CPRG256
FINAL PROJECT

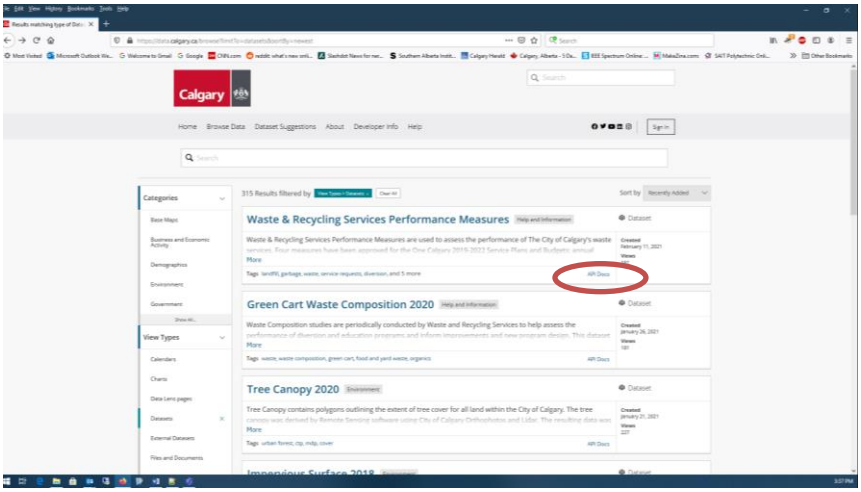
NO LATE SUBMISSIONS WILL BE ACCEPTED. ONLY PROJECTS THAT HAVE BEEN SUBMITTED BY THE DUE DATE WILL BE GRADED. THIS IS AN INDIVIDUAL ASSIGNMENT SUBMISSION

Create a directory 'cprg256/finalproject'. Create the required subdirectories for the following applications.

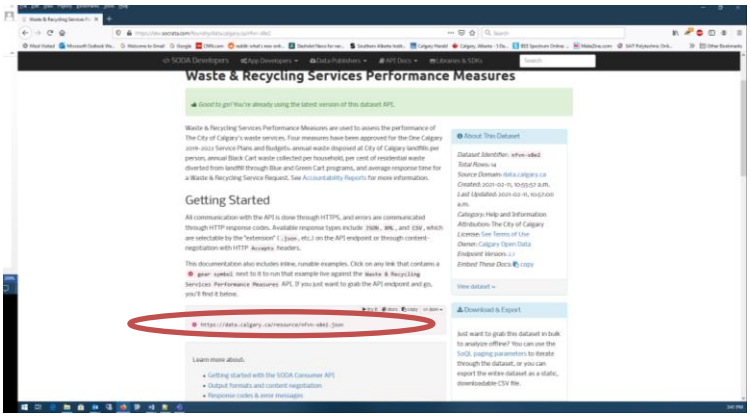
PART A:

Create a directory titled 'datasets' under cprg256/finalproject.

The City of Calgary publishes data about city operations and can be found here: https://data.calgary.ca . On the site, you can click on 'Datasets' and on some of the links, you will see a small link to API docs.



Clicking on API Docs will bring you to the next page:



The highlighted area is the link to the json file for this dataset. You use this hyperlink in you 'AJAX' call to retrieve the data. You can view the json file by selecting the </>json drop down menu and choosing json. Note: It may take a few minutes to generate the data; some of the datasets are quite large.

Here are some datasets:

Traffic Incidents: https://data.calgary.ca/resource/35ra-9556.json
Traffic Cameras: https://data.calgary.ca/resource/k7p9-kppz.json
Crime Stats: https://data.calgary.ca/resource/848s-4m4z.json
Building Permits: https://data.calgary.ca/resource/c2es-76ed.json

