

Web Development Assignment

Name : Apurv Kulkarni
ID : 191080079

Instructions:

1. Make a pdf having the following content:
 - a. Final Answer / Code (Whichever applicable)
 - b. Screenshot of output (when you have to code)

Questions:

1. Familiarize yourself with the Date object and solve the following exercises a.
Get number of days in a month (Input will have month number and year)

Test Cases:

Input: 1 2012

Output: 31

Input: 2 2016

Output: 29

- **Solution:**

Code:

```
function daysInMonth(month, year) {  
    return new Date(year, month, 0).getDate();  
}  
  
console.log(daysInMonth(1, 2012));  
console.log(daysInMonth(2, 2016));
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop
$ node q1.js
31
29
```

b. Get minimum date from an array of dates

Test Case:

['2015/01/01', '2014/05/09', '2014/05/27']

Output: 2014/05/09

- **Code:**

```
function getMinDate(dates){
  return dates.reduce((a,b)=> a<b?a:b);
}

var dates = [];
dates.push(new Date("2015-01-01"));
dates.push(new Date("2014-05-09"));
dates.push(new Date("2014-05-27"));

console.log(getMinDate(dates));
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q1b.js
2014-05-09T00:00:00.000Z
```

2. Familiarize yourself with Arrays and solve the following exercises:

a. Given is an array of numbers. Round the numbers and display the sum using `reduce()` method.

Input: [1.1, 2.3, 15.5, 4.7]

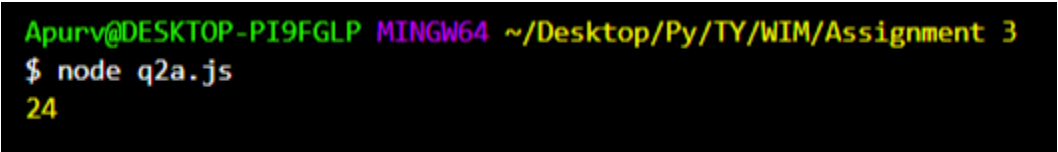
Output: 24

- **Solution:**

Code:

```
function getAns(arr){  
  return arr.reduce((a, b) => a + Math.round(b), 0);  
}  
  
var data = [1.1, 2.3, 15.5, 4.7];  
  
console.log(getAns(data));
```

Output:

A terminal window with a black background and yellow and green text. The prompt is 'Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3'. The command entered is '\$ node q2a.js'. The output is '24'.

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3  
$ node q2a.js  
24
```

b. Given an array of names and ages of people applying for driver's licenses. Return the names of people eligible to apply (18 or above) using the `filter()` method.

Input: [{name: "Mary", age: 21},{name:"John", age: 17},{name:"Mark", age: 24},{name: "Susan", age: 16},{name: "Jacob", age: 19}]

Output: Mary, Mark, Jacob

- **Solution:**

Code:

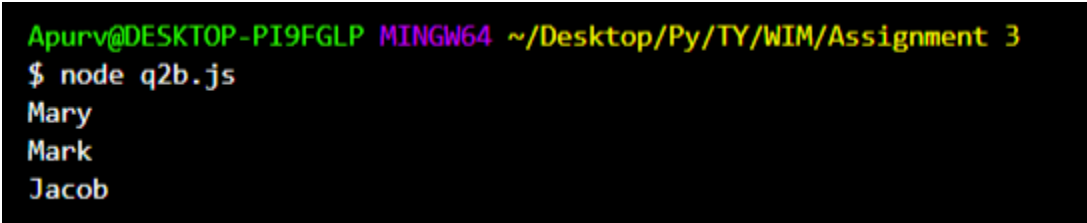
```
function eligible(data){
  return data.filter((person) => person.age>=18);
}

var data = [{name: "Mary", age: 21}, {name: "John", age: 17}, {name:"Mark", age: 24},
{name:"Susan", age: 16}, {name: "Jacob", age: 19}];

data = eligible(data);

for(let i=0; i<data.length; i++){
  console.log(data[i].name);
}
```

Output:



```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q2b.js
Mary
Mark
Jacob
```

- c. Given an array of fruits. Perform slice() operations to get the required output. (There could be multiple ways to obtain an output)

Input: [apples, oranges, bananas, guavas, berries, peaches]

i) Output 1: apples, oranges, bananas

Code:

```
var data = ["apples", "oranges", "bananas", "guavas", "berries", "peaches"];
var ans = data.splice(0, 3);

for(let i=0; i<ans.length; i++){
  console.log(ans[i]);
}
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q2c1.js
apples
oranges
bananas
```

ii) Output 2: peaches

Code:

