function ajaxRequest(method, url, data, callback){

    let request = new XMLHttpRequest();

    request.open(method, url, true);

    if(method == "POST"){

        request.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded');

    }

    request.onreadystatechange = function(){

        if(request.readyState == 4){

            if(request.status == 200){

                let response = request.responseText;

                callback(response);

             }

else {

err(response.statusText);

}

        }

    }

    request.send(data);

}

We will use this function in our assignment one so we need to understand it before we use it.

A typical usage is that I defined a callback function like this

function myCallback(response){

alert(response); // Show the response text

}

Then I can make a function like

ajaxRequest('GET', 'php/getWeather.php?town=hamilton', null, myCallback);

The procedure is: we will create XMLHttpRequest to open 'php/getWeather.php?town=hamilton' with GET method. Once the request returns results, we will pass the results to myCallback function to show the result.

When we try to query the cities that a user has not selected, we can do：

One approach is get a list of cities that the user has already added, and then make another query to `weather` table to find a list of unselected cities that are not in the list.

This query usually needs two queries. But we can try to use 'subquery' and NOT in keywords to combine these two queries into a single one.

A useful link about subquery is

https://www.mysqltutorial.org/mysql-subquery/

Q: How to obtain a list of towns that users have not selected before? This will be used to add a new town weather for users.

A: This query contains two steps.

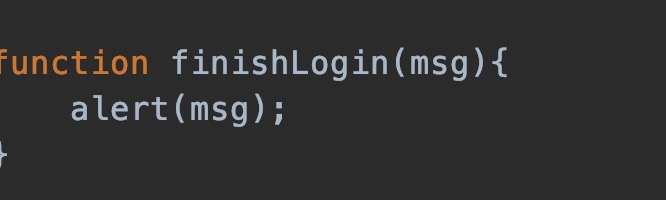
1. For a userid, query a list of towns selected by that user from 'userweatherjoin' table, e.g. user id =3 has already selected four towns (Hamilton, Christchurch, Dunedin and Wellington).

2. Query a list of town weather from 'weather' table where towns are not in user towns. That mean, we will query weather table, except not in user town, and return the weather of Auckland and Tauranga.

We can try to use MySQL 'subQuery' to query the list of user towns (Hamilton, Christchurch, Dunedin and Wellington), and then make another query to 'weather' table where towns are not in the list of user towns.

By using 'subquery' and 'NOT IN', we can combine two queries into a single query, to simplify the PHP code. A useful link about subquery is

<https://www.mysqltutorial.org/mysql-subquery/>



XHR Object Properties

Onreadystatechange

Register a listener for this event to specify a callback function

readystate

Keep track of the request’s progress, used to determine when to process the response

responseText

Text that’s returned by the server

responseXML

If the response is in XML format this property contains the XML document

status

HTTP status code for the request

statusText

Additional information on the request’s status, can be used to display error when request fails

readyState

 readyState has 5 possible values which indicate whether or not the request is in a state ready to be processed

◦ 0 – request not initialised

◦ 1 – request loading

◦ 2 – request loaded

◦ 3 – data actively being sent from server

◦ 4 – request complete

status

 status has many possible values, of most interest to us are

◦ 200 – request was successful

◦ 404 – requested resource not found

◦ 500 – error during request processing

 A full list can be found at:

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

XHR Object Methods

open

Initialises the request, has 2 mandatory parameters (method and url)

send

Sends the request to the server, optional parameter allows data to be POSTed to the server

setRequestHeader

Alters the request header, often used to set content-type field

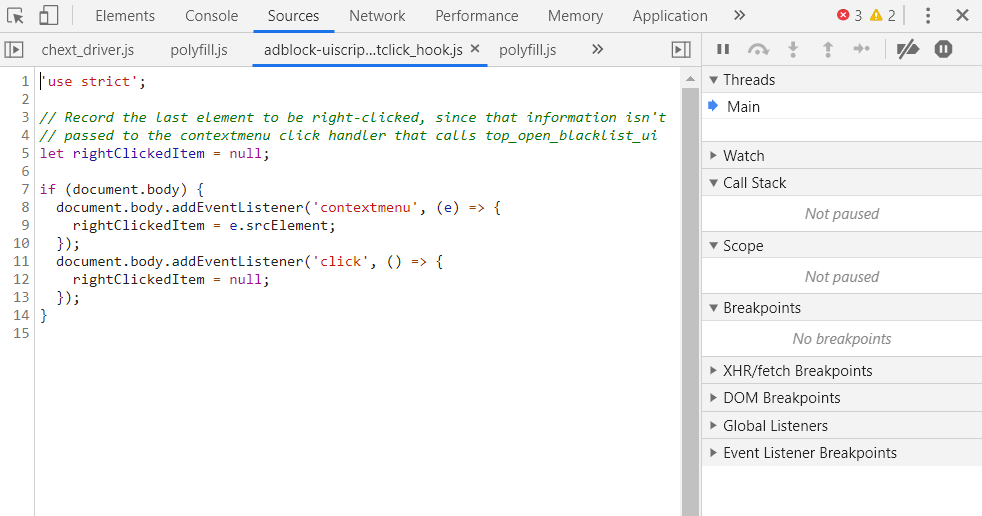
getResponseHeader

Returns the header data that precedes the response body, used to determine the type of the response to help parse correctly

getAllResponseHeaders

Returns an array containing all of the headers abort Cancels the current request

For JS, we could use Chrome editor tool. Start your website > Right Click > Inspect > Sources Tab > Click on your JS file. Then you enter the debug mode, and then can set break pont at any line in JS and observe the variable values in Chrome Debug mode.



For PHP, we could make each PHP to receive GET parameter (for debugging purpose) and directly enter the url with parameter in Chrome to show the results on the page. For example, we have doLogin.php script, and it takes username and password as GET paramters. Then we start our website and enter the below urls in Chrome

http://localhost:80/weather/php/doLogin.php?username=aaa&password=bbb

It will output the result on the webpage.

