Name : Vansh Mittal
Contact No. : +91 9927220660
Email : vanshmittal1701@gmail.com
LinkedIn : https://www.linkedin.com/in/vansh-mittal-51432b224/
GitHub : vansh1701 (github.com)

Global Trend Programming Profile Assessment Questions

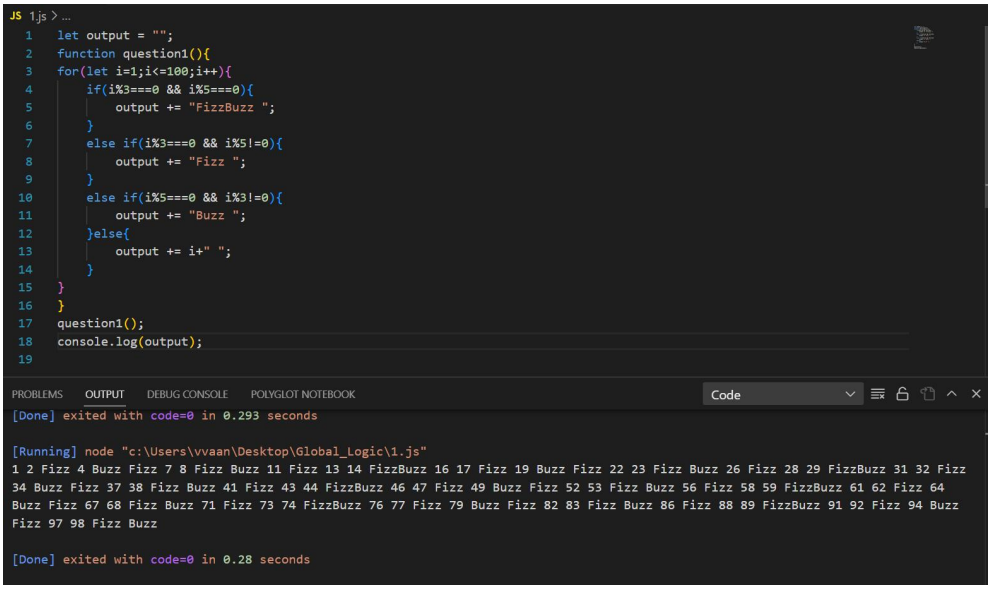
Questions

1. Write a function that prints the numbers from 1 to 100. But for multiples of three, print "Fizz" instead of the number, and for the multiples of five, print "Buzz". For numbers that are multiples of both three and five, print "FizzBuzz".

CODE:

```
let output = "";
function question1(){
for(let i=1;i<=100;i++){
    if(i%3===0 && i%5===0){
        output += "FizzBuzz ";
    }
    else if(i%3===0 && i%5!=0){
        output += "Fizz ";
    }
    else if(i%5===0 && i%3!=0){
        output += "Buzz ";
    }else{
        output += i+" ";
    }
}
}
question1();
console.log(output);
```

OUTPUT:



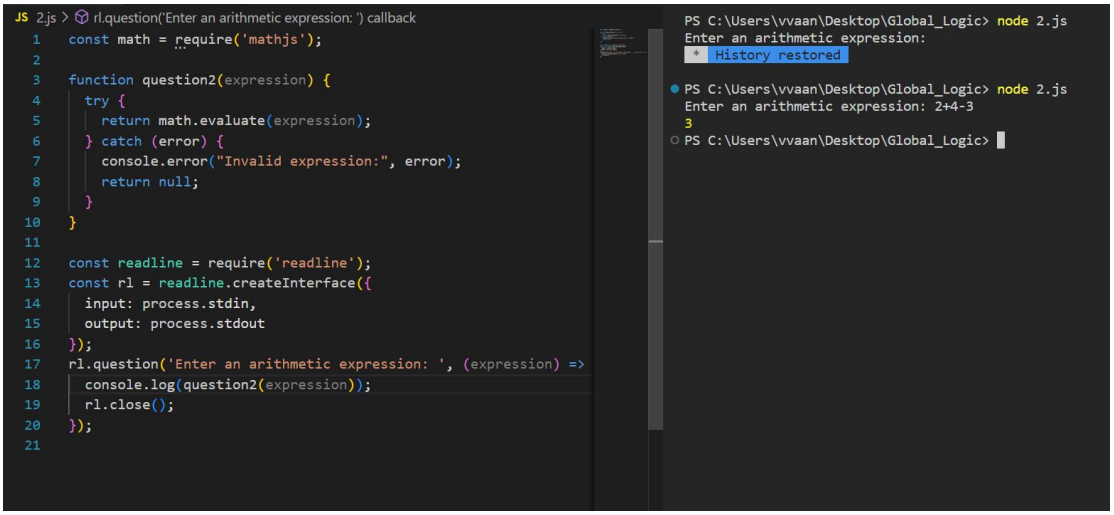
2. Write a function that takes a string input representing a simple arithmetic expression (only addition and subtraction) and returns the result.

