The following incorrect code snippet was submitted by your colleague.

const arr = [10, 12, 15, 21];

for (var i = 0; i < arr.length; i++) {

setTimeout(function() {

console.log('The value is : ' + arr [i]);

}, 3000);

}

1) What will the output be of the code snippet above?

//Index: 4, value: undefined

//Index: 4, value: undefined

//Index: 4, value: undefined

//Index: 4, value: undefined

2) What’s wrong with the code snippet ?

Typical closure problem.

When setTimeout() executes after 3s, it's going to use the value of i at that time, because the function closes over the values from the surrounding block.

I had no idea simply changing "var" to "let" would make it so the value of i was "held" until after the timeout finished.

I definitely need to look more into scoping and closures. Thanks again

3) Please write a new code snippet where you fixed the issue?

const arr = [10, 12, 15, 21];

for (let i = 0; i < arr.length; i++) {

setTimeout(function() {

console.log('Index: ' + i + ', value: ' + arr[i]);

}, 3000);

}

4) Explain the cause of the issue to your colleague in plain english.

Answer the above questions in a no