Lab 2 – JavaScript Functions

Use any JavaScript editor (online/offline) to do the tasks.

There are several online JS editors available. You can work with anyone. A few of them are:

https://repl.it/languages/nodejs

https://www.mycompiler.io/new/nodejs

https://www.jdoodle.com/execute-nodejs-online/

https://ideone.com/

OR

Install **Nodejs** (https://nodejs.org/en/download/) and **Visual Studio Code** (https://code.visualstudio.com/download).

You can check if Nodejs is already installed or not through command line:

> node -v

It will return the Nodejs version

Task #1 (JS functions – not arrow functions)

Implement min and max methods that return minimum and maximum value of supplied arguments. Implement your own algorithm to find the minimum and maximum value.
- min(4,8,1,3) // returns 1
- max(4,6,5,3,2) // returns 6

Task #2

Implement a program with four functions (add, subtract, multiply and divide). Each function should have different number of arguments passed.

• First function 'add' should check the undefined arguments within the defined function.

• Second function 'subtract' should use the ES6 default parameter to tackle the same problem.

• Third function 'multiply' should use the ES6 rest parameters to multiply each argument with one another.

• Fourth 'divide' should use the 'Arguments' object to finish the job.

Task #3

Implement a generic method named SolveThis() which takes a JS object. depending upon the key, it performs the operation and returns another object with the result. You can implement this functions with or without Arrow Function.

For Example:

```
SolveThis({sum: [3,2,4], max: [2,4,3,5], min: [5,3,4,3]}) // returns { sum: 9, max: 5, min: 3 }
```

It should perform above implemented functions inside, such as, round, abs, ceil, floor, min, max, random etc

Hint:

// Create Object dynamically with dynamic keys

var res = {};

res['sum'] = 6;

res['min'] = 7;

console.log(res); // output: Object {sum: 6, min: 7}

Task #4

You need to write below JS functions to ES6 arrow functions and assign each to **const** identifier.

```
function profile (name, last) {
  return name + " " + last
}

function ThisIsANumber(num) {
  return num * 45;
}

function retrieveAnEven (myarray) {
  let num = [];
  for (let j of myarray) {
    if (j % 2 === 0) {
      num.push(j);
    }
  }
  return num;
}
```

Submission Guide:

Create separate .js files for each task and zip them.

Create a pdf file with source code and its output

In the end, you must upload two files i.e., a zip file (containing all .js files) and a pdf.