ']

Row 642: ['3 Fonteinen Intens Rood', None, 1, '375', '6%', 'bottle', 'Ale', ' Belgium ']

Row 645: ['Rally Beer Company Extra Mile Session IPA', '3.15', None, None, ' Canada ', None, 'Ale', 'Rally Beer Company']

Row 647: ['Caledon Hills Summer Solstice', '3.15', None, None, ' Canada ', None, 'Ale', 'Caledon Hills']

Row 648: ['Black Bellows Beach Freak', '4.45', None, None, ' Canada ', None, 'Ale', 'Black Bellows']

Row 652: ['Nickel Brook Tongue Tied DDH IPA', '3.75', None, None, ' Canada ', None, 'Ale', 'Nickel Brook']

Row 655: ['Henderson x Society of Beer Drinking Ladies Hibiscus Pale Ale', '3.5', None, None, ' Canada ', None, 'Ale', 'Sobdl (Society Of Beer Drinking Ladies)']

Row 656: ['Collingwood Brewery Freestyle Tropiskull Thunder IPA', '3.55', None, None, ' Canada ', None, 'Ale', 'Collingwood Brewery']

Row 659: ['Hendersons Radicle Salted Lime Gose', '3.3', None, None, ' Canada ', None, 'Ale', 'Henderson Brewing Co']

Row 660: ['Nickel Brook Jam Stand Raspberry', '3.85', None, None, ' Canada ', None, 'Ale', 'Nickel Brook']

Row 662: ['Flying Dog Gonzo Imperial Porter', None, 1, '355', '10%', 'bottle', 'Ale', ' Maryland, United States ']

Row 670: ['Harviestoun Brewery Bitter & Twisted', None, 1, '330', '4.2%', 'bottle', 'Ale', ' Scotland, United Kingdom ']

