# Lesson 1

function test1() {

var spreadsheet = SpreadsheetApp.getActive();

spreadsheet.getRange('A1').activate();

spreadsheet.getCurrentCell().setValue('Hello');

spreadsheet.getRange('B1').activate();

spreadsheet.getCurrentCell().setValue('World');

spreadsheet.getRange('A1:B1').activate();

spreadsheet.getActiveRangeList().setFontWeight('bold')

.setFontSize(20)

.setBackground('#0000ff');

spreadsheet.getRange('D1').activate();

spreadsheet.getCurrentCell().setValue('Done');

};

# Lesson 2

function test2(){

const ss = SpreadsheetApp.getActiveSpreadsheet();

Logger.log(ss.getUrl());

Logger.log(ss.getId());

Logger.log(ss.getName());

}

# Lesson 3

function testSheet1(){

const ss = SpreadsheetApp.getActiveSpreadsheet();

const sheet = ss.getActiveSheet();

const data = sheet.getDataRange();

const values = data.getValues();

values.forEach((val)=>{

Logger.log(val[1]);

})

Logger.log(values);

Logger.log(sheet.getName());

}

# Lesson 4

function test2(){

const url = 'https://docs.google.com/spreadsheets/d/1rrORyEVHbvl\_8I44q-\_iIpO2SuY79qBKo66NEfKwg24/';

const ss = SpreadsheetApp.openByUrl(url);

//const sheet = ss.getSheets()[1];

const sheet = ss.getSheetByName('New 3');

//sheet.setName('UPDATED 500');

if(sheet != null){

Logger.log(sheet.getIndex());

}else{

Logger.log(sheet);

}

}

function test1() {

const id = '1rrORyEVHbvl\_8I44q-\_iIpO2SuY79qBKo66NEfKwg24';

const ss = SpreadsheetApp.openById(id);

const sheets = ss.getSheets();

sheets.forEach((sheet,index)=>{

Logger.log(sheet.getName());

sheet.setName('Updated '+index);

})

Logger.log(sheets);

}

# Lesson 5

function makeNewOne(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheets = ss.getSheets();

Logger.log(sheets);

const newName = 'Sheet New';

let sheet = ss.getSheetByName(newName);

if(sheet == null){

sheet = ss.insertSheet();

sheet.setName(newName);

}

Logger.log(sheet.getIndex());

}

# Lesson 6

function addColors(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheet = ss.getSheets()[0];

let counter = 0;

for(let i=1;i<51;i++){

let backColor = 'green';

for(let x=0;x<5;x++){

let val = 'A'.charCodeAt()+x;

let letterVal = String.fromCharCode(val);

Logger.log(letterVal);

counter++;

if((counter%2)==0){

backColor = 'pink';

}else{

backColor = 'yellow';

}

sheet.getRange(letterVal+i).setBackground(backColor);

}

}

}

# Lesson 7

function addColors2(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheet = ss.getSheets()[0];

let backColor = 'red';

let mySize = 10;

for(let rows = 1;rows<51;rows++){

for(let cols=1;cols<11;cols++){

let total = rows + cols;

if((total%2)==0){

backColor = 'red';

}else{

backColor = 'pink';

}

let range = sheet.getRange(rows,cols);

range.setBackground(backColor);

range.setFontColor('white');

range.setFontSize(mySize+cols);

range.setValue(total);

}

}

}

# Lesson 8

function getMyRange(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheet = ss.getSheets()[0];

const range = sheet.getRange(1,4,2,2);

const data = range.getValues();

range.setValues([['test1','test2'],['test3','test4']]);

range.setBackground('blue');

Logger.log(data);

}

# Lesson 9 & 10

function testData1(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const sheet = SpreadsheetApp.openById(id).getSheets()[0];

const lastCol = sheet.getLastColumn();

const lastRow = sheet.getLastRow();

const range = sheet.getRange(1,1,lastRow,lastCol);

const rangeData = sheet.getDataRange();

const lastCorner = sheet.getRange(lastRow,lastCol);

lastCorner.setBackground('red');

Logger.log(lastCorner.getValue());

Logger.log(rangeData.getValues());

Logger.log(lastCol,lastRow);

Logger.log(range.getValues());

}

# Lesson 11

function test2(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

ss.setActiveSheet(ss.getSheets()[1]);

const sheet = ss.getActiveSheet();

const range = sheet.getRange('B2:G5');

const dimArr = [range.getLastRow(),range.getNumRows(),range.getLastColumn(),range.getNumColumns()];

Logger.log(dimArr);

sheet.setActiveRange(range);

///range.setBackground('yellow');

