Assignment-8

/\*     JavaScript Fundamentals – Part 1    \*/

/\* LECTURE: Values and Variables   \*/

/\* 1. Declare variables called 'country', 'continent' and 'population' and assign their values according to your own country (population in millions)

   2. Log their values to the console    \*/

    let country="India";

    let continent="Asia";

    let population= 1417;

    console.log("Country Name : "+ country)

    console.log("Continent Name: "+continent);

    console.log(population+" Millions");

/\* LECTURE: Data Types

1. Declare a variable called 'isIsland' and set its value according to your country. The variable should hold a Boolean value. Also declare a variable 'language', but don't assign it any value yet

2. Log the types of 'isIsland', 'population', 'country' and 'language' to the console   \*/

//already in first problem we declared the country and population so we can use those variables here

let isIsland=false;

let language;

console.log("country Name : "+country)

console.log(country +" isIsland : "+isIsland);

console.log(population+" Millions population are in "+country);

console.log(language);

/\*LECTURE: let, const and var

1. Set the value of 'language' to the language spoken where you live (some countries have multiple languages, but just choose one)

2. Think about which variables should be const variables (which values will never change, and which might change?). Then, change these variables to const.

3. Try to change one of the changed variables now, and observe what happens   \*/

 language="English";

 language="Hindi"

 console.log(language);//using let keyword re-assign the value is possible

 const mother\_tongue="Telugu";

 const name9="BhanuPrasad"

//  name9="Bhanu"

 console.log(name9)

 //it will gives an error because const variable not possible to re assign the value it is already constant

 /\*LECTURE: Basic Operators

1. If your country split in half, and each half country would contain half the population, then how many people would live in each half?\*/

 if(country==="India")

    {

        if(population > 0)

            {

                console.log("after split each half would have population is : "+(population/2)+" Millions");

            }

    }

/\*2. Increase the population of your country by 1 and log the result to the console  \*/

  population += 1;

  console.log("After increment the country by 1 : "+ population +" Millions");

/\*3.Finland has a population of 6 million. Does your country have more people than Finland\*/

let finland\_pop= 6;

if(population > finland\_pop)

    {

        console.log("My country have more people than Finland" +" i.e "+ population+ " Millions");

    }

    else{

        console.log("Finland have more people than My country");

    }

/\*4.The average population of a country is 33 million people. Does your country have less people than the average country?   \*/

 let AvgCountry\_pop= 33;

 if(population<AvgCountry\_pop)

    {

        console.log("Average country have more population than my country");

    }

    else{

        console.log("my country population is more than Average country");

    }

/\*5. Based on the variables you created, create a new variable 'description' which contains a string with this format: 'Portugal is in Europe, and its 11 million people speak portuguese'  \*/

 let description='Portugal is in Europe, and its 11 million people speak portuguese';

 /\*LECTURE: Strings and Template Literals

1.Recreate the 'description' variable from the last assignment, this time using the template literal syntax   \*/

let a\_country="Portugal"

let a\_continent="Europe"

let a\_population=11;

let a\_language="Portuguese"

description= `'${a\_country} is in ${a\_continent}, and its ${a\_population} million people speak ${a\_language}.'`

console.log(description)

/\*LECTURE: Taking Decisions: if / else Statements

1.If your country's population is greater that 33 million, log a string like this to the console: 'Portugal's population is above average'. Otherwise, log a string like 'Portugal's population is 22 million below average' (the 22 is the average of 33 minus the country's population)

2. After checking the result, change the population temporarily to 13 and then to 130. See the different results, and set the population back to original

  \*/

  if(a\_population> 33)

    {

        console.log( 'Portugal population is above average' );

    }

    else{

        console.log('Portugal population is 22 million below average');

    }

    a\_population=13;

    if(a\_population> 33)

        {

            console.log( 'Portugal population is above average' );

        }

        else{

            console.log('Portugal population is 22 million below average');