Row 672: ['Equilibrium Brewery Astrophysics IPA', None, 1, '473', '6.5%', 'can', 'Ale', ' New York, United States ']

```
[316]: ale_data[14] = ['Sweetwater Ipa', '3.8', 1, '473', '6.3%', 'can', 'Ale',  
    ↪ 'Sweetwater Brewing']  
ale_data[37] = ['Side Launch Passionate Sour', '3.95', 1, '473', '4.3%', 'can',  
    ↪ 'Ale', 'Side Launch']  
ale_data[73] = ['Lambiek Fabriek Juicy & Wild', '11.60', 1, '750', '8.8%',  
    ↪ 'bottle', 'Ale', None]  
ale_data[89] = ["Henderson's Brewing Co. Pearson Express IPA", '3.65', 1,  
    ↪ '473', '6.5%', 'can', 'Ale', 'Henderson Brewing']  
ale_data[171] = ['Mill Street Brewery 1 in 12 Phantasm NEIPA', '4.50', 1,  
    ↪ '473', '7%', 'can', 'Ale', ' Ontario, Canada ']  
ale_data[221] = ['New Ontario Brewing Big & Juicy IPA', '3.55', 1, '473', '6.  
    ↪ 5%', 'can', 'Ale', 'New Ontario Brewing']  
ale_data[415] = ['Collective Arts Blood Orange Cranberry & Vanilla Sour', '4.  
    ↪ 75', 1, '473', '5.6%', 'can', 'Ale', None]  
ale_data[419] = ['Village Brewery Binge Watch New Zealand Style Pale Ale', '3.  
    ↪ 50', 1, '473', '4.5%', 'can', 'Ale', ' Alberta, Canada ']  
ale_data[423] = ['Jart-Elle Oude Kriek', '9.30', 1, '375', '6%', 'bottle',  
    ↪ 'Ale', ' Belgium ']  
ale_data[424] = ['Thornbridge Brewery Jaipur IPA', '3.60', 1, '440', '5.9%',  
    ↪ 'can', 'Ale', ' England, United Kingdom ']  
ale_data[425] = ['Omnipollo Fatamorgana Folk', '3.85', 1, '473', '3.5%', 'can',  
    ↪ 'Ale', ' Ontario, Canada ']  
ale_data[426] = ['Rush X Henderson Xanabrew - Belgian Strong Pale Ale', '12.  
    ↪ 95', 1, '500', '10.5%', 'bottle', 'Ale', ' Ontario, Canada ']  
ale_data[429] = ['Siren Caribbean Chocolate Mole Cake Stout', '5.55', 1, '330',  
    ↪ '8.8%', 'can', 'Ale', ' England, United Kingdom ']
```

```

ale_data[431] = ['Windswept Brewing Wolf Dark & Strong Scottish Ale', '4.00',
↪1, '330', '6%', 'bottle', 'Ale', ' Scotland,   United Kingdom ']
ale_data[435] = ['Clavie Smoked Porter', '4.00', 1, '330', '5.5%', 'bottle',
↪'Ale', ' Scotland,   United Kingdom ']
ale_data[444] = ['Collective Arts Daily Forecast', '3.95', 1, '355', '4.5%',
↪'can', 'Ale', 'Collective Arts']
ale_data[482] = ['Russell Brewing A Wee Angry Scotch Ale', '3.95', 1, '473', '6.
↪5%', 'can', 'Ale', 'Russell Brewing Company Ltd']
ale_data[506] = ['Highlander Blacksmith Smoked Porter', '3.40', 1, '473', '6.
↪5%', 'can', 'Ale', 'Highlander Brewing']
ale_data[538] = ['Collective Arts Stranger Than Fiction, Porter', '3.45', 1,
↪'473', '5.5%', 'can', 'Ale', 'Collective Arts']
ale_data[561] = ['Dandy Brewing Julia Peach Sour', '4.50', 1, '473', '6.5%',
↪'can', 'Ale', ' Alberta,   Canada ']
ale_data[597] = ['Perth Brewery Mocha Stout', '3.50', 1, '473', '5.5%', 'can',
↪'Ale', 'Perth Brewery']
ale_data[609] = ['High Park Pumpkin Spice Latte', '3.50', 1, '473', '6%',
↪'can', 'Ale', ' Ontario,   Canada ']
ale_data[617] = ['Lambiek Fabriek - Brett-Elle', '11.75', 1, '375', '6.3%',
↪'bottle', 'Ale', ' Belgium ']
ale_data[624] = ['Mill Street Tea Time Lemon Wheat', '3.75', 1, '473', '5%',
↪'can', 'Ale', ' Ontario,   Canada ']
ale_data[628] = ['Lake of the Woods Dead Mans Switch NEIPA', '3.45', 1, '473',
↪'6.8%', 'can', 'Ale', 'Lake Of The Woods']
ale_data[638] = ['Brouwerij The Musketeers Troubadour - Obscura', '4.60', 1,
↪'330', '8.2%', 'bottle', 'Ale', ' Belgium ']
ale_data[642] = ['3 Fonteynen Intens Rood', '20.00', 1, '375', '6%', 'bottle',
↪'Ale', ' Belgium ']
ale_data[645] = ['Rally Beer Company Extra Mile Session IPA', '3.15', 1, '355',
↪'3.5%', 'can', 'Ale', 'Rally Beer Company']
ale_data[647] = ['Caledon Hills Summer Solstice', '3.15', 1, '473', '4.8%',
↪'can', 'Ale', 'Caledon Hills']
ale_data[648] = ['Black Bellows Beach Freak', '4.45', 1, '473', '5.1%', 'can',
↪'Ale', 'Black Bellows']
ale_data[652] = ['Nickel Brook Tongue Tied DDH IPA', '3.75', 1, '473', '5.6%',
↪'can', 'Ale', 'Nickel Brook']
ale_data[655] = ['Henderson x Society of Beer Drinking Ladies Hibiscus Pale
↪Ale', '3.50', 1, '473', '5.5%', 'can', 'Ale', 'Sobdl ( Society Of Beer
↪Drinking Ladies )']
ale_data[656] = ['Collingwood Brewery Freestyle Tropiskull Thunder IPA', '3.
↪55', 1, '473', '6.8%', 'can', 'Ale', 'Collingwood Brewery']
ale_data[659] = ['Hendersons Radicle Salted Lime Gose', '3.30', 1, '473', '4.
↪4%', 'can', 'Ale', 'Henderson Brewing Co']
ale_data[660] = ['Nickel Brook Jam Stand Raspberry', '3.85', 1, '473', '4.0%',
↪'can', 'Ale', 'Nickel Brook']