Logger.log(sheet.getName());

const selectedSel = sheet.getSelection();

const selRange = selectedSel.getActiveRange();

const data = selRange.getValues();

selRange.setBackground('purple');

Logger.log(data);

}

function test2(){

const ss = SpreadsheetApp.getActiveSpreadsheet();

const sheet = ss.getActiveSheet();

Logger.log(sheet.getName());

const selectedSel = sheet.getSelection();

const selRange = selectedSel.getActiveRange();

const data = selRange.getValues();

Logger.log(data);

}

# Lesson 12

function copyMe(){

const ss = SpreadsheetApp.getActiveSpreadsheet();

const sheet = ss.getActiveSheet();

const range = sheet.getActiveRange();

const data = range.getValues();

createASheet(data,ss,range);

range.setBackground('red');

}

function createASheet(data,ss,range){

const numSheets = ss.getSheets();

const sheetName = 'Sheet '+ numSheets.length;

let newSheet = ss.getSheetByName(sheetName);

if(newSheet == null){

newSheet = ss.insertSheet();

newSheet.setName(sheetName);

}else{

//newSheet.clearContents();

//newSheet.clearFormats();

newSheet.clear();

}

const newRange = newSheet.getRange(1,1,range.getNumRows(),range.getNumColumns());

newRange.setValues(data);

}

# Lesson 13

function onOpen(){

const ui = SpreadsheetApp.getUi();

ui.createMenu('New Opts')

.addItem('first','first')

.addItem('two','second')

.addSeparator()

.addSubMenu(ui.createMenu('sub')

.addItem('first','third')

.addItem('two','fourth')

)

.addItem('five','fifth')

.addToUi();

}

function first(){

logOut('ran first');

}

function second(){

logOut('ran second');

}

function third(){

logOut('ran third');

}

function fourth(){

logOut('ran fourth');

}

function fifth(){

logOut('ran fifth');

}

function logOut(val){

const ss = SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();

ss.appendRow([val]);

}

# Lesson 14

function onOpen(){

const ui = SpreadsheetApp.getUi();

ui.createMenu('adv')

.addItem('copy','copytolog')

.addToUi();

}

function copytolog(){

const ss = SpreadsheetApp.getActiveSpreadsheet()

const sheet = ss.getActiveSheet();

const range = ss.getSelection().getActiveRange();

const data = range.getValues();

let sheetLog = ss.getSheetByName('log');

if(sheetLog == null){

sheetLog = ss.insertSheet();

sheetLog.setName('log');

}

const newRange = sheetLog.getDataRange();

const startRow = newRange.getLastRow() +1;

const setRange = sheetLog.getRange(startRow,1,range.getNumRows(),range.getNumColumns());

setRange.setBackground('red');

setRange.setValues(data);

sheetLog.appendRow([startRow]);

}

# Lesson 15

function onOpen(){

const ui = SpreadsheetApp.getUi();

ui.createMenu('adv')

.addItem('alert','popUp1')

.addItem('prompt1','popUp2')

.addItem('prompt2','popUp3')

.addToUi();

}

function popUp3(){

const ui = SpreadsheetApp.getUi();

const rep = ui.prompt('Do you like Apps Script rate 1-5?',ui.ButtonSet.YES\_NO\_CANCEL);

logVal(rep.getSelectedButton());

if(rep.getSelectedButton() == ui.Button.YES){

logVal('YES User rated ' + rep.getResponseText());

}else if(rep.getSelectedButton() == ui.Button.NO){

logVal('NO User rated ' + rep.getResponseText());

}else{

logVal('User Cancel');

}

}

function popUp2(){

const ui = SpreadsheetApp.getUi();

const rep = ui.prompt('Tell me your name?');

logVal(rep.getSelectedButton());

if(rep.getSelectedButton() == ui.Button.OK){

logVal(rep.getResponseText());

}else{

logVal('Prompt Closed');

}

}

function popUp1(){

const ui = SpreadsheetApp.getUi();

const rep = ui.alert('confirm','Do you agree',ui.ButtonSet.YES\_NO);

logVal(rep);

if(rep == ui.Button.YES){

logVal('yes was pressed');

}else{

logVal('no was pressed');

}

}

function logVal(val){

const ss = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('log');

ss.appendRow([val]);

}

# Lesson 16

const GLVAL = 'Testing Global Value';

function onOpen(){

const ui = SpreadsheetApp.getUi();

ui.createMenu('adv')

.addItem('showModal1','modal1')

.addItem('showModal2','modal2')

.addItem('showModal3','modal3')

.addToUi();