        }

    a\_population=130;

    if(a\_population> 33)

        {

            console.log( 'Portugal population is above average' );

        }

        else{

            console.log('Portugal population is 22 million below average');

        }

        a\_population=11;

/\*LECTURE: Type Conversion and Coercion

Predict the result of these 5 operations without executing them:  '9' - '5'; '19' - '13' + '17';  '19' - '13' + 17;  '123' < 57; 5 + 6 + '4' + 9 - 4 - 2;

2. Execute the operations to check if you were right  \*/

/\*My prediction are

 '9' - '5':

Both '9' and '5' are strings, but the - operator forces them to be converted to numbers.

So, it will be 9 - 5, which is 4.

'19' - '13' + '17':

'19' - '13' will be 19 - 13, which is 6.

Then, 6 + '17' will concatenate 6 (number) with '17' (string), resulting in '617'.

'19' - '13' + 17:

'19' - '13' will be 19 - 13, which is 6.

Then, 6 + 17 will be 23.

'123' < 57:

'123' is a string, and 57 is a number.

In this case, the string '123' will be converted to the number 123.

123 < 57 is false.

5 + 6 + '4' + 9 - 4 - 2:

5 + 6 will be 11.

Then, 11 + '4' will concatenate 11 (number) with '4' (string), resulting in '114'.

Then, '114' + 9 will still result in '1149' (string concatenation).

Finally, '1149' - 4 - 2 will be 1149 - 4 - 2, resulting in 1143.

\*/

 console.log('9' - '5')

 console.log('19' - '13' + '17');

 console.log('19' - '13' + 17);

 console.log('123' < 57);

 console.log(5 + 6 + '4' + 9 - 4 - 2);

 //predictions are correct

 /\*LECTURE: Equality Operators: == vs. ===

1. Declare a variable 'numNeighbours' based on a prompt input like this: prompt('How many neighbour countries does your country have?');

2. If there is only 1 neighbour, log to the console 'Only 1 border!' (use loose equality == for now)

3. Use an else-if block to log 'More than 1 border' in case 'numNeighbours' is greater than 1

4. Use an else block to log 'No borders' (this block will be executed when 'numNeighbours' is 0 or any other value)

5. Test the code with different values of 'numNeighbours', including 1 and 0.

6. Change == to ===, and test the code again, with the same values of 'numNeighbours'. Notice what happens when there is exactly 1 border! Why is this happening?

7. Finally, convert 'numNeighbours' to a number, and watch what happens now when you input 1

8. Reflect on why we should use the === operator and type conversion in this situation   \*/

// Import the prompt-sync module

const prompt = require('prompt-sync')();

let numNeighbours;

// i will commenting the 184 because for below new codes

// numNeighbours = prompt('How many neighbour countries does your country have?');

// using ==

if (numNeighbours == 1) {

  console.log('Only 1 border!');

} else if (numNeighbours > 1) {

  console.log('More than 1 border');

} else {

  console.log('No borders');

}

//changing code == to ===

if (numNeighbours === 1) {

    console.log('Only 1 border!');

  } else if (numNeighbours > 1) {

    console.log('More than 1 border');

  } else {

    console.log('No borders');

  }

// i understand that == is not considering the what datatype it is but === is consider the what type of data and value also

/\*1. Comment out the previous code so the prompt doesn't get in the way

2. Let's say Sarah is looking for a new country to live in. She wants to live in a country that speaks english, has less than 50 million people and is not an island.

3. Write an if statement to help Sarah figure out if your country is right for her. You will need to write a condition that accounts for all of Sarah's criteria. Take your time with this, and check part of the solution if necessary.