```



```
ale_data[662] = ['Flying Dog Gonzo Imperial Porter', '5.00', 1, '355', '10%',  
↳ 'bottle', 'Ale', ' Maryland, United States ']  
ale_data[670] = ['Harviestoun Brewery Bitter & Twisted', '3.20', 1, '330', '4.  
↳ 2%', 'bottle', 'Ale', ' Scotland, United Kingdom ']  
ale_data[672] = ['Equilibrium Brewery Astrophysics IPA', '4.95', 1, '473', '6.  
↳ 5%', 'can', 'Ale', ' New York, United States ']
```

```
[317]: missingInformationReport(cider_data)
```

```
Row 58: ['Pearbucha Flavoured Cider', '3.6', None, None, ' Canada ', None,  
'Cider', 'Pombucha']
```

```
Row 59: ['Ardiel Cider House Big John Lightly Hopped Cider', '3.7', None, None,  
' Canada ', None, 'Cider', 'Ardiel Cider']
```

```
Row 114: ["Ernest Cider Winter's Blush", None, 1, '473', '5.7%', 'can', 'Cider',  
' Ontario, Canada ']
```

```
Row 125: ['Brickworks Ciderhouse Mixed Pack', '19.95', None, None, ' Canada ',  
None, 'Cider', 'Brickworks Ciderhouse']
```

```
Row 128: [None, None, None, None, None, None, 'Cider', None]
```

```
[318]: cider_data[58] = ['Pearbucha Flavoured Cider', '3.6', 1, '473', '4.0%', 'can',  
↳ 'Cider', 'Pombucha']  
cider_data[59] = ['Ardiel Cider House Big John Lightly Hopped Cider', '3.70',  
↳ 1, '473', '6.5%', 'can', 'Cider', 'Ardiel Cider']  
cider_data[114] = ["Ernest Cider Winter's Blush", '3.65', 1, '473', '5.7%',  
↳ 'can', 'Cider', ' Ontario, Canada ']  
cider_data[125] = ['Brickworks Ciderhouse Mixed Pack', '19.95', 6, '473', '5.  
↳ 25%', 'can', 'Cider', 'Brickworks Ciderhouse']  
cider_data[128] = ['Seagram Cider', '2.95', 1, '473', '5.3%' , 'can', 'Cider',  
↳ 'will be removed']
```

1.0.6 Data formatting

```
[319]: def formatData(data):  
    for row in data:  
        row[1] = float(row[1])  
        row[2] = int(row[2])  
        row[3] = int(row[3])  
        row[4] = float(row[4][: -1])  
        row[5] = row[5].capitalize()  
        row.pop()
```

```
[320]: formatData(lager_data);
```

```
[321]: formatData(ale_data)
```

```
[322]: formatData(cider_data)
```

1.0.7 Creating pandas dataframe, adding columns, and additional data cleaning

```
[331]: df_lager = pd.DataFrame(lager_data, columns = ['Name', 'Price ($)', 'Quantity', 'Volume (mL)', 'ABV (%)', 'Container', 'Type'])
df_ale = pd.DataFrame(ale_data, columns = ['Name', 'Price ($)', 'Quantity', 'Volume (mL)', 'ABV (%)', 'Container', 'Type'])
df_cider = pd.DataFrame(cider_data, columns = ['Name', 'Price ($)', 'Quantity', 'Volume (mL)', 'ABV (%)', 'Container', 'Type'])

df = pd.concat([df_lager, df_ale, df_cider], ignore_index = True)
```

```
[332]: df
```

```
[332]:
```