CODE :

```
const math = require('mathjs');

function question2(expression) {
  try {
    return math.evaluate(expression);
  } catch (error) {
    console.error("Invalid expression:", error);
    return null;
  }
}
const readline = require('readline');
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});
rl.question('Enter an arithmetic expression: ', (expression) => {
  console.log(question2(expression));
  rl.close();
});
```

OUTPUT :



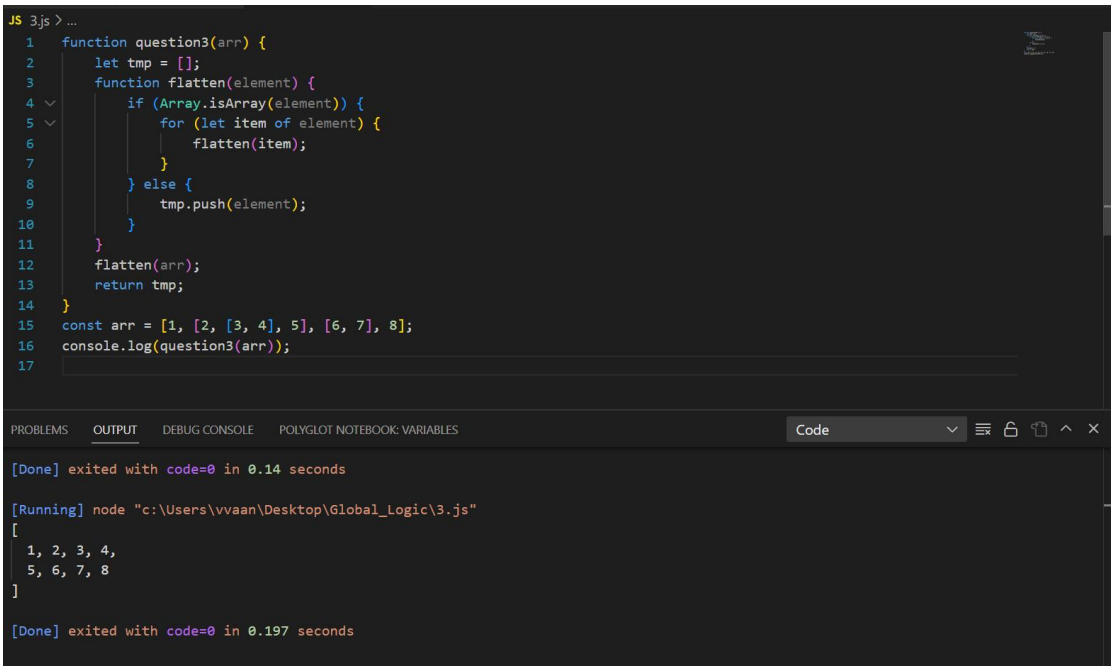
3. Write a function that takes a nested array and returns a flattened array.