Here is a sample of the json file for the Building Permit datasets:


```
{
        "permitnum": "BP2021-04324",
        "statuscurrent": "In Review",
        "applieddate": "2021-03-22T00:00:00.000",
        "permittype": "Single Construction Permit",
        "permittypemapped": "Building",
        "permitclass": "1106 - Single Family House",
        "permitclassgroup": "Single Family",
        "permitclassmapped": "Residential",
        "workclass": "New",
        "workclassgroup": "New",
        "workclassmapped": "New",
        "description": "SFD,GARAGE,PORCH,",
        "contractorname": "JAYMAN MASTERBUILT / WENDI INTERIORS",
        "housingunits": "1",
        "estprojectcost": "295360.54",
        "totalsqft": "1807",
        "originaladdress": "330 RED SKY TC NE",
        "communitycode": "RSN",
        "communityname": "REDSTONE",
        "latitude": "51.17361948379003",
        "longitude": "-113.94850406578414",
        "location": {
            "latitude": "51.17361948379003",
            "longitude": "-113.94850406578414",
            "human_address": "{\"address\": \"\", \"city\": \"\", \"state\": \"\", \"zip\":
\"\"}"
        },
        "locationcount": "1",
        "locationtypes": "Titled Parcel",
        "locationaddresses": "330 RED SKY TC NE",
        "locationswkt": "POINT (-113.94850406578414 51.17361948379003)",
        "locationsgeojson": "{\"type\":\"Point\",\"coordinates\":[-
113.94850406578414,51.17361948379003]}",
        ":@computed_region_4b54_tmc4": "11",
        ":@computed_region_4a3i_ccfj": "4",
        ":@computed_region_kxmf_bzkv": "3",
        ":@computed_region_p8tp_5dkv": "1"
    }
```


Other datasets will have their own structure.
Please note, there is no consistent format between datasets.

Requirements

For this assignment, you must choose four datasets from https://data.calgary.ca. You have the choice of using any of the above dataset or finding other datasets from https://data.calgary.ca .

VITAL: Your dataset must include a location that includes latitude and longitude.

This assignment will require the use of HTML, CSS, JavaScript and AJAX.

Create the file index.html. This will be the first page of the assignment. You are to create a web page with a creative a pleasing interface that lists and describes the four datasets that can be searched.

When a dataset is selected an AJAX call will load the html page containing a form that will allow the dataset to be searched.

The form will require no less than 3 and no more than 5 entries that can be searched. DO NOT SEARCH ON LATITUDE AND LONGITUDE.

Search Criteria

You have some options on how to search for data.

Search fields may be connected to a unique Event Listener that is set to call a function when the 'keyup' or 'onkeyup' event occurs. As data is typed into the search field, the search function will display all records that start with that search string.

- Example: If you are searching by 'id' and you enter '2' all the records with 'id' starting with '2' will be displayed. If you enter '25' all the records with 'id' starting with '25' will be displayed.

A search field can also provide a range for data:

-Example: You want to list all building permits with an estimated project cost greater than $100 000

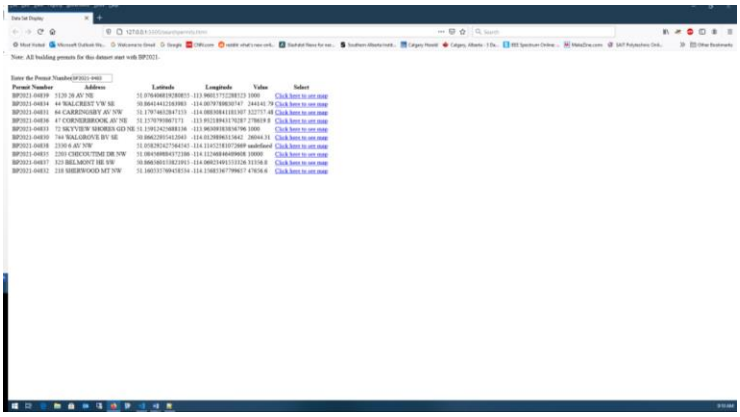
You do not need to display all of the data from the dataset record as they can be quite large. Choose the most relevant data from the dataset records.