| | Name | Price (\$) | Quantity | Volume (mL) | \ |
|------|----------------------------------|------------|----------|-------------|---|
| 0 | Heineken | 17.50 | 6 | 330 | |
| 1 | Modelo Especial | 18.95 | 6 | 355 | |
| 2 | Pilsner Urquell | 3.55 | 1 | 500 | |
| 3 | Corona Extra | 17.50 | 6 | 330 | |
| 4 | Peroni Nastro Azzurro | 16.95 | 6 | 330 | |
| ... | ... | ... | ... | ... | |
| 1401 | Thornbury Craft Cranberry Cider | 3.95 | 1 | 473 | |
| 1402 | Ernest Cider Spiced Cherry | 3.75 | 1 | 473 | |
| 1403 | Seagram Cider | 2.95 | 1 | 473 | |
| 1404 | Tempt Cider No. 9 | 1.85 | 1 | 330 | |
| 1405 | Two Blokes - Hex Press Dry Cider | 4.10 | 1 | 473 | |

| | ABV (%) | Container | Type |
|------|---------|-----------|-------|
| 0 | 5.0 | Bottle | Lager |
| 1 | 4.5 | Bottle | Lager |
| 2 | 4.4 | Can | Lager |
| 3 | 4.6 | Bottle | Lager |
| 4 | 5.2 | Bottle | Lager |
| ... | ... | ... | ... |
| 1401 | 5.3 | Can | Cider |
| 1402 | 5.5 | Can | Cider |
| 1403 | 5.3 | Can | Cider |
| 1404 | 4.5 | Can | Cider |
| 1405 | 6.4 | Can | Cider |

```
[1406 rows x 7 columns]
```

```
[333]: df['Price ($)'] = pd.to_numeric(df['Price ($)'])
df['Quantity'] = pd.to_numeric(df['Quantity'])
df['ABV (%)'] = pd.to_numeric(df['ABV (%)'])
df['Volume (mL)'] = pd.to_numeric(df['Volume (mL)'])
```

```
[334]: df['Cost per mL of Alcohol ($)'] = (df['Price ($)'] / (df['Quantity'] *
↳ (df['ABV (%)'] / 100) * df['Volume (mL)'])).round(3)
```

```
[346]: df.sort_values(by = "Cost per mL of Alcohol ($)")[0:20]
```

```
[346]:
```

| | Name | Price (\$) | Quantity | Volume (mL) | \ |
|------|---------------------------------------|------------|----------|-------------|---|
| 80 | Pabst Blue Ribbon 5.9% | 13.45 | 6 | 473 | |
| 281 | Boxer Ice | 3.25 | 1 | 710 | |
| 29 | Old Milwaukee Ice | 12.95 | 6 | 473 | |
| 526 | Laker Ice | 25.95 | 12 | 473 | |
| 251 | James Ready 6.0 | 2.35 | 1 | 473 | |
| 556 | No Name Beer | 8.55 | 6 | 341 | |
| 481 | Holsten Maibock | 2.95 | 1 | 500 | |
| 147 | Crest Super Lager | 4.20 | 1 | 500 | |
| 491 | Holsten Festbock | 2.95 | 1 | 500 | |
| 88 | DAB Maibock | 2.95 | 1 | 500 | |
| 445 | Faxe 10% Extra Strong | 4.25 | 1 | 500 | |
| 406 | Lowenbrau Original | 12.60 | 6 | 473 | |
| 204 | Old Milwaukee Ice | 3.30 | 1 | 710 | |
| 131 | Busch Ice | 13.45 | 6 | 473 | |
| 267 | Carling Ice | 13.45 | 6 | 473 | |
| 59 | Laker Ice | 13.45 | 6 | 473 | |
| 866 | Amsterdam Fracture Juicy Imperial IPA | 4.25 | 1 | 473 | |
| 359 | Laker Ice | 40.95 | 24 | 355 | |
| 1100 | Dieu Du Ciel - Peche Mortel | 3.90 | 1 | 473 | |
| 239 | Faxe Extra Strong | 4.10 | 1 | 473 | |