CODE :

```
function question3(arr) {
    let tmp = [];
    function flatten(element) {
        if (Array.isArray(element)) {
            for (let item of element) {
                flatten(item);
            }
        } else {
```

```
            tmp.push(element);
        }
    }
    flatten(arr);
    return tmp;
}
const arr = [1, [2, [3, 4], 5], [6, 7], 8];
console.log(question3(arr));
```

OUTPUT :

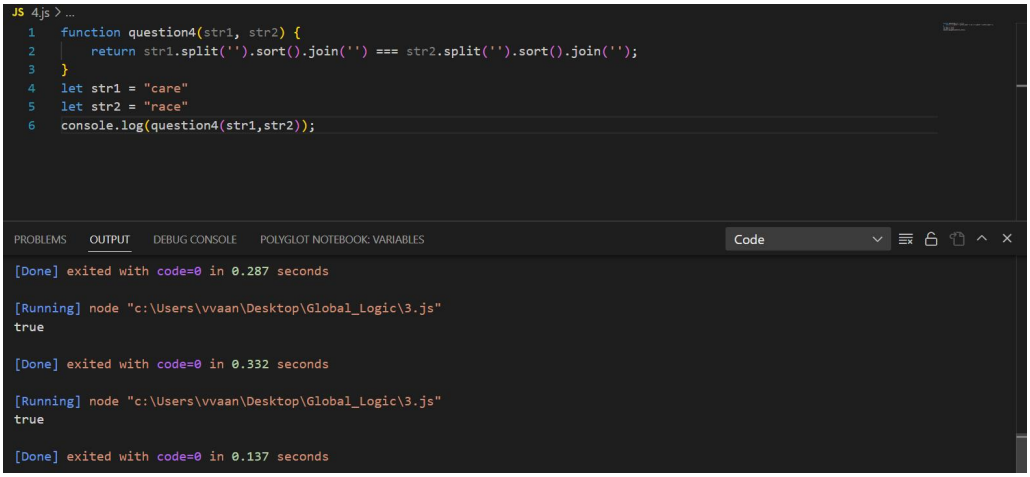


4. Write a function that checks if two given strings are anagrams of each other.

CODE :

```
function areAnagrams(str1, str2) {
return str1.split('').sort().join('') ===
str2.split('').sort().join(''); }
let str1 = "care"
let str2 = "race"
console.log(areAnagrams(str1,str2));
```

OUTPUT :

