HDSB Web Scraper 1st Part

Includes; School Name, School Address, School Email, School Phone Number, School Website.

To make sure that all of the school's information is saved, there is a loop that saves the website URL and runs the code for each website.

```
wbs_url_bad = ["https://abb.hdsb.ca",
"https://act.hdsb.ca",
"https://ald.hdsb.ca",
"https://alx.hdsb.ca",
"https://avp.hdsb.ca",
"https://ajm.hdsb.ca",
#ect..
]


for x in wbs_url_bad:
    wbs_url = x
    outcome = requests.get(wbs_url)
    doc = BeautifulSoup(outcome.text, "html.parser")
```

Finding School Name

The web scraper finds the school name by,

1. Finding the HTML class QTKDff p46B7e which is in the top corner of the website and that contains the name of the school.
2. It finds the text in that class. If the text is not found the code saves the name as N/A.

```
find_nam = doc.find(class_="QTKDff p46B7e")
nam_result = find_nam.string if find_nam else "N/A"
```

Finding Address

The web scraper finds addresses by

1. locating a ", ON " or ", ON" or ", ON "
2. Saves the entire textbox as the address. (ex. 2145 Grand Oak Trail, Oakville, ON L6M 4S7, finds ", ON " and saves the whole address.)
3. If locating a ", ON " or ", ON" or ", ON " fails then the address is saved as "N/A".

```
find_ON = doc.select_one("span:-soup-contains(', ON ' )") or
doc.select_one("span:-soup-contains(', ON')") or
doc.select_one("span:-soup-contains(', ON  ')")
Add_result = (find_ON.get_text().strip() if find_ON else "N/A")
```

Find School Email

1. Finds "@hdsb.ca"
2. Saves the entire textbox as the address. (ex. crw@hdsb.ca finds "@hdsb.ca " and saves the whole email
3. If step 1 fails then the email is saved as "N/A".

```
find_domain = doc.select_one("span:-soup-contains('@hdsb.ca')")
mal_result = (find_domain.get_text().strip() if find_domain else
"N/A")
```

Find School Phone Number

1. Looks for 'T: '
2. Saves the rest of the text after it.
3. If it cannot find "T: ", the phone number is saved as "N/A"

```
find_T = doc.select_one("span:-soup-contains('T: ')")
P_result = find_T.get_text().strip()[3:16] if find_T else "N/A
```

Find School Website

1. Saves the website it scraped

```
 wbs_result = wbs_url
```

After saving everything it is all saved in a CSV file

```
    full_data = f"{nam_result};{Add_result};{mal_result};{P_result};
{wbs_result}\n"


    with open("data1.csv","a") as file:
        for loop in range(99):
            if loop == 99:
                break
        file.write(full_data)
```

Then all of this restarts for the next schools untill the end of the loop.

HDSB Web Scarper 2ed Part

By using the HDSB school details pages the scraper collects; School Names, Grade Groups, Number Of Students, School Principle And Vice Principle/Senior Administrative Assistant.

To make sure that all of the school's information is saved, there is a loop that saves the website URL and runs the code for each website.

```
dwbs_url_bad =["http://www.hdsb.ca/schooldetails/SchoolDetails.aspx?
sc=1100",
"http://www.hdsb.ca/schooldetails/SchoolDetails.aspx?sc=1003",
"http://www.hdsb.ca/schooldetails/SchoolDetails.aspx?sc=1005",
#ect
]
for y in dwbs_url_bad:
    dwbs_url = y
    outcome = requests.get(dwbs_url)
    doc = BeautifulSoup(outcome.text, "html.parser")
```

Finding School Name

The web scraper finds the school name by,

1. Finding the HTML class SchoolNameLinkTitle which is in the top corner of the website and that contains the name of the school.
2. It finds the text in that class and saves it as a school name.

```
find_nam = doc.find(class_="SchoolNameLinkTitle")
nam_result = find_nam.string
```

Finding School Grade Group

1.Finds HTML class "SearchResultDetailBoxes"

2.Finds the text in the 2ed text box

3.Saves as Grade Group

```
find_grd = doc.find_all("div",class_="SearchResultDetailBoxes" )[1]
html_content5 = str(find_grd)
pattern = r"(.*) - (.*)"
match = re.search(pattern, html_content5)
if match:
   grd_find = match.group(0).strip()
   grd_result = (grd_find)
```

Finding the Number Of Students

1. Finds HTML class "SearchResultDetailBoxes"
2. Finds the text in the 3ed text box
3. Saves as Number of students

```
findc_stn = doc.find_all("div",class_="SearchResultDetailBoxes" )[2]
find_stn = findc_stn.find("td")
stn_reslut = find_stn.text.strip()
```

Finding Prinsple Name and Email

1. Finds class "SchoolDetailedInfo".

2. finds the 1st link(Its a hyperlink).
3. Finds the email in the "mailto:\(email\)" hyperlink and saves as email.
4. Finds the hyperlink label and saves it as the name.

```
findc_pri = doc.find("div",class_="SchoolDetailedInfo" )
find_pri = findc_pri.find_all("a")[0]
html_content_pri = str(find_pri)
pattern_pri = r'<a href="mailto:(.*)">(.*)</a>'
match_pri = re.search(pattern_pri, html_content_pri)
if match_pri:
    pri_n_reslut = match_pri.group(2).strip()
    pri_e_reslut = match_pri.group(1).strip()
```

Finding Vice Prinsple/Senior Administrative Assistant Name and Email

1. Finds class "SchoolDetailedInfo".
2. finds the 2st link(Its a hyperlink).
3. Finds the email in the "mailto:\(email\)" hyperlink and saves as email.
4. Finds the hyperlink label and saves it as the name.

```
findc_pri = doc.find("div",class_="SchoolDetailedInfo" )
find_pri = findc_pri.find_all("a")[1]
html_content_pri = str(find_pri)
pattern_pri = r'<a href="mailto:(.*)">(.*)</a>'
match_pri = re.search(pattern_pri, html_content_pri)
if match_pri:
    pri_n_reslut = match_pri.group(2).strip()
    pri_e_reslut = match_pri.group(1).strip()
```

After saving everything it is all saved in a CSV file

```
full_data2 =f"{nam_result};{grd_result};{stn_reslut};{pri_n_reslut};
{pri_e_reslut};{saa_n_reslut};{saa_e_reslut}"
f = open("data2.csv", "a")
f.write("\n"+full_data2)
f.close()
```

Then all of this restarts for the next schools until the end of the loop.