Display location using Google Maps

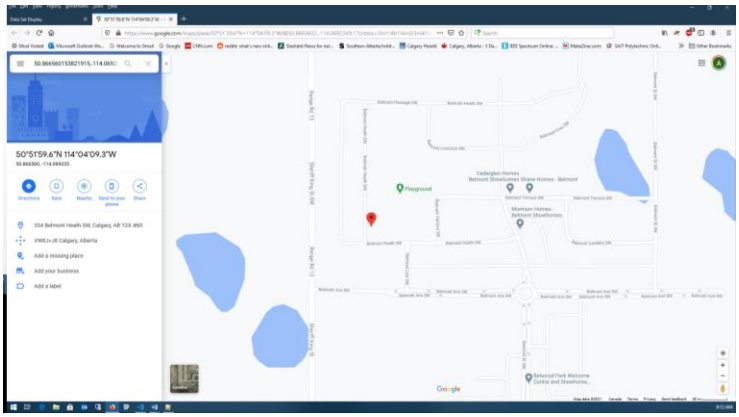
The user will have the option of selecting a search result and view the location on google maps. When a records is selected, a new window tab will open and display a Google Map with the 'pin' at the latitude and longitude of the search result.

Example

NOTE: This is only a partial example of the required functionality. This generates a list of possible records and a corresponding link to display the location on a google map.



Clicking on one of the links displays the following:



PART B:

Create a subdirectory titled ‘quiz’ under cprg256/finalproject.

You are to create a web page that will display a quiz and create the necessary JavaScript to complete the following:

- Process the ‘finalquiz.xml’ file to display the multiple choice questions within the index.html page. Each question has 4 possible answers.

- Create a function that will grade the quiz and display the grade out of 5. Within ‘finalquiz.xml’ is an element ‘rightanswers’ that has a comma delimited string with the correct answers. Use CSS to create an appropriate interface.

Here is an example of what the page could look like:

Question 1:

In a switch statement, the _____ case clause is used to process exceptional conditions and is usually listed last.

- A) break
- B) default
- C) else
- D) then

At the end of the quiz, create a button ‘Grade Quiz’ that will display the number of correct answers out of five. E.g. ‘Grade 3/5’

Part C:

Create a subdirectory titled ‘welldata’ under cprg256/finalproject.

You will create an ‘index.html’ page that will display and search for data as described below.

You have been provided with two files: ‘welldata.xml’ and ‘productiondata.xml’. The ‘welldata.xml’ contains the following data:

- Location of the well using the Alberta Township System (See below for details)
- Depth of the well in meters
- Perforation depth (top) in meters
- Perforation zone in meters
- Pump stroke length
- Strokes per minute

A well location in accordance with the Alberta Township System shown below:

Example: B15-98-17-W5

Section		Township		Range		Meridian
B15	-	98	-	17	-	W5

VALID DATA RANGE:

Section: A-D and 1-16

Township: 1-126

Range: 1-24

Meridian: W and 4-6

The 'productiondata.xml' file contains the following data:

- A well location
- Date
- Oil production in m^3/day
- Water production in m^3/day
- Gas production in 10^3 m^3/day

Your application will do the following:

Search by specific location of a well.

The user can enter the full location using the Alberta Township System e.g. B15-98-17-W5. Your application will search 'welldata.xml' to find the oil well. If the oil well is found, it searches 'productiondata.xml' for the production information for that well. The data is then displayed on the web page in separate <form> or element. If no well is found, an error message will be printed to the page.

Any data input must be validated according to the Valid Data Range listed above.

Include buttons that will start the search or clear data from the form.

The application must use AJAX to process the xml files and display the search responses.

SUBMISSION REQUIREMENTS

You must use JavaScript to implement your programming solutions. NOTE: If you plan to use an Third Party Applications (eg. Bootstrap), you must consult your instructor in advance.

You are required to use CSS/Styling to create pleasing and useful interfaces for your applications.

Validate your pages BEFORE submission. Use the validator at w3schools.com

'Zip' ALL of your project files and upload them to the Brightspace site under Assignments/Final Project