4. If yours is the right country, log a string like this: 'You should live in Portugal :)'. If not, log 'Portugal does not meet your criteria :('

5. Probably your country does not meet all the criteria. So go back and temporarily change some variables in order to make the condition true (unless you live in Canada :D)  \*/

let n\_name="sarah";

a\_language="English";

a\_isIsland=false;

a\_population =30;

if(a\_language=="English" && a\_population< 50 )

    {

       console.log('You should live in Portugal' );

    }

    else{

        console.log('Portugal does not meet your criteria :(');

    }

/\*LECTURE: The switch Statement

Use a switch statement to log the following string for the given 'language':

chinese or mandarin: 'MOST number of native speakers!'

spanish: '2nd place in number of native speakers'

 english: '3rd place'

hindi: 'Number 4'

arabic: '5th most spoken language'

for all other simply log 'Great language too :D'  \*/

let inp\_langauge="madarin"

switch(inp\_langauge)

{

    case "chinese":console.log('MOST number of native speakers!');

    break;

    case "madarin":console.log('MOST number of native speakers!');

    break;

    case "spanish":console.log('2nd place in number of native speakers' );

    break;

    case "english":console.log('3rd place' );

    break;

    case "Hindi":console.log('Number 4');

    break;

    case "arabic":console.log('5th most spoken language');

    break;

    default:console.log( 'Great language too :D');

}

/\* LECTURE: The Conditional (Ternary) Operator

1. If your country's population is greater than 33 million, use the ternary operator to log a string like this to the console: 'Portugal's population is above average'. Otherwise, simply log 'Portugal's population is below average'. Notice how only one word changes between these two sentences!

2. After checking the result, change the population temporarily to 13 and then to 130. See the different results, and set the population back to original  \*/

// Original population

a\_population = 11;

let message = a\_population > 33 ? "Portugal's population is above average" : "Portugal's population is below average";

console.log(message);

// Change population to 13 million

a\_population= 13;

message = a\_population > 33 ? "Portugal's population is above average" : "Portugal's population is below average";

console.log(message);

// Change population to 130 million

a\_population = 130;

message = a\_population > 33 ? "Portugal's population is above average" : "Portugal's population is below average";

console.log(message); // Logs "Portugal's population is above average"

// Set population back to original

a\_population = 11;

message = a\_population > 33 ? "Portugal's population is above average" : "Portugal's population is below average";

console.log(message); // Logs "Portugal's population is below average"

/\*   JavaScript Fundamentals – Part 2   \*/

/\* ECTURE: Functions

1. Write a function called 'describeCountry' which takes three parameters: 'country', 'population' and 'capitalCity'. Based on this input, the function returns a string with this format: 'Finland has 6 million people and its capital city is Helsinki'

2. Call this function 3 times, with input data for 3 different countries. Store the returned values in 3 different variables, and log them to the console   \*/

function describeCountry(country, population, capitalCity) {

    return `${country} has ${population} million people and its capital city is ${capitalCity}`;

  }

  // Call the function for three different countries

  let country1 = describeCountry('India', 1217, 'NewDelhi');

  let country2 = describeCountry('Germany', 83, 'Berlin');

  let country3 = describeCountry('Japan', 126, 'Tokyo');

  console.log(country1);

  console.log(country2);

  console.log(country3);

  /\*LECTURE: Function Declarations vs. Expressions

1. The world population is 7900 million people. Create a function declaration called 'percentageOfWorld1' which receives a 'population' value, and returns the percentage of the world population that the given population represents. For example, China has 1441 million people, so it's about 18.2% of the world population

2. To calculate the percentage, divide the given 'population' value by 7900 and then multiply by 100

3. Call 'percentageOfWorld1' for 3 populations of countries of your choice, store the results into variables, and log them to the console

4. Create a function expression which does the exact same thing, called 'percentageOfWorld2', and also call it with 3 country populations (can be the same populations) LECTURE: Arrow Functions 1. Recreate the last assignment, but this time create an arrow function called 'percentageOfWorld3'  \*/

function percentageOfWorld1(World\_population) {

    return (World\_population / 7900) \* 100;

