**Introduction:**

water remedy aims to help people in Melbourne to manage their water usage. On the one hand, we provide interactive solutions to roof water collection; on the other hand, we suggest people to use DIY cleaners to replace chemical detergents in order to reduce water pollution.

We got two major functions:

1. Clean water:

We provide top five common harmful detergents in three scenarios, kitchen, laundry and bathroom.

And click alternatives, will show the DIY cleaner formula in an interesting way.

2. Water saver:

In this part, we will give a guideline step by step on how to measure their roof; how much water can be saved from roof. Then what water tanks should be considered and how to install the tank.

**Install:**

Go to the git hub link: <https://github.com/jayce-cxy/WaterRemedyProject>

Click Code then Download as ZIP.

Graphical user interface, text, application, chat or text message

Description automatically generated

**Overview of main function codes:**

***Draw the sharp of the roof and calculate the area:***

1. First use API KEY to activate usage rights

Add the <script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyD8juZ2HNagjrCBrnnjuBkhccADbTfjgt8&callback=initMap&libraries=drawing,places,geometry&v=weekly"

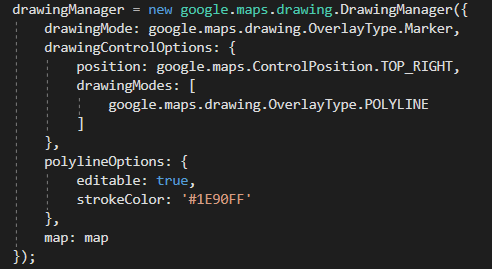
defer></script>

at the bottom of the body.

1. Initialize variables:



1. Creates a drawing manager attached to the map that allows the user to draw



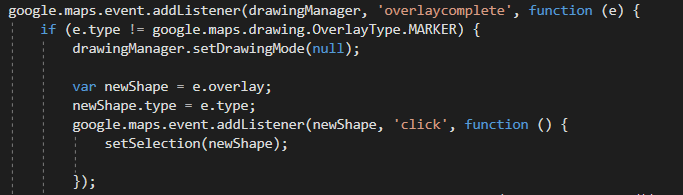
You can also add more tool bars than “polyline” such as:

rectangleOptions: polyOptions,

circleOptions: polyOptions,

polygonOptions: polyOptions,

1. Add event listener “overlaycomplete” that selects the newly drawn shape when the user mouses down on it.



1. After that use computeArea() to calculate the area of that shape (Function document: <https://developers.google.com/maps/documentation/javascript/geometry>)



***Calculate the storage:***

1. Get all the necessary data form database

The details code of retrieving data is in “public ActionResult RoofWater()” under the HomeController.

1. The calculation model

|  |
| --- |
| 1. var area = document.getElementById('houseArea').value; 2. var totalSaveMoney = 0.0; 3. var yearStorage; 4. var indoorSave = 0.0; 5. var outdoorSave = 0.0; 6. var currentMonthIndoorReq = @ViewBag.currentMonthIndoorReq; 7. var rainfallList = @ViewBag.list; 8. var rainfallMonth = @Html.Raw(Json.Encode(@ViewBag.rfmList)); 9. var indoorRequired = @Html.Raw(Json.Encode(@ViewBag.indoorReq)); 10. var outdoorRequired = @Html.Raw(Json.Encode(@ViewBag.outdoorReq)); 11. var avgMonthRainfall = 0; 12. console.log(area); 13. console.log(rainfallList); 14. yearStorage = parseFloat(area) \* rainfallList; 15. console.log(yearStorage); 16. document.getElementById('lyear').innerHTML = yearStorage.toFixed(0); 17. avgMonthRainfall = yearStorage / 12; 18. document.getElementById('storageWater').innerHTML = avgMonthRainfall.toFixed(0); 19. for (var i = 0; i < rainfallMonth.length; i++) { 20. var currentMonthStorage = parseFloat(area) \* parseFloat(rainfallMonth[i]); 21. parseFloat(rainfallMonth[i]); 22. if (currentMonthStorage == 0) { 23. totalSaveMoney = 0.0; 24. $('#cal').notify("Please measure your roof", options); 25. break; 26. } 27. monthStorageArray.push(parseFloat(currentMonthStorage.toFixed(0))); 28. if (currentMonthStorage >= outdoorRequired[i]) { 29. indoorSave += outdoorRequired[i] / 1000 \* 2.683; 30. } else { 31. indoorSave += currentMonthStorage / 1000 \* 2.683; 32. } 33. var remainingWater = currentMonthStorage - indoorRequired[i]; 34. if (remainingWater < 0) { 35. outdoorSave += 0.0; 36. } else if (remainingWater < outdoorRequired[i]) { 37. outdoorSave += remainingWater / 1000 \* 3.2552; 38. } else if (remainingWater >= outdoorRequired[i]) { 39. outdoorSave += outdoorRequired[i] / 1000 \* 3.2552; 40. } 41. } 42. totalSaveMoney = outdoorSave + indoorSave; 43. document.getElementById('savemoney').innerHTML = totalSaveMoney.toFixed(0); |

the brief explanation of above:

