//2022.01.27

//แนะนำจาวาสคริปท์เบื้องต้นhttps://developers.google.com/earth-engine/tutorials/tutorial\_js\_01

//1.การคอมเม้นท์ เขียนกำกับขั้นตอนหรือสิ่งที่กำลังทำ เพื่อจะได้เตือนความจำ

//สองขีดใช้คอมเม้นท์แต่ละบรรทัด

/\* ใช้คอมเม้นท์

หลายบรรทัด

\*/

//2.Hello World!

print('Hello World!');

//3.ชนิดข้อมูล

//-string

var greetString = 'สวัสดีครับ!';//ตัวแปร string

print(greetString);

//-number

var number = 42;

print('The answer is:', number);

//Lists

// ใช้วงเล็บ [] เพื่อสร้าง Lists

var listOfNumbers = [0, 1, 1, 2, 3, 5];

print('List of numbers:', listOfNumbers);//ลอง listOfNumbers[1]

//ลิสต์สามารถใช้เก็บข้อความและชนิดข้อมูลอื่นได้

var listOfStrings = ['a', 'b', 'c', 'd'];

print('List of strings:', listOfStrings);

//Objects

//ออปเจ็กในจาวาสคริปท์คือดิกชันนารี key:value ใช้ {}เพื่อสร้าง dictionary

// Use curly brackets {} to make a dictionary of key:value pairs.

var object = {

foo: 'bar',

baz: 13,

stuff: ['this', 'that', 'the other thing']

};

print('Dictionary:', object);

// การเข้าถึงดิกชันารีด้วย []

print('Print foo:', object['foo']);

// การเข้าถึงดิกชันนารีด้วยจุด .

print('Print stuff:', object.stuff);

//funtions

//ใช้ฟังค์ชันเพื่อให้โค้ดอ่านง่ายและนำไปดัดแปลงเพื่อใช้ได้ง่ายในภายหลัง

//ตัวอย่าง syntax

/\*

var myFunction = function(parameter1, parameter2, parameter3) {

statement;

statement;

statement;

return statement;

};

\*/

//ตัวอย่างการใช้งานฟังก์ชัน

// สร้างฟังก์ชันที่ชื่อ reflect โดยมีพารามิเตอร์ที่ชื่อ element

var reflect = function(element) {

// Return the argument.

return element;

};

print('A good day to you!', reflect('Back at you!'));

//ตัวอย่างการใช้ฟังก์ชันที่มีอยู่แล้วใน GEE

var aString = ee.Algorithms.String(42);//อ่านจาก DOCs เป็นการแปลงตัวเลขเป็นข้อความ

print('age is : ', aString);

//4.Earth Engine Objects

//https://developers.google.com/earth-engine/tutorials/tutorial\_js\_02

//การใช้ออปเจ็คใส่ไปใน container เพื่อส่งไปยังเซิร์ฟเวอร์และประมวลผลในระบบกุเกิ้ลคลาวด์

//4.1 string

// กำหนด string แล้วใส่ไปใน EE container.

var aString = 'To the cloud!'; //string

var eeString = ee.String(aString); //container

print('Where to?', eeString); //กุเกิ้ลจะประมวลผล

//กำหนดข้อความแล้วใส่ไปใน container พร้อมกันเลย

var serverString = ee.String('This is on the server.');

print('String on the server:', serverString);

//4.2 numbers

// สร้าง number object ในระบบเซิร์ฟเวอร์ ้ด้วยการใช้เมธอดเพื่อสร้างค่าคงที่ในเซิร์ฟเวอร์

var serverNumber = ee.Number(Math.E);

print('e=', serverNumber);

//เรียก ee.String และ ee.Number ว่า เมธอดชนิด constructors ที่ต้องการค่า argrument/parameters

//ให้ใช้ Earth Engine methods กับตัว Earth Engine object ที่สร้างขึ้นมาดังนี้

var logE = serverNumber.log(); // ไม่ใช้ Math.log()

print('log(e)=', logE); // log() เป็นเมธอดที่ใช้สำหรับ ee.Number object

//4.3 Lists

// หากจะสร้างลิสต์ใน EE ให้ใช้ ee.List

// การสร้างลิสต์แบบ hard way.

var eeList = ee.List([1, 2, 3, 4, 5]);

// การสร้างลิสต์แบบ easy way!

var sequence = ee.List.sequence(1, 5);

print('Sequence:', sequence);

//หากจะนำค่าออกจากลิสต์ที่อยู่ใน EE ต้องใช้เมธอดเช่นกันที่ชื่อ get()

// ใช้เมธอดget() เพื่อดึงค่าออกมาจาก ee.List

var value = sequence.get(2);

print('Value at index 2:', value);

//4.4 Dictionaries

// สามารถสร้างดิกชันนารีใน Earth Engine ได้ด้วย ee.dictionary

// Make a Dictionary on the server.