  }

  // Call percentageOfWorld1 for 3 populations and log the results

  let chinaPercentage = percentageOfWorld1(1441);

  let indiaPercentage = percentageOfWorld1(1380);

  let usaPercentage = percentageOfWorld1(331);

  console.log(chinaPercentage);

  console.log(indiaPercentage);

  console.log(usaPercentage);

  //Create a function expression which does the exact same thing, called 'percentageOfWorld2'

  function percentageOfWorld1(World\_population) {

    return (World\_population / 7900) \* 100;

  }

  // Call percentageOfWorld1 for 3 populations and log the results

  let chinaPercentage2 = percentageOfWorld1(1441);

  let indiaPercentage2 = percentageOfWorld1(1380);

  let SouthAfricaPercentage2 = percentageOfWorld1(331);

  console.log(chinaPercentage2);

  console.log(indiaPercentage2);

  console.log(SouthAfricaPercentage2);

  let percentageOfWorld3 = World\_population => (World\_population / 7900) \* 100;

// Call percentageOfWorld3 for 3 populations and log the results

let chinaPercentage3 = percentageOfWorld3(1441);

let indiaPercentage3 = percentageOfWorld3(1380);

let southAfricaPercentage3 = percentageOfWorld3(331);

console.log(chinaPercentage3);

console.log(indiaPercentage3);

console.log(southAfricaPercentage3);

/\*LECTURE: Functions Calling Other Functions

1. Create a function called 'describePopulation'. Use the function type you like the most. This function takes in two arguments: 'country' and 'population', and returns a string like this: 'China has 1441 million people, which is about 18.2% of the world.'

 2. To calculate the percentage, 'describePopulation' call the 'percentageOfWorld1' you created earlier 3. Call 'describePopulation' with data for 3 countries of your choice  \*/

 // Function expression for percentageOfWorld1 (from the previous assignment)

  // Function expression for describePopulation

  const describePopulation = function(country\_Name4,World\_population) {

    const percentage = percentageOfWorld1(World\_population);

    return `${country\_Name4} has ${population} million people, which is about ${percentage}% of the world.`;

  };

  // Call describePopulation for 3 countries and log the results

  console.log(describePopulation('China', 1441));

  console.log(describePopulation('India', 1217));

  console.log(describePopulation('USA', 331));

  /\*LECTURE: Arrow Functions

Recreate the last assignment, but this time create an arrow function called 'percentageOfWorld3'

  \*/

 // Arrow function for percentageOfWorld3

percentageOfWorld3 = word\_population => (word\_population / 7900) \* 100;

// Function expression for describePopulation

   const describePopulation7 = function(country\_Name4, word\_population) {

  const percentage3 = percentageOfWorld3(word\_population);

  return `${country\_Name4} has ${word\_population} million people, which is about ${percentage3}% of the world.`;

};

// Call describePopulation for 3 countries and log the results

console.log(describePopulation7('China', 1441));

console.log(describePopulation7('India', 1380));

console.log(describePopulation7('USA', 331));

/\*LECTURE: Introduction to Arrays

1. Create an array containing 4 population values of 4 countries of your choice. You may use the values you have been using previously. Store this array into a variable called 'populations'

 2. Log to the console whether the array has 4 elements or not (true or false)

3. Create an array called 'percentages' containing the percentages of the world population for these 4 population values. Use the function 'percentageOfWorld1' that you created earlier to compute the 4 percentage values  \*/

 percentageOfWorld1 = population => (population / 7900) \* 100;

// Task 1: Create an array containing 4 population values of 4 countries

const populations = [1441, 1380, 331, 201]; // Population values for China, India, USA, and Brazil respectively

// Task 2: Log whether the array has 4 elements or not

console.log(populations.length === 4); // Should log true

// Task 3: Create an array containing the percentages of the world population for the 4 population values

const percentages = [

  percentageOfWorld1(populations[0]),

  percentageOfWorld1(populations[1]),

  percentageOfWorld1(populations[2]),

  percentageOfWorld1(populations[3])

