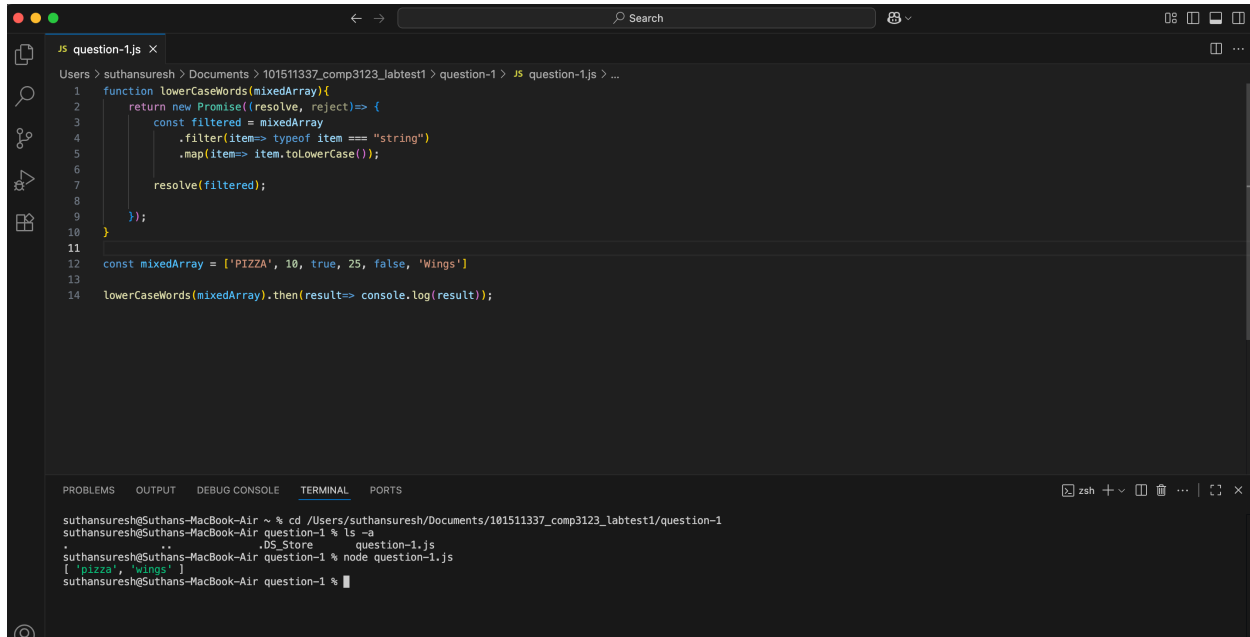


COMP2123 - Lab Test 1

Suthan Sureshkumar - 101511337

Question 1

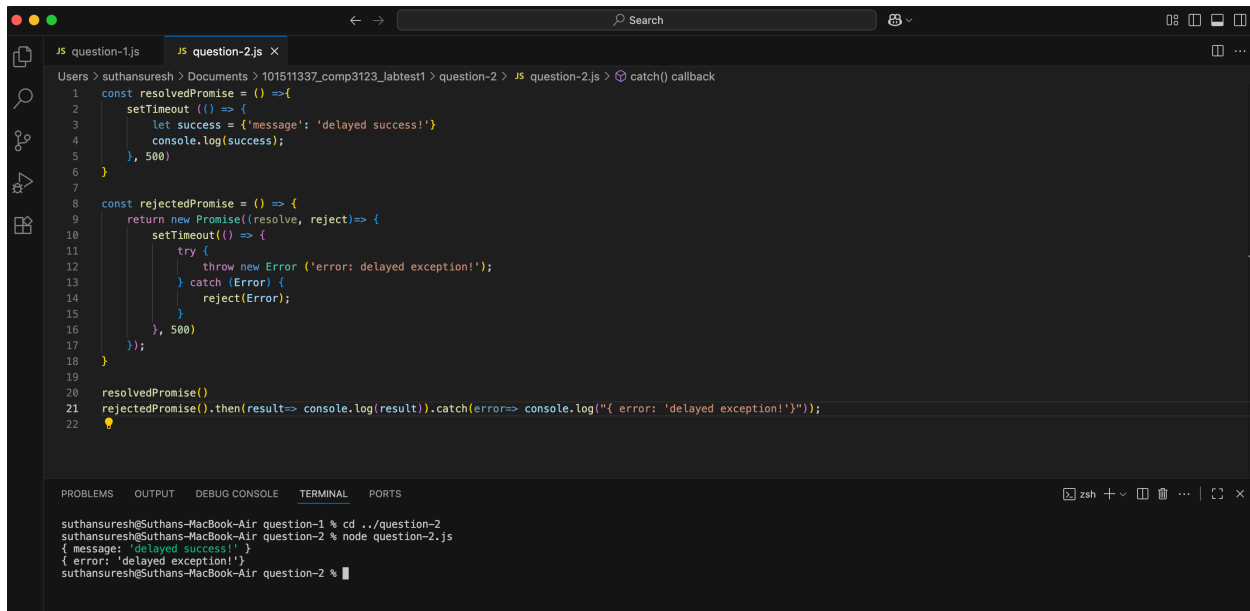


The screenshot shows a VS Code editor window with a file named `question-1.js`. The code defines a function `lowerCaseWords` that takes an array and returns a Promise that resolves to a new array containing only strings from the input array, converted to lowercase. The function is then called with a mixed array, and the result is logged to the console.

```
1 function lowerCaseWords(mixedArray){
2   return new Promise((resolve, reject)=> {
3     const filtered = mixedArray
4       .filter(item=> typeof item === "string")
5       .map(item=> item.toLowerCase());
6
7     resolve(filtered);
8   });
9 }
10
11
12 const mixedArray = ['PIZZA', 10, true, 25, false, 'Wings']
13
14 lowerCaseWords(mixedArray).then(result=> console.log(result));
```

The terminal output shows the command `node question-1.js` being executed, resulting in the output: `['pizza', 'wings']`.

Question 2



The screenshot shows a VS Code editor window with a file named `question-2.js`. The code defines two Promises: `resolvedPromise` and `rejectedPromise`. `resolvedPromise` is a Promise that resolves after 500ms with a success message. `rejectedPromise` is a Promise that rejects after 500ms with an error message. The two promises are then chained together, and the result is logged to the console.

```
1 const resolvedPromise = () => {
2   setTimeout(() => {
3     let success = { message: 'delayed success!' }
4     console.log(success);
5   }, 500)
6 }
7
8 const rejectedPromise = () => {
9   return new Promise((resolve, reject)=> {
10     setTimeout(() => {
11       try {
12         throw new Error('error: delayed exception!');
13       } catch (Error) {
14         reject(Error);
15       }
16     }, 500)
17   });
18 }
19
20 resolvedPromise()
21 rejectedPromise().then(result=> console.log(result)).catch(error=> console.log(`{ error: 'delayed exception!'}`));
22
```

The terminal output shows the command `node question-2.js` being executed, resulting in the output: `{ message: 'delayed success!' }` followed by `{ error: 'delayed exception!' }`.

Question 3