}

function modal3(){

const ui = SpreadsheetApp.getUi();

const html = HtmlService.createHtmlOutput('<h1>Hello World</h1><p>Tested</p>');

html.setHeight(500);

html.setWidth(800);

ui.showDialog(html);

}

function modal2(){

const ui = SpreadsheetApp.getUi();

//const html = HtmlService.createHtmlOutput('<h1>Hello World</h1><p>Tested</p>');

const html = HtmlService.createHtmlOutputFromFile('temp');

html.setHeight(500);

html.setWidth(800);

ui.showModelessDialog(html,'Modeless');

}

function modal1(){

const ui = SpreadsheetApp.getUi();

//const html = HtmlService.createHtmlOutput('<h1>Hello World</h1><p>Tested</p>');

const html = HtmlService.createTemplateFromFile('temp1').evaluate();

ui.showModalDialog(html,'test 1');

}

function logVal(val){

const ss = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('log');

ss.appendRow([val]);

}

<!DOCTYPE html>

<html>

<head>

<base target="\_top">

<style>

body{

background:red;

}

</style>

</head>

<body>

<h1>Hello</h1>

<p>Testing</p>

<script>

document.querySelector('h1').innerHTML = 'JAVASCRIPT';

</script>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<base target="\_top">

</head>

<body>

<h1>Test 1</h1>

<?= GLVAL ?>

</body>

</html>

# Lesson 17

const GLVAL = 'Testing Global Value';

let COUNTER = 0;

function onOpen(){

PropertiesService.getDocumentProperties().setProperty('cnt',0);

const ui = SpreadsheetApp.getUi();

ui.createMenu('adv')

.addItem('showSide1','side1')

.addItem('showSide2','side2')

.addItem('showSide3','side3')

.addToUi();

}

function side1(){

const ui = SpreadsheetApp.getUi();

const html = HtmlService.createHtmlOutput('<h1>Hello World</h1><p>Tested</p>');

ui.showSidebar(html);

}

function side2(){

const ui = SpreadsheetApp.getUi();

const html = HtmlService.createHtmlOutputFromFile('temp');

ui.showSidebar(html);

}

function side3(){

COUNTER = PropertiesService.getDocumentProperties().getProperty('cnt');

COUNTER++;

PropertiesService.getDocumentProperties().setProperty('cnt',COUNTER);

const ui = SpreadsheetApp.getUi();

const html = HtmlService.createTemplateFromFile('temp1').evaluate();

ui.showSidebar(html);

}

# Lesson 18

function addContent(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheet = ss.getSheetByName('test');

Logger.log(sheet);

let startPos = 5;

let startVal = sheet.getRange(startPos,1).getValue();

sheet.getRange(startPos,1).setValue(startVal + ' START');

sheet.insertRowAfter(startPos);

sheet.getRange(startPos+1,1).setValue('AFTER');

sheet.insertRowBefore(startPos);

sheet.getRange(startPos,1).setValue('BEFORE');

let tempArr = [sheet.getLastRow()+1,'test',2,'hello world'];

sheet.appendRow(tempArr);

}

# Lesson 19

function prepender(val,sheet){

sheet.insertRowBefore(1);

let cloneArr = val.map((x)=>x);

cloneArr.push('START');

const range = sheet.getRange(1,1,1,cloneArr.length);

range.setValues([cloneArr]);

}

function addContent2(){

const id = '1looDMwg\_ztAb2tiuRx6Xk3MXFtQ4yLc1vumVbiSnzu0';

const ss = SpreadsheetApp.openById(id);

const sheet = ss.getSheetByName('test');

let tempArr = [sheet.getLastRow()+1,'NEW CONTENT'];

prepender(tempArr,sheet);

tempArr.push('END');

sheet.appendRow(tempArr);

}

# Lesson 20

function SELVALA1(){

return SpreadsheetApp.getActive().getActiveRange().getA1Notation();

}

function addForm(){

const ss = SpreadsheetApp.getActive();

const sheet = ss.getSheetByName('test');

const range = sheet.getRange('C1:C15');

range.setFormula('=SUM(A1:B1)');

range.setFontColor('red');

range.setBackground('pink');

}

# Lesson 21

function onOpen(){

const ui = SpreadsheetApp.getUi();

ui.createMenu('ADV')

.addItem('addComment','adder')

.addToUi();

}

function adder(){

const ui = SpreadsheetApp.getUi();

const cell = SpreadsheetApp.getActive().getActiveSheet().getActiveCell();

const rep = ui.prompt('What comment would you like to add?');

if(rep.getSelectedButton() == ui.Button.OK){

cell.setComment(rep.getResponseText());

}

}