1. App Script

Step 1 : Create new app script project. Click here to create App Script.

Step 2 : Copy and paste the below script which handles two attributes (id,name). You can change the logic, attributes as required.

function doGet(e){

var op = e.parameter.action;

var ss=SpreadsheetApp.openByUrl("Your Spread sheet URL");

var sheet = ss.getSheetByName("Sheet1");

if(op=="insert")

return insert\_value(e,sheet);

//Make sure you are sending proper parameters

if(op=="read")

return read\_value(e,ss);

if(op=="update")

return update\_value(e,sheet);

if(op=="delete")

return delete\_value(e,sheet);

}

//Recieve parameter and pass it to function to handle

function insert\_value(request,sheet){

var id = request.parameter.id;

var country = request.parameter.name;

var flag=1;

var lr= sheet.getLastRow();

for(var i=1;i<=lr;i++){

var id1 = sheet.getRange(i, 2).getValue();

if(id1==id){

flag=0;

var result="Id already exist..";

} }

//add new row with recieved parameter from client

if(flag==1){

var d = new Date();

var currentTime = d.toLocaleString();

var rowData = sheet.appendRow([currentTime,id,country]);

var result="Insertion successful";

}

result = JSON.stringify({

"result": result

});

return ContentService

.createTextOutput(request.parameter.callback + "(" + result + ")")

.setMimeType(ContentService.MimeType.JAVASCRIPT);

}

function read\_value(request,ss){

var output = ContentService.createTextOutput(),

data = {};

//Note : here sheet is sheet name , don't get confuse with other operation

var sheet="sheet1";

data.records = readData\_(ss, sheet);

var callback = request.parameters.callback;

if (callback === undefined) {

output.setContent(JSON.stringify(data));

} else {

output.setContent(callback + "(" + JSON.stringify(data) + ")");

}

output.setMimeType(ContentService.MimeType.JAVASCRIPT);

return output;

}

function readData\_(ss, sheetname, properties) {

if (typeof properties == "undefined") {

properties = getHeaderRow\_(ss, sheetname);

properties = properties.map(function(p) { return p.replace(/\s+/g, '\_'); });

}

var rows = getDataRows\_(ss, sheetname),

data = [];

for (var r = 0, l = rows.length; r < l; r++) {

var row = rows[r],

record = {};

for (var p in properties) {

record[properties[p]] = row[p];

}

data.push(record);

}

return data;

}

function getDataRows\_(ss, sheetname) {

var sh = ss.getSheetByName(sheetname);

return sh.getRange(2, 1, sh.getLastRow() - 1, sh.getLastColumn()).getValues();

}

function getHeaderRow\_(ss, sheetname) {

var sh = ss.getSheetByName(sheetname);

return sh.getRange(1, 1, 1, sh.getLastColumn()).getValues()[0];

}

//update function

function update\_value(request,sheet){

var output = ContentService.createTextOutput();

var id = request.parameter.id;

var flag=0;

var country = request.parameter.name;

var lr= sheet.getLastRow();

for(var i=1;i<=lr;i++){

var rid = sheet.getRange(i, 2).getValue();

if(rid==id){

sheet.getRange(i,3).setValue(country);

var result="value updated successfully";

flag=1;

}

}

if(flag==0)

var result="id not found";

result = JSON.stringify({

"result": result

});

return ContentService

.createTextOutput(request.parameter.callback + "(" + result + ")")

.setMimeType(ContentService.MimeType.JAVASCRIPT);

}

function delete\_value(request,sheet){

var output = ContentService.createTextOutput();

var id = request.parameter.id;

var country = request.parameter.name;

var flag=0;

var lr= sheet.getLastRow();

for(var i=1;i<=lr;i++){

var rid = sheet.getRange(i, 2).getValue();

if(rid==id){

sheet.deleteRow(i);

var result="value deleted successfully";

flag=1;

}

}

if(flag==0)

var result="id not found";

result = JSON.stringify({

"result": result

});

return ContentService

.createTextOutput(request.parameter.callback + "(" + result + ")")

.setMimeType(ContentService.MimeType.JAVASCRIPT);

}

Step 3 : Change the url of the spread sheet. Make sure sheet is shared .[anyone with the link can view]

Step 4 : Go to Publish -> Deploy as web app. A window pop up, here ->Who has access to the app: -> Anyone

then Publish/Update. Copy the published url and store it.

2. HTML PART

Step 1 : Create a file crud\_test.html and copy the below code.

