

1) What is the result of this code in the console

```
for (var i = 0; i < 100; i++) {  
  setTimeout(function() {  
    console.log(i);  
  }, 200);  
}
```

Answer:

```
e > for (var i = 0; i < 100; i++) {  
  setTimeout(function() {  
    console.log(i);  
  }, 200);  
}  
< 100  
100 100  
> |
```

The reason being is that for loop only exits until the set condition breaks. For the above for loop, it will exit once $i = 100$. Therefore, i equals 100 when the for loop exits.

`setTimeout` function in JavaScript usually takes a callback function as an argument. A callback function is a function that is executed after another function finishes running. In this case, it will run after for loop finishes. At this point, i is already 100 when the `console.log(i)` is about to be executed. Due to the closure of JavaScript, the `console.log` has access to the $i = 100$ which is defined as an outer layer of the `setTimeout`. Thus, when the `console.log` is executed, it logs 100

Declare the variable i in the for loop with `let` instead of `var`. The `let` keyword from ES6 creates a separate scope for the code block that allows for loop to print the consecutive variables.

```
> for (let i = 0; i < 100; i++) {  
  setTimeout(function() {  
    console.log(i);  
  }, 200);  
}  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18
```

Question 2 Answer

The result will be "Hello" due to async and await because it fulfilled the pending state of the promise .
if we get the result undefined

Question 3 Explain the difference

Var

The scope of var is globally

```
var name ="Hamad"; var age =25; if (age >20 ){ var name =" Update Hamad shafiq"; console.log(name);  
Update Hamad shafiq
```

Let

Scope of let is blocked scope but it can be updated but not declared

```
> let name ="Hamad"; let height =6; if (height >5 ){ let name =" Update Hamad shafiq"; console.log("within Block working "+name));  
console.log(name);  
within Block working Update Hamad shafiq VM1574:1  
Hamad VM1574:1  
< undefined  
>
```

Const

Scope of const also blocked but it cannot be updated and nor declared

```
> if (true){ const name ="Hamad dev"; const name ="Hamad upate" } console.log(name)  
✖ Uncaught SyntaxError: Identifier 'name' has already been declared
```

Question 4

```
PS E:\project and client Work\React App\node> node server.js  
["hello","another"]  
Example app listening on port 3000
```