

Question 1

Not yet answered

Marked out of 1.00

What is the output of this code?

```
console.log('start');

setTimeout(() => {
  console.log('timeout');
}, 0);

Promise.resolve().then(() => {
  console.log('promise');
});

console.log('end');
```

- ☐ a. start
end
promise
timeout
promise
- ☐ b. start
end
timeout
promise
- ☐ c. start
end
promise
timeout
- ☐ d. start
promise
end
timeout

Question 2

Not yet answered

Marked out of 1.00

What will the following code output?

```
for (var i = 0; i < 3; i++) {  
  setTimeout(() => console.log(i), 100);  
}
```

- ☐ a. undefined
undefined
undefined
- ☐ b. 0
0
0
- ☐ c. 3
3
3
- ☐ d. 0
1
2

Question 3

Not yet answered

Marked out of 1.00

What does the following print?

```
const obj = {  
  count: 10,  
  inc: function () {  
    setTimeout(() => {  
      this.count++;  
      console.log(this.count);  
    }, 100);  
  }  
};  
obj.inc();
```

- ☐ a. 10
- ☐ b. NaN
- ☐ c. undefined
- ☐ d. 11

Question 4

Not yet answered

Marked out of 1.00

What is the final output?

```
async function f1() {  
  throw 'Error in f1';  
}  
  
async function f2() {  
  try {  
    await f1();  
  } catch (e) {  
    console.log('Caught: ', e);  
  } finally {  
    return 'Done';  
  }  
}  
  
f2().then(console.log);
```

- ☐ a. Caught: Error in f1
Done
- ☐ b. Unhandled Promise Rejection
Done
- ☐ c. Done
Caught: Error in f1
- ☐ d. Error in f1
Done