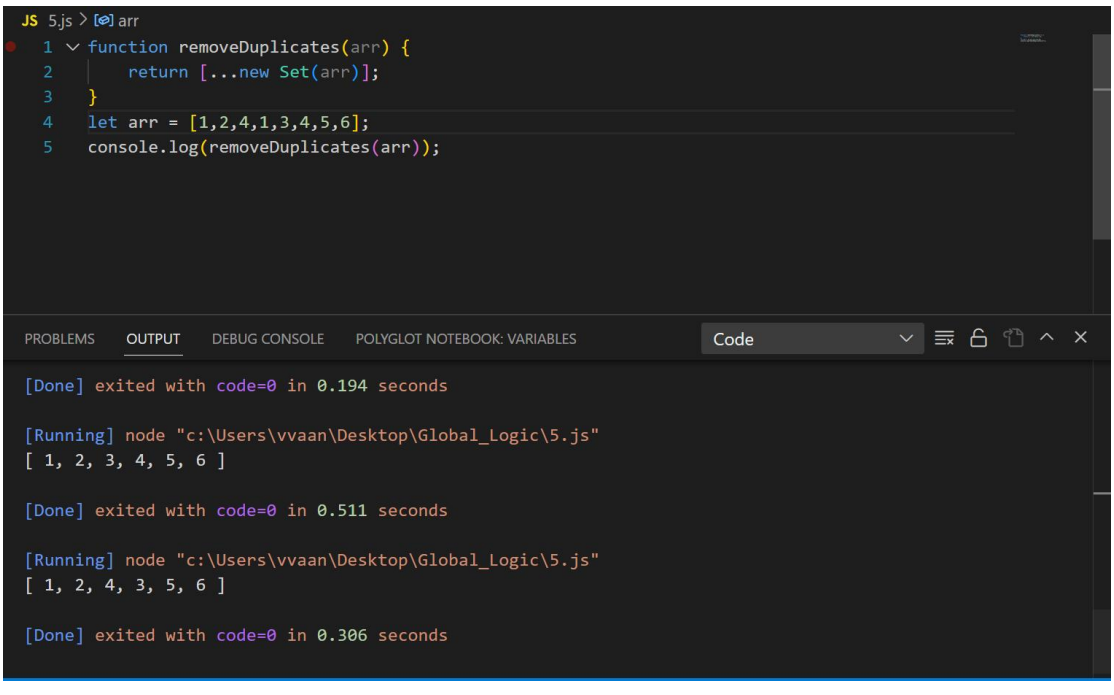


5. Write a function that takes an array and returns a new array with duplicates removed.

CODE :

```
function removeDuplicates(arr) {
    return [...new Set(arr)];
}
let arr = [1,2,4,1,3,4,5,6];
console.log(removeDuplicates(arr));
```

OUTPUT :



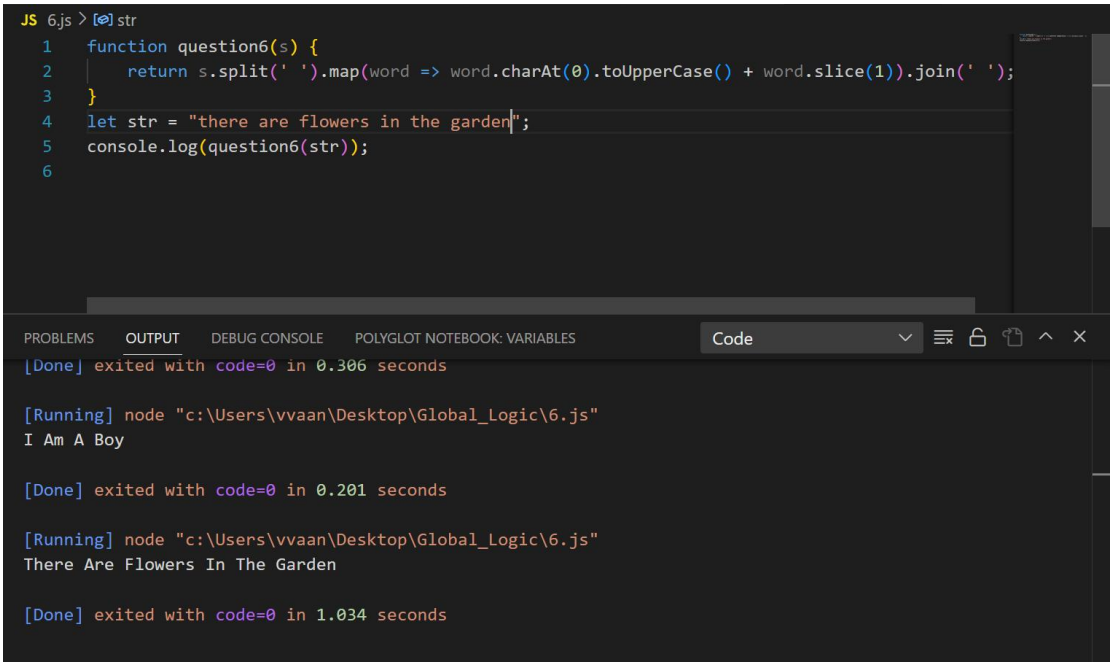
6. Write a function that takes a string and capitalizes the first letter of each word in the string.