];

console.log(percentages);

/\*LECTURE: Basic Array Operations (Methods)

1. Create an array containing all the neighbouring countries of a country of your choice. Choose a country which has at least 2 or 3 neighbours. Store the array into a variable called 'neighbours'

 2. At some point, a new country called 'Utopia' is created in the neighbourhood of your selected country. So add it to the end of the 'neighbours' array

3. Unfortunately, after some time, the new country is dissolved. So remove it from the end of the array

4. If the 'neighbours' array does not include the country ‘Germany’, log to the console: 'Probably not a central European country :D'

5. Change the name of one of your neighbouring countries. To do that, find the index of the country in the 'neighbours' array, and then use that index to change the array at that index position. For example, you can search for 'Sweden' in the array, and then replace it with 'Republic of Sweden'.  \*/

// 1: Create an array containing neighbouring countries

let neighbours = ['France', 'Spain', 'Italy'];

// 2: Add 'Utopia' to the end of the 'neighbours' array

neighbours.push('Utopia');

//3: Remove 'Utopia' from the end of the array

neighbours.pop();

// 4: Check if 'neighbours' array includes 'Germany' and log a message accordingly

if (!neighbours.includes('Germany')) {

  console.log('Probably not a central European country :D');

}

// 5: Change the name of one neighbouring country

let index = neighbours.indexOf('Spain'); // Find the index of 'Spain'

if (index !== -1) {

  neighbours[index] = 'Republic of Spain'; // Replace 'Spain' with 'Republic of Spain'

}

// Log the updated 'neighbours' array

console.log(neighbours);

/\*LECTURE: Introduction to Objects

Create an object called 'myCountry' for a country of your choice, containing properties 'country', 'capital', 'language', 'population' and 'neighbours' (an array like we used in previous assignments)  \*/

// Create an object called 'myCountry'

let myCountry = {

    country: 'France',

    capital: 'Paris',

    language: 'French',

    population: 67, // in millions

    neighbours: ['Spain', 'Germany', 'Italy']

  };

  console.log(myCountry);

/\*LECTURE: Dot vs. Bracket Notation

1. Using the object from the previous assignment, log a string like this to the console: 'Finland has 6 million finnish-speaking people, 3 neighbouring countries and a capital called Helsinki.'

2. Increase the country's population by two million using dot notation, and then decrease it by two million using brackets notation.  \*/

// Using the object from the previous assignment

  myCountry = {

    country: 'Finland',

    capital: 'Helsinki',

    language: 'Finnish',

    population: 6, // in million

    neighbours: ['Sweden', 'Norway', 'Russia']

  };

  console.log(`${myCountry.country} has ${myCountry.population} million ${myCountry.language}-speaking people, ${myCountry.neighbours.length} neighbouring countries and a capital called ${myCountry.capital}.`);

  //Increase and decrease the country's population

  myCountry.population += 2; // Increase by two million using dot notation

  myCountry['population'] -= 2; // Decrease by two million using brackets notation

  // Log the updated population

  console.log(`Updated population: ${myCountry.population} million`);

  /\*LECTURE: Object Methods

1. Add a method called 'describe' to the 'myCountry' object. This method will log a string to the console, similar to the string logged in the previous assignment, but this time using the 'this' keyword.

 2. Call the 'describe' method

 3. Add a method called 'checkIsland' to the 'myCountry' object. This method will set a new property on the object, called 'isIsland'. 'isIsland' will be true if there are no neighbouring countries, and false if there are. Use the ternary operator to set the property.  \*/

 // Previous assignment object

   myCountry = {

    country: 'Finland',

    capital: 'Helsinki',

    language: 'Finnish',

    population: 6, // in million

    neighbours: ['Sweden', 'Norway', 'Russia'],

    // 1: Add a 'describe' method to the 'myCountry' object

    describe: function() {

      console.log(`${this.country} has ${this.population} million ${this.language}-speaking people, ${this.neighbours.length} neighbouring countries and a capital called ${this.capital}.`);