var dictionary = ee.Dictionary({

e: Math.E,

pi: Math.PI,

phi: (1 + Math.sqrt(5)) / 2

});

// Get some values from the dictionary.

print('Euler:', dictionary.get('e'));

print('Pi:', dictionary.get('pi'));

print('Golden ratio:', dictionary.get('phi'));

// Get all the keys:

print('Keys: ', dictionary.keys());

//4.6 Dates

//สร้าง date object ได้จาก ee.date

// Define a date in Earth Engine.

var date = ee.Date('2015-12-31');

print('Date:', date);

// Get the current time using the JavaScript Date.now() method.

var now = Date.now();

print('Milliseconds since January 1, 1970', now);

// Initialize an ee.Date object.

var eeNow = ee.Date(now);

print('Now:', eeNow);

//4.7 Digression: passing parameters by name

//การสร้างออปเจ็ก date ด้วยการใส่ค่าวันเดือนปีเข้าไปได้เลย

var aDate = ee.Date.fromYMD(2017, 1, 13);

print('aDate:', aDate);

//หรืออีกวิธีสามารถใส่แบบใช้ key ก็ได้ จะช่วยให้อ่านง่ายด้วย

var theDate = ee.Date.fromYMD({

day: 13,

month: 1,

year: 2017

});

print('theDate:', theDate);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//5.การใช้ for loops ใน EE https://developers.google.com/earth-engine/tutorials/tutorial\_js\_03

//ไม่มี for loop ใน GEE แต่ให้ใช้ map() แทน

//ตัวอย่างการยกกำลังแต่ละอิลิเมนท์ใน list ด้วยการใช้ map()

// สร้างลิสต์ที่มีเลข 1 ถึง 10

var myList = ee.List.sequence(1, 10);

// ใช้ map() เพื่อกระทำในแต่ละอิลิเมนท์ โดยใช้ในรูปของฟังก์ชันที่ define ไว้ก่อนแล้ว

var computeSquares = function(number) {

// We define the operation using the EE API.

return ee.Number(number).pow(2);

};

// ใช้ฟังก์ชันด้วยการใช้ map()

var squares = myList.map(computeSquares);

print(squares); // [1, 4, 9, 16, 25, 36, 49, 64, 81]

//6.เงื่อนไขif/else

//ไม่นิยมใช้ ee.Algorithms.If() เพราะช้า แต่จะใช deferred execution

// ตัวอย่างต้องการค่าที่เป็นเลขคี่ ด้วยการใช้วิธีการหาร mod (2)

// หลังจากหาร 2 แล้ว หากได้ค่าเป็น 0 แสดงว่าเป็นเลขคู่ หากได้ค่าเป็น 1 แสดงว่าเป็นเลขคี่

// แล้วจึงนำ 2 ไปคูณกับเศษเหลือ

// สุดท้ายจะเหลือแต่จำนวนคี่ ด้วยการลบจำนวนที่เท่ากับ 0 ออกไป

var getOddNumbers = function(number) {

number = ee.Number(number); // Cast the input to a Number so we can use mod.

var remainder = number.mod(2);

return number.multiply(remainder);

};

var newList = myList.map(getOddNumbers);

// Remove the 0 values.

var oddNumbers = newList.removeAll([0]);

var squares = oddNumbers.map(computeSquares);

print(squares); // [1, 9, 25, 49, 81]

//6.1 สามารถนำแนวคิดด้านบนมาใช้ในงานจริง เช่น ต้องการกระทำการแบ่งชุดข้อมูลออกเป็นสองชุดแล้วทำการประมวลผล แล้วจึงนำมารวมกันทีหลัง

var collection = ee.ImageCollection('LANDSAT/LC08/C01/T1\_TOA');

// Divide the collection into 2 subsets and apply a different algorithm on them.

var subset1 = collection.filter(ee.Filter.lt('SUN\_ELEVATION', 40));

var subset2 = collection.filter(ee.Filter.gte('SUN\_ELEVATION', 40));

var processed1 = subset1.map(function(image) {

return image.multiply(2);

});

var processed2 = subset2;

// Merge the collections to get a single collection.

var final = processed1.merge(processed2);

print('Original collection size', collection.size());

print('Processed collection size', final.size());

//7.การคำนวณแบบสะสม Cumulative Iteration

//ใช้ผลลัพธ์จากรอบคำนวณปัจจุบันในรอบการคำนวณใหม่

var algorithm = function(current, previous) {

previous = ee.List(previous);

var n1 = ee.Number(previous.get(-1));

var n2 = ee.Number(previous.get(-2));

return previous.add(n1.add(n2));

};

// คำนวณ 10 รอบ

var numIteration = ee.List.repeat(1, 10);

var start = [0, 1];

var sequence = numIteration.iterate(algorithm, start);

print(sequence); // [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

/\*

ให้นิสิตฝึกทบทวน ในแต่ละข้อ คิดโจทย์เอง พิมพ์ลองเล่น ลองดัดแปลงโค้ดดู

\*/