| | ABV (%) | Container | Type | Cost per mL of Alcohol (\$) |
|-----|---------|-----------|-------|-----------------------------|
| 80 | 5.9 | Can | Lager | 0.080 |
| 281 | 5.5 | Can | Lager | 0.083 |
| 29 | 5.5 | Can | Lager | 0.083 |
| 526 | 5.5 | Can | Lager | 0.083 |
| 251 | 6.0 | Can | Lager | 0.083 |
| 556 | 5.0 | Bottle | Lager | 0.084 |
| 481 | 7.0 | Can | Lager | 0.084 |
| 147 | 10.0 | Can | Lager | 0.084 |
| 491 | 7.0 | Can | Lager | 0.084 |
| 88 | 7.0 | Can | Lager | 0.084 |
| 445 | 10.0 | Can | Lager | 0.085 |
| 406 | 5.2 | Can | Lager | 0.085 |
| 204 | 5.5 | Can | Lager | 0.085 |
| 131 | 5.5 | Can | Lager | 0.086 |

| | | | | |
|------|------|-----|-------|-------|
| 267 | 5.5 | Can | Lager | 0.086 |
| 59 | 5.5 | Can | Lager | 0.086 |
| 866 | 10.5 | Can | Ale | 0.086 |
| 359 | 5.5 | Can | Lager | 0.087 |
| 1100 | 9.5 | Can | Ale | 0.087 |
| 239 | 10.0 | Can | Lager | 0.087 |

```
[348]: df = df.drop_duplicates(subset = ['Name', 'Price ($)', 'Quantity', 'Volume_
↪(mL)', 'ABV (%)', 'Container', 'Type'])
df = df.reset_index(drop = True)
```

```
[358]: df["Cost per Unit ($)"] = (df["Price ($)"] / df["Quantity"]).round(3)
```

```
[359]: df.sort_values(by = "Cost per Unit ($)")[0:20]
```

```
[359]:
```

| | Name | Price (\$) | Quantity | Volume (mL) | ABV (%) | \ |
|-----|-------------------------|------------|----------|-------------|---------|---|
| 53 | Molson Cold Shots 6.0 | 5.00 | 4 | 222 | 6.0 | |
| 555 | No Name Beer | 8.55 | 6 | 341 | 5.0 | |
| 506 | Coors Light | 45.95 | 28 | 341 | 4.0 | |
| 301 | Bud Light | 45.95 | 28 | 341 | 4.0 | |
| 518 | Molson Canadian | 45.95 | 28 | 341 | 5.0 | |
| 491 | Laker Premium Lager | 39.50 | 24 | 341 | 5.0 | |
| 500 | Laker Light | 39.50 | 24 | 341 | 4.0 | |
| 504 | Laker Ice | 39.50 | 24 | 341 | 5.5 | |
| 140 | Coronita Extra | 9.95 | 6 | 207 | 4.6 | |
| 199 | Miller High Life | 49.95 | 30 | 355 | 4.6 | |
| 462 | Labatt Blue | 49.95 | 30 | 355 | 5.0 | |
| 535 | Pabst Blue Ribbon Light | 39.95 | 24 | 341 | 4.0 | |
| 358 | Laker Ice | 40.95 | 24 | 355 | 5.5 | |
| 492 | Laker Lager | 40.95 | 24 | 355 | 5.0 | |
| 422 | Busch Lager | 52.95 | 30 | 355 | 4.7 | |
| 334 | Budweiser | 49.95 | 28 | 341 | 5.0 | |
| 543 | Busch Light | 10.75 | 6 | 355 | 4.0 | |
| 550 | Keystone Light | 53.95 | 30 | 355 | 4.0 | |
| 294 | James Ready 5.5 | 43.40 | 24 | 341 | 5.5 | |
| 310 | Rolling Rock | 43.50 | 24 | 341 | 4.5 | |