    },

    // 3: Add a 'checkIsland' method to the 'myCountry' object

    checkIsland: function() {

      this.isIsland = this.neighbours.length === 0 ? true : false;

    }

  };

  // 2: Call the 'describe' method

  myCountry.describe();

  // Call the 'checkIsland' method

  myCountry.checkIsland();

  // Log the 'isIsland' property

  console.log(myCountry.isIsland);

  /\*LECTURE: Iteration: The for Loop

1.There are elections in your country! In a small town, there are only 50 voters. Use a for loop to simulate the 50 people voting, by logging a string like this to the console (for numbers 1 to 50): 'Voter number 1 is currently voting' \*/

// Simulate voters voting

for (let i = 1; i <= 50; i++) {

  console.log("Voter number "+ i + "is currently voting");

}

/\*LECTURE: Looping Arrays, Breaking and Continuing

1. Let's bring back the 'populations' array from a previous assignment

2. Use a for loop to compute an array called 'percentages2' containing the percentages of the world population for the 4 population values. Use the function 'percentageOfWorld1' that you created earlier

3. Confirm that 'percentages2' contains exactly the same values as the 'percentages' array that we created manually in the previous assignment, and reflect on how much better this solution is  \*/

// Previous assignment populations array

//  populations = [1441, 1380, 331, 201]; // Population values for China, India, USA, and Brazil respectively

// Task 2: Use a for loop to compute an array called 'percentages2'

let percentages2 = [];

for (let i = 0; i < populations.length; i++) {

  percentages2.push(percentageOfWorld1(populations[i]));

}

// Task 3: Confirm that 'percentages2' contains the same values as the 'percentages' array

 const percentages8 = [

  percentageOfWorld1(populations[0]),

  percentageOfWorld1(populations[1]),

  percentageOfWorld1(populations[2]),

  percentageOfWorld1(populations[3])

];

let sameValues = true;

for (let i = 0; i < percentages.length; i++) {

  if (percentages8[i] !== percentages2[i]) {

    sameValues = false;

    break;

  }

}

console.log("Do 'percentages' and 'percentages2' contain the same values?", sameValues);

/\*LECTURE: Looping Backwards and Loops in Loops

1. Store this array of arrays into a variable called 'listOfNeighbours' [['Canada', 'Mexico'], ['Spain'], ['Norway', 'Sweden', 'Russia']];

2. Log only the neighbouring countries to the console, one by one, not the entire arrays. Log a string like 'Neighbour: Canada' for each country

3. You will need a loop inside a loop for this. This is actually a bit tricky, so don't worry if it's too difficult for you! But you can still try to figure this out anyway � �

  \*/

// 1: Store the array of arrays into a variable called 'listOfNeighbours'

let listOfNeighbours = [['Canada', 'Mexico'], ['Spain'], ['Norway', 'Sweden', 'Russia']];

//2: Log only the neighbouring countries to the console, one by one

for (let i = 0; i < listOfNeighbours.length; i++) {

  for (let j = 0; j < listOfNeighbours[i].length; j++) {

    console.log(listOfNeighbours[i][j]);

  }

}

/\*LECTURE: The while Loop

1. Recreate the challenge from the lecture 'Looping Arrays, Breaking and Continuing', but this time using a while loop (call the array 'percentages3')

 2. Reflect on what solution you like better for this task: the for loop or the while loop?  \*/

//   populations = [1441, 1380, 331, 201]; // Population values for China, India, USA, and Brazil respectively

// 1: Recreate the challenge using a while loop to compute an array called 'percentages3'

let percentages3 = [];

let i = 0;

while (i < populations.length) {

  percentages3.push(percentageOfWorld1(populations[i]));

  i++;

}

// Log the 'percentages3' array to the console

console.log(percentages3);