<html>

<head>

<style>

table, th, td

{

margin:10px 0;

border:solid 1px #333;

padding:2px 4px;

font:15px Verdana;

}

th {

font-weight:bold;

}

#loader {

border: 16px solid #f3f3f3;

border-radius: 50%;

border-top: 16px solid blue;

border-bottom: 16px solid blue;

width: 60px;

height: 60px;

-webkit-animation: spin 2s linear infinite;

animation: spin 2s linear infinite;

visibility:hidden;

}

@-webkit-keyframes spin {

0% { -webkit-transform: rotate(0deg); }

100% { -webkit-transform: rotate(360deg); }

}

@keyframes spin {

0% { transform: rotate(0deg); }

100% { transform: rotate(360deg); }

}

</style>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js"></script>

<script>

var script\_url = "Your published web app url";

// Make an AJAX call to Google Script

function insert\_value() {

$("#re").css("visibility","hidden");

document.getElementById("loader").style.visibility = "visible";

$('#mySpinner').addClass('spinner');

var id1= $("#id").val();

var name= $("#name").val();

var url = script\_url+"?callback=ctrlq&name="+name+"&id="+id1+"&action=insert";

var request = jQuery.ajax({

crossDomain: true,

url: url ,

method: "GET",

dataType: "jsonp"

});

}

function update\_value(){

$("#re").css("visibility","hidden");

document.getElementById("loader").style.visibility = "visible";

var id1= $("#id").val();

var name= $("#name").val();

var url = script\_url+"?callback=ctrlq&name="+name+"&id="+id1+"&action=update";

var request = jQuery.ajax({

crossDomain: true,

url: url ,

method: "GET",

dataType: "jsonp"

});

}

function delete\_value(){

$("#re").css("visibility","hidden");

document.getElementById("loader").style.visibility = "visible";

$('#mySpinner').addClass('spinner');

var id1= $("#id").val();

var name= $("#name").val();

var url = script\_url+"?callback=ctrlq&name="+name+"&id="+id1+"&action=delete";

var request = jQuery.ajax({

crossDomain: true,

url: url ,

method: "GET",

dataType: "jsonp"

});

}

// print the returned data

function ctrlq(e) {

$("#re").html(e.result);

$("#re").css("visibility","visible");

read\_value();

}

function read\_value() {

$("#re").css("visibility","hidden");

document.getElementById("loader").style.visibility = "visible";

var url = script\_url+"?action=read";

$.getJSON(url, function (json) {

// Set the variables from the results array

// CREATE DYNAMIC TABLE.

var table = document.createElement("table");

var header = table.createTHead();

var row = header.insertRow(0);

var cell1 = row.insertCell(0);

var cell2 = row.insertCell(1);

cell1.innerHTML = "<b>ID</b>";

cell2.innerHTML = "<b>Name</b>";

// ADD JSON DATA TO THE TABLE AS ROWS.

for (var i = 0; i < json.records.length; i++) {

tr = table.insertRow(-1);

var tabCell = tr.insertCell(-1);

tabCell.innerHTML = json.records[i].ID;

tabCell = tr.insertCell(-1);

tabCell.innerHTML = json.records[i].NAME;

}

// FINALLY ADD THE NEWLY CREATED TABLE WITH JSON DATA TO A CONTAINER.

var divContainer = document.getElementById("showData");

divContainer.innerHTML = "";

divContainer.appendChild(table);

document.getElementById("loader").style.visibility = "hidden";

$("#re").css("visibility","visible");

});

}

</script>

</head>

<body>

<div align="center">

<h1>CRUD OPERATION ON GOOGLE SPREAD SHEET, WEB APPLICATION USING GOOGLE APP SCRIPT .</h1>

<p>This is simple application, You can develop your own logic based on your requiremnets.<p>

<p>Be careful with the parameters you are sending, it should match the parameters that are recived in google app script.</p>

<form >

ID

<input type = "text" name ="id" id="id">

Name

<input type = "text" name ="name" id="name">

</form>

<div id="loader"></div>

<p id="re"></p>

<input type = "button" id = "b1" onClick="insert\_value()" value = "Insert"></input>

<input type="button" onclick="read\_value()" value="Read" />

<input type="button" onclick="update\_value()" value="Update" />

<input type="button" onclick="delete\_value()" value="Delete" />

<a href="https://docs.google.com/spreadsheets/d/1OmqvphAqP4WEcrsZAMNNeL2YVn3RJffVOIGgdPiyEVA/edit?usp=sharing" target="\_blank">Click here to open Spread Sheet </a>

<div id="showData"></div>

</div>

</body>

<div align="center">

<p>ANDROID LABS 2017 | All Rights Reserved</p>

</div>

<html>

Step 2 : Change the script url to your published web app script which was stored in step 4

<script>

var script\_url = "https://script.google.com/macros/s/AKfycbwyW-xxxxxxxxxxxxxxxxxxx-P1lzLGt/exec?";

-----------------------------

-----------------------------

</script>

Step 3 : Now check for operations