| | Container | Type | Cost per mL of Alcohol (\$) | Cost per Unit (\$) |
|-----|-----------|-------|-----------------------------|--------------------|
| 53 | Can | Lager | 0.094 | 1.250 |
| 555 | Bottle | Lager | 0.084 | 1.425 |
| 506 | Bottle | Lager | 0.120 | 1.641 |
| 301 | Bottle | Lager | 0.120 | 1.641 |
| 518 | Bottle | Lager | 0.096 | 1.641 |
| 491 | Bottle | Lager | 0.097 | 1.646 |
| 500 | Bottle | Lager | 0.121 | 1.646 |
| 504 | Bottle | Lager | 0.088 | 1.646 |
| 140 | Bottle | Lager | 0.174 | 1.658 |

| | | | | |
|-----|--------|-------|-------|-------|
| 199 | Can | Lager | 0.102 | 1.665 |
| 462 | Can | Lager | 0.094 | 1.665 |
| 535 | Bottle | Lager | 0.122 | 1.665 |
| 358 | Can | Lager | 0.087 | 1.706 |
| 492 | Can | Lager | 0.096 | 1.706 |
| 422 | Can | Lager | 0.106 | 1.765 |
| 334 | Bottle | Lager | 0.105 | 1.784 |
| 543 | Can | Lager | 0.126 | 1.792 |
| 550 | Can | Lager | 0.127 | 1.798 |
| 294 | Bottle | Lager | 0.096 | 1.808 |
| 310 | Bottle | Lager | 0.118 | 1.812 |

```
[361]: df['Total Volume (mL)'] = df['Volume (mL)'] * df['Quantity']
```

```
[376]: df['Total Alcohol Content (mL)'] = df['Total Volume (mL)'] * (df['ABV (%)'] / 100).round(3)
```

```
[377]: df.sort_values(by = ['Total Volume (mL)'][0:10], ascending = False)
```

```
[377]:
```

| | Name | Price (\$) | Quantity \ |
|------|-----------------------------------------------|------------|------------|
| 1244 | MacLean's Farmhouse Blonde | 77.25 | 24 |
| 305 | Budweiser | 58.95 | 30 |
| 199 | Miller High Life | 49.95 | 30 |
| 264 | Busch Light | 55.95 | 30 |
| 278 | Bud Light | 58.95 | 30 |
| ... | ... | ... | ... |
| 1197 | Duvel 666 | 4.20 | 1 |
| 1194 | Brouwerij The Musketeers Troubadour - Obscura | 4.60 | 1 |
| 503 | Eggenberg Brewery's Samichlaus | 6.25 | 1 |
| 1029 | Saison Dupont | 3.20 | 1 |
| 21 | Stonewall Light | 1.85 | 1 |

| | Volume (mL) | ABV (%) | Container | Type | Cost per mL of Alcohol (\$) | \ |
|------|-------------|---------|-----------|-------|-----------------------------|---|
| 1244 | 473 | 4.8 | Can | Ale | 0.142 | |
| 305 | 355 | 5.0 | Can | Lager | 0.111 | |
| 199 | 355 | 4.6 | Can | Lager | 0.102 | |
| 264 | 355 | 4.0 | Can | Lager | 0.131 | |
| 278 | 355 | 4.0 | Can | Lager | 0.138 | |
| ... | ... | ... | ... | ... | ... | |
| 1197 | 330 | 6.6 | Bottle | Ale | 0.193 | |
| 1194 | 330 | 8.2 | Bottle | Ale | 0.170 | |
| 503 | 330 | 14.0 | Bottle | Lager | 0.135 | |
| 1029 | 330 | 6.5 | Bottle | Ale | 0.149 | |
| 21 | 275 | 4.0 | Bottle | Lager | 0.168 | |