CODE :

```
function question6(s) {
```

```
    return s.split(' ').map(word => word.charAt(0).toUpperCase() +
word.slice(1)).join(' ');
}
let str = "there are flowers in the garden";
console.log(question6(str));
```

OUTPUT :

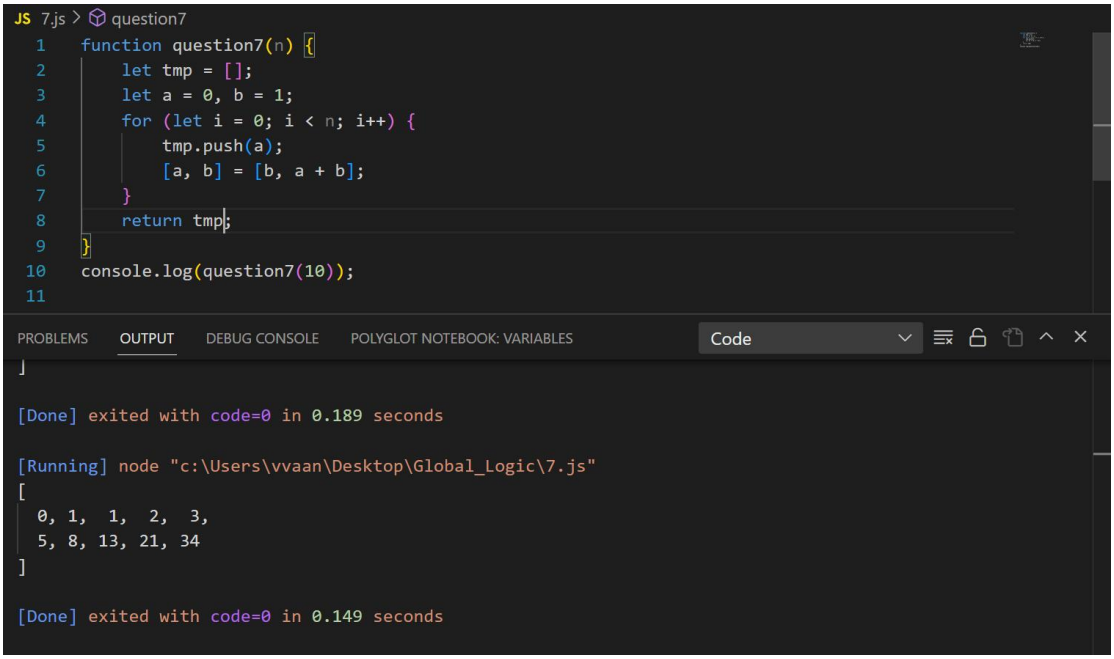


7. Write a function that generates the first `n` numbers of the Fibonacci sequence.

CODE :

```
function question7(n) {
    let tmp = [];
    let a = 0, b = 1;
    for (let i = 0; i < n; i++) {
        tmp.push(a);
        [a, b] = [b, a + b];
    }
    return tmp;
}
console.log(question7(10));
```

OUTPUT :



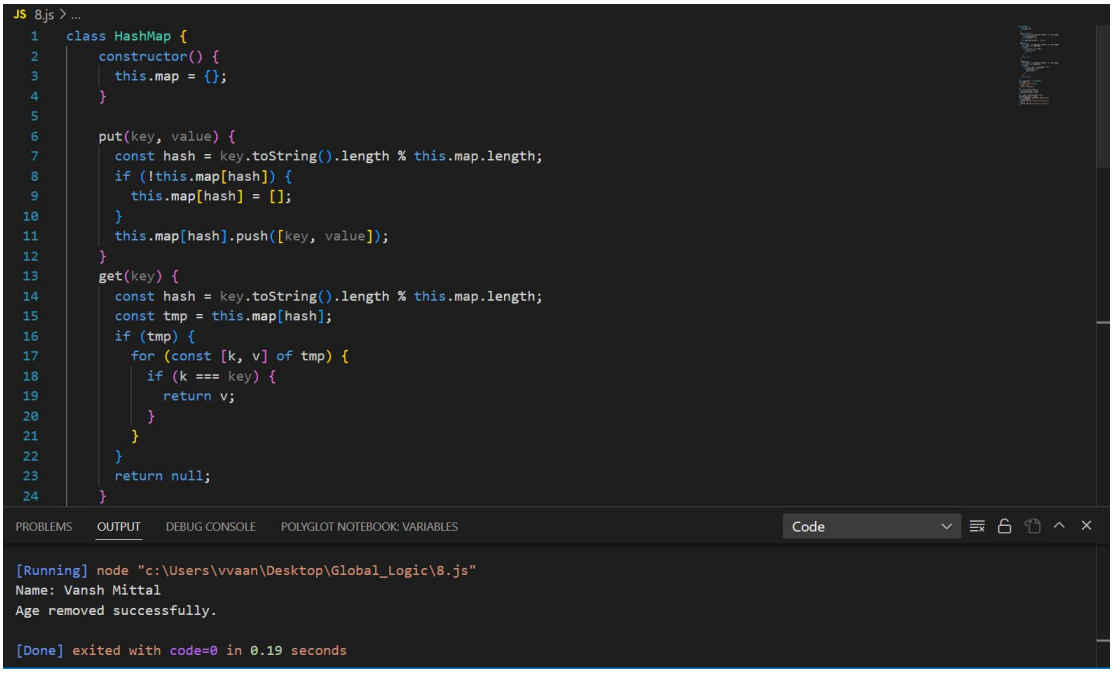
8. Implement a simple HashMap class with *put*, *get*, and *remove* methods.