```
var data = ["apples", "oranges", "bananas", "guavas", "berries", "peaches"];
var ans = data.splice(data.length-1);

for(let i=0; i<ans.length; i++){
  console.log(ans[i]);
}
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q2c2.js
peaches
```

d. Given the above array use splice() operation to change the input to the given output

Output: apples, lemon, guavas, berries, peaches

- **Solution:**

Code:

```
var data = ["apples", "oranges", "bananas", "guavas", "berries", "peaches"];  
data.splice(1, 2); // Removing oranges, bananas from array  
data.splice(1, 0, "lemon"); // Inserting lemon at index 1  
for(let i=0; i<data.length; i++){  
  console.log(data[i]);  
}
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3  
$ node q2d.js  
apples  
lemon  
guavas  
berries  
peaches
```

3. Variables and Data Types:

```
a) var num = 6;
    let num2 = 6;

    if (num===num2)
    {
        console.log("yes");
    }
    else
    {
        console.log("No")
    }
```

What will be the output?

Solution:

Output is **yes**

A terminal window with a black background and yellow and green text. The prompt is 'Apurv@DESKTOP-PI9FGLP MINGW64 ~'. The command entered is '~/Desktop/Py/TY/WIM/Assignment 3' followed by '\$ node q.js'. The output is 'yes'.

```
b)
var i=0;

if(i===0)
{
    let x=1;
    console.log(x);
    let x=8;
    console.log(x);
}
else
{
    console.log("i has different value")
}
What will be the output?
```

Solution:

The code will not be executed due to an error.

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q.js
C:\Users\Apurva\Desktop\Py\TY\WIM\Assignment 3\q.js:6
let x=8;
    ^

SyntaxError: Identifier 'x' has already been declared
    at wrapSafe (internal/modules/cjs/loader.js:979:16)←[39m
    at Module._compile (internal/modules/cjs/loader.js:1027:27)←[39m
    at Object.Module._extensions..js (internal/modules/cjs/loader.js:1092:10)←[39m
    at Module.load (internal/modules/cjs/loader.js:928:32)←[39m
    at Function.Module._load (internal/modules/cjs/loader.js:769:14)←[39m
    at Function.executeUserEntryPoint [as runMain] (internal/modules/run_main.js:72:12)←[39m
    at internal/main/run_main_module.js:17:47←[39m
```

c) Which is not a data type in Javascript?

- i) Int
- ii) Var
- iii) Let
- iv) const

Answer: const

4. Call back and functions

a. function divideByHalf(sum){
 console.log(Math.floor(sum / 2));
}

function multiplyBy2(sum){
 console.log(sum * 2);
}

function operationOnSum(num1,num2,operation){
 var sum = num1 + num2;
 operation(sum);
}


```
operationOnSum(3, 3, divideByHalf);  
operationOnSum(5, 5, multiplyBy2);
```

What is the output?

Solution:

The output is as shown below:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3  
$ node q.js  
3  
20
```

- b. Write a JavaScript function that accepts a string as a parameter and converts the first letter of each word of the string in upper case.

Example string: 'the quick brown fox'

Expected Output: 'The Quick Brown Fox '

- **Solution:**

Code:

```
function titleCase(str) {  
    var splitStr = str.toLowerCase().split(' ');  
    for (var i = 0; i < splitStr.length; i++) {  
        splitStr[i] = splitStr[i].charAt(0).toUpperCase() + splitStr[i].substring(1);  
    }  
    return splitStr.join(' ');  
}  
  
console.log(titleCase("the quick brown fox"));
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q.js
The Quick Brown Fox
```

5. Objects:

a. let obj1 = {
 firstProp: 1,
 secondProp: {
 innerProp: 2,
 },
 thirdProp: 3

Loop over and print each property of the object

- **Solution:**

Code:

```
let obj1 = { firstProp: 1,  
            secondProp: { innerProp: 2 },  
            thirdProp: 3  
          }  
  
for (const item in obj1) {  
  console.log(obj1[item]);  
}
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q.js
1
{ innerProp: 2 }
3
```

```
b. let obj1 = {
    firstProp: 1,
    secondProp: {
        innerProp: 2,
    },
    thirdProp: 3
}
let obj2 = {
    firstProp: 1,
    secondProp: {
        innerProp: 2,
    },
    thirdProp: 3
}
```

Perform a deep comparison of the objects to check whether they are the same or not.

- **Solution:**

Code:

```
let obj1 = { firstProp: 1,
    secondProp: { innerProp: 2 },
    thirdProp: 3
}

let obj2 = { firstProp: 1,
    secondProp: { innerProp: 2 },
    thirdProp: 3
}

console.log( JSON.stringify(obj1) === JSON.stringify(obj2) );
```

Output:

```
Apurv@DESKTOP-PI9FGLP MINGW64 ~/Desktop/Py/TY/WIM/Assignment 3
$ node q.js
true
```