| | Cost per Unit (\$) | Total Volume (mL) | Total Alcohol Content (mL) |
|------|--------------------|-------------------|----------------------------|
| 1244 | 3.219 | 11352 | 544.896 |

| | | | |
|------|-------|-------|---------|
| 305 | 1.965 | 10650 | 532.500 |
| 199 | 1.665 | 10650 | 489.900 |
| 264 | 1.865 | 10650 | 426.000 |
| 278 | 1.965 | 10650 | 426.000 |
| ... | ... | ... | ... |
| 1197 | 4.200 | 330 | 21.780 |
| 1194 | 4.600 | 330 | 27.060 |
| 503 | 6.250 | 330 | 46.200 |
| 1029 | 3.200 | 330 | 21.450 |
| 21 | 1.850 | 275 | 11.000 |

[1404 rows x 11 columns]

```
[380]: df.sort_values(by = ['Total Alcohol Content (mL)'][0:10], ascending = False)
```

```
[380]:
```

| | Name | Price (\$) | Quantity \ |
|------|------------------------------------------------|------------|------------|
| 471 | Busch Ice | 55.95 | 30 |
| 496 | James Ready 5.5 | 54.95 | 30 |
| 1244 | MacLean's Farmhouse Blonde | 77.25 | 24 |
| 229 | Molson Canadian | 55.95 | 30 |
| 219 | Moosehead Lager | 59.95 | 30 |
| ... | ... | ... | ... |
| 1229 | Bobcaygeon Brewing Bobber 100 Calorie Hazy IPA | 3.50 | 1 |
| 790 | Lake Of Bays Low Tide 90 Calorie IPA | 3.45 | 1 |
| 21 | Stonewall Light | 1.85 | 1 |
| 246 | Corona Sunbrew | 12.95 | 6 |
| 688 | Bellwoods Jelly King Non-Alc Sour | 5.00 | 1 |

| | Volume (mL) | ABV (%) | Container | Type | Cost per mL of Alcohol (\$) | \ |
|------|-------------|---------|-----------|-------|-----------------------------|---|
| 471 | 355 | 6.00 | Can | Lager | 0.088 | |
| 496 | 355 | 5.50 | Can | Lager | 0.094 | |
| 1244 | 473 | 4.80 | Can | Ale | 0.142 | |
| 229 | 355 | 5.00 | Can | Lager | 0.105 | |
| 219 | 355 | 5.00 | Can | Lager | 0.113 | |
| ... | ... | ... | ... | ... | ... | |
| 1229 | 473 | 2.50 | Can | Ale | 0.296 | |
| 790 | 473 | 2.50 | Can | Ale | 0.292 | |
| 21 | 275 | 4.00 | Bottle | Lager | 0.168 | |
| 246 | 330 | 0.01 | Bottle | Lager | 65.404 | |
| 688 | 473 | 0.05 | Can | Ale | 21.142 | |

| | Cost per Unit (\$) | Total Volume (mL) | Total Alcohol Content (mL) |
|------|--------------------|-------------------|----------------------------|
| 471 | 1.865 | 10650 | 639.000 |
| 496 | 1.832 | 10650 | 585.750 |
| 1244 | 3.219 | 11352 | 544.896 |
| 229 | 1.865 | 10650 | 532.500 |
| 219 | 1.998 | 10650 | 532.500 |

| | | | |
|------|-------|------|--------|
| ... | ... | ... | ... |
| 1229 | 3.500 | 473 | 11.825 |
| 790 | 3.450 | 473 | 11.825 |
| 21 | 1.850 | 275 | 11.000 |
| 246 | 2.158 | 1980 | 0.000 |
| 688 | 5.000 | 473 | 0.000 |

[1404 rows x 11 columns]

```
[381]: df["Total Alcohol Content (mL)"] = df["Total Alcohol Content (mL)"].round(3)
```

1.0.8 Export dataframe to .csv file

```
[392]: df.to_csv("lcbo_data.csv", index = False)
```

```
[ ]:
```