CODE :

```
class HashMap {
    constructor() {
      this.map = {};
    }
    put(key, value) {
      const hash = key.toString().length % this.map.length;
      if (!this.map[hash]) {
        this.map[hash] = [];
      }
      this.map[hash].push([key, value]);
    }
    get(key) {
      const hash = key.toString().length % this.map.length;
      const tmp = this.map[hash];
      if (tmp) {
        for (const [k, v] of tmp) {
          if (k === key) {
            return v;
          }
        }
      }
      return null;
    }
    remove(key) {
      const hash = key.toString().length % this.map.length;
      const tmp = this.map[hash];
      if (tmp) {
        for (let i = 0; i < tmp.length; i++) {
          if (tmp[i][0] === key) {
            tmp.splice(i, 1);
            return true;
          }
```

```
      }
    }
    return false;
  }
}
const question8 = new HashMap();
const items = [
  ["name", "Vansh Mittal"],
  ["age", 21],
  ["city", "Dehradun"],
];
for (const item of items) {
  const [key, value] = item;
  question8.put(key, value);
}
const name = question8.get("name");
console.log("Name:", name);
const removedAge = question8.remove("age");
if (removedAge) {
  console.log("Age removed successfully.");
} else {
  console.log("Age not found in the map.");
}
```

OUTPUT :



9. Write a function that filters out even numbers from an array.

CODE :

```
function question9(arr) {
    return arr.filter(num => num % 2 !== 0);
}
let arr = [1,2,34,5,6,7,8,9,10]
console.log(question9(arr));
```

OUTPUT :

```
JS 9.js > ...
1   function question9(arr) {
2       return arr.filter(num => num % 2 !== 0);
3   }
4   let arr = [1,2,34,5,6,7,8,9,10]
5   console.log(question9(arr));
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  POLYGLOT NOTEBOOK: VARIABLES  Code

[Running] node "c:\Users\vvaan\Desktop\Global_Logic\9.js"
[ 1, 5, 7, 9 ]

[Done] exited with code=0 in 0.228 seconds
```

10. Write a function that converts a given string to title case (capitalizing the first letter of each word).

CODE :

```
function question10(s) {
    return s.split(' ').map(word => word.charAt(0).toUpperCase() +
word.slice(1)).join(' ');
}
let s = "i want to get settle in new york."
console.log(question10(s));
```

OUTPUT :

```
JS 10.js > ...
1   function question10(s) {
2       return s.split(' ').map(word => word.charAt(0).toUpperCase() + word.slice(1)).join(' ');
3   }
4   let s = "i want to get settle in new york."
5   console.log(question10(s));
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  POLYGLOT NOTEBOOK: VARIABLES  Code

[Running] node "c:\Users\vvaan\Desktop\Global_Logic\10.js"
I Want To Get Settle In New York.

[Done] exited with code=0 in 0.33 seconds
```