1. This month's water storage volume (V) = roof area \* monthly average rainfall
2. Monthly indoor water savings ($):

if V >= indoor usage, indoor water can be saved = indoor water volume/1000 \* 2.683

if V <indoor usage, indoor water can be saved = V/1000 \* 2.683

1. Monthly outdoor water savings ($):

The remaining water volume is recorded as RV, RV = (V-indoor water volume)

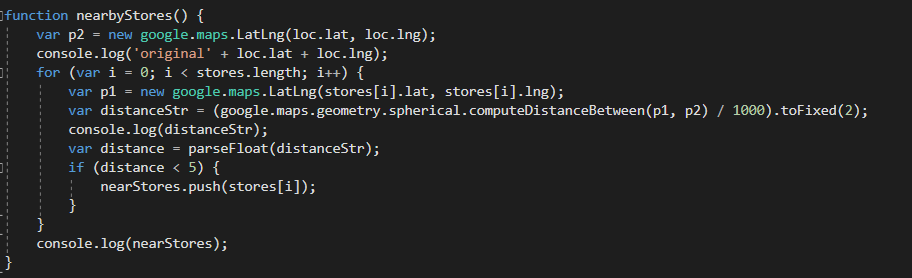
RV ≥ 3004, outdoor water can be saved = 3004/1000 \* 3.2552

RV <3004, save water bills outdoors = RV/1000 \* 3.2552

1. Monthly water savings = monthly indoor water savings + monthly outdoor water savings
2. One year's water savings = the sum of 12 months' water savings

***Find the nearby stores:***

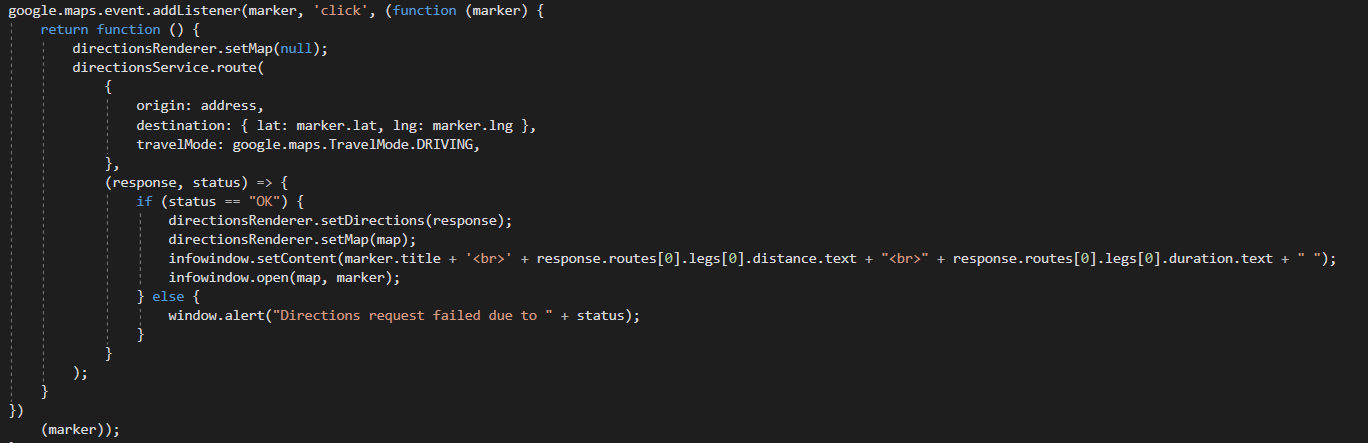
1. Calculate the distance form the current location to the location of stores to find the most closet stores:



1. Initialize variables:



1. Add click listener to the button, which can show the duration and distance (Function document: <https://developers.google.com/maps/documentation/javascript/directions>):



***Convert HTML to PDF:***

1. Import the JS file:

<script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/1.3.2/jspdf.min.js"></script>

1. Add ID to the module you want to convert:

e.g. <div id=”convertToPdf”>…</div>

1. Write the script:

Initialize jsPDF ---- var doc = new jsPDF('a4');

1. Get the DOM object:

const shareContent = document.getElementById('summaryPDF');

1. Then the function code:

$('#pdf').click(function () {

doc.fromHTML($('#printPDF').html(), 10, 10, {

'elementHandlers': specialElementHandler,

'width': 150

});

doc.save('WaterTankPlan.pdf');

});

The double ‘10’ behand the html() means the margin pdf top and left.

(Function document: <http://raw.githack.com/MrRio/jsPDF/master/docs/>)

**Contributor:**

Xinyu Cao

Anni Li

Hsing Chen

Kishore Babu Sab