TEMPLE UNIVERSITY

Digital Systems

12.1 Loops

FOX MIS

Loops

Digital Product Management

FOX
MIS

Motivations

- So, what if you wanted to say Hello World ten times?

Source: https://lh3.googleusercontent.com/ALTNv-dmwvGK0MTCa4XTQ9mCD1PMZZNfsaWSCt7PF9gbhpzol1hsHN5x-C6PXvjVldlkbek=s151

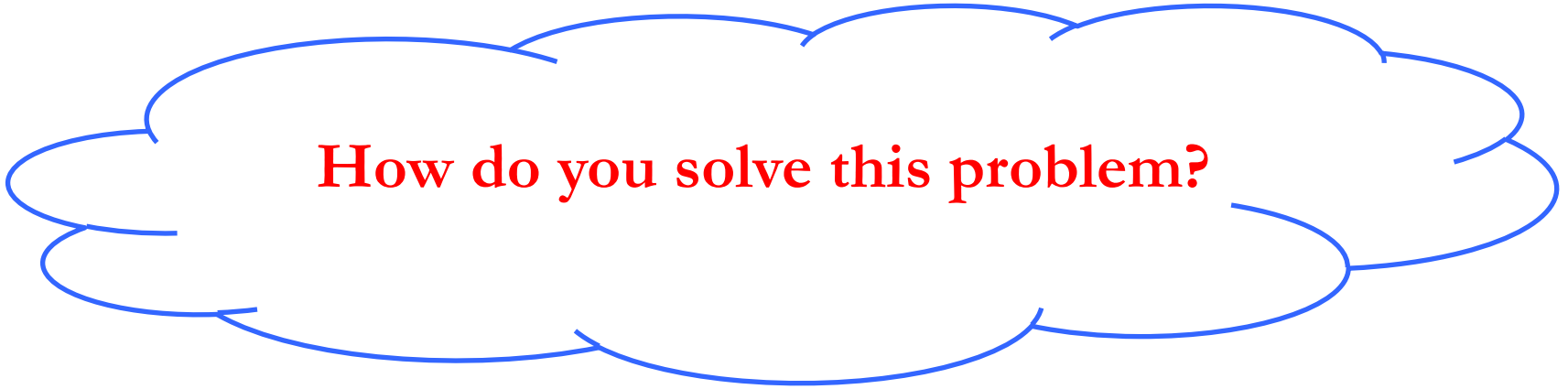
Source: https://lh3.googleusercontent.com/b7Cy0wGI1lYwR4mUZ7jpBLGxd0h0K_qBsK4zh61CXoEd2d1E5d4SV4KVh8a02KIH-tSy=s85

Motivations

10 times

```
1   <!DOCTYPE html>
2   <html>
3   <body>
4
5       <title> Steve Sclarow </title>
6   <script>
7
8   alert ("Hello World");
9   alert ("Hello World");
10  alert ("Hello World");
11  alert ("Hello World");
12  alert ("Hello World");
13  alert ("Hello World");
14  alert ("Hello World");
15  alert ("Hello World");
16  alert ("Hello World");
17  alert ("Hello World");
18
19  </script>
20  </body>
21  </html>
```

Motivations



Motivations

Wouldn't this be easier?

```
for (let i = 0; i < 10; i++)
{
    alert("Hello World!");
}
```

Enter the loop

- A sequence of instructions that is repeated until a certain condition is reached.

- An operation is done, such as getting an item of data and changing it, and then some condition is checked such as whether a counter has reached a prescribed number.

Source: https://lh3.googleusercontent.com/ID3-M72vTnUEBpUxd_l835K2WC_ZUVjSkp7shlUbyX8jRDwPb2i7G-e7e9axmD19FbUEwg=s85

You will often want to repeat some code many MANY times.

A LOOP will help you out.

Meet the loops!

- There are three types of loops you can use to repeat some code:

  - **for** loop
  - **while** loop
  - **do-while** loop

Source: https://lh3.googleusercontent.com/JSm2SSzaaB-sCwjg17mUWXPnca7FtOFItKZPsRby1DolGhyb_Kq_Nx7XB_AV4mMLluY8=s111

for Loop

Digital Product Management

FOX
MIS

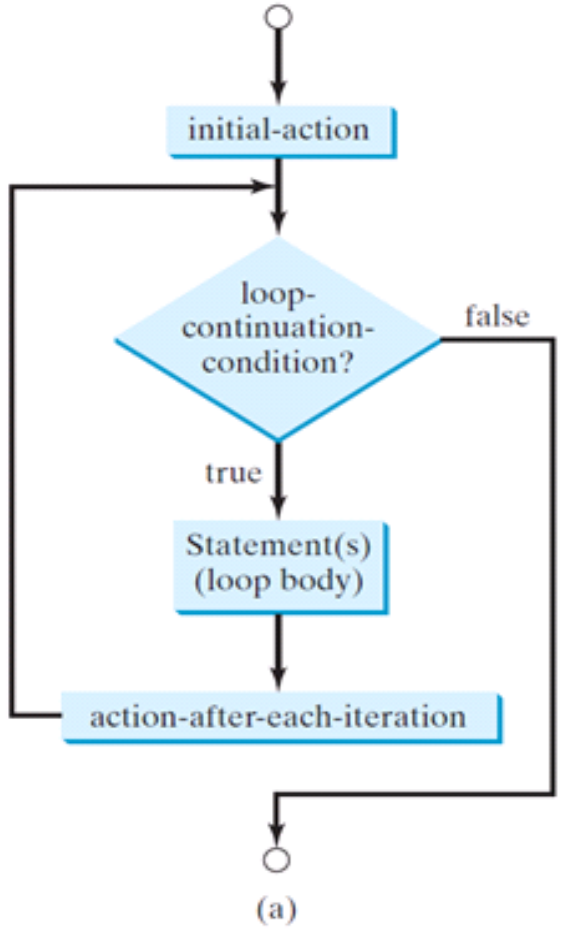
Introducing for Loop

```
let sum=0;

for(let number=1;number<5;number++){
      sum=sum+number;
}
alert("The sum is "+sum);
```

for Loop Flow Chart

```
for (initial-action; loop-continuation-condition; action-after-each-iteration)
{
  // loop body;
  Statement(s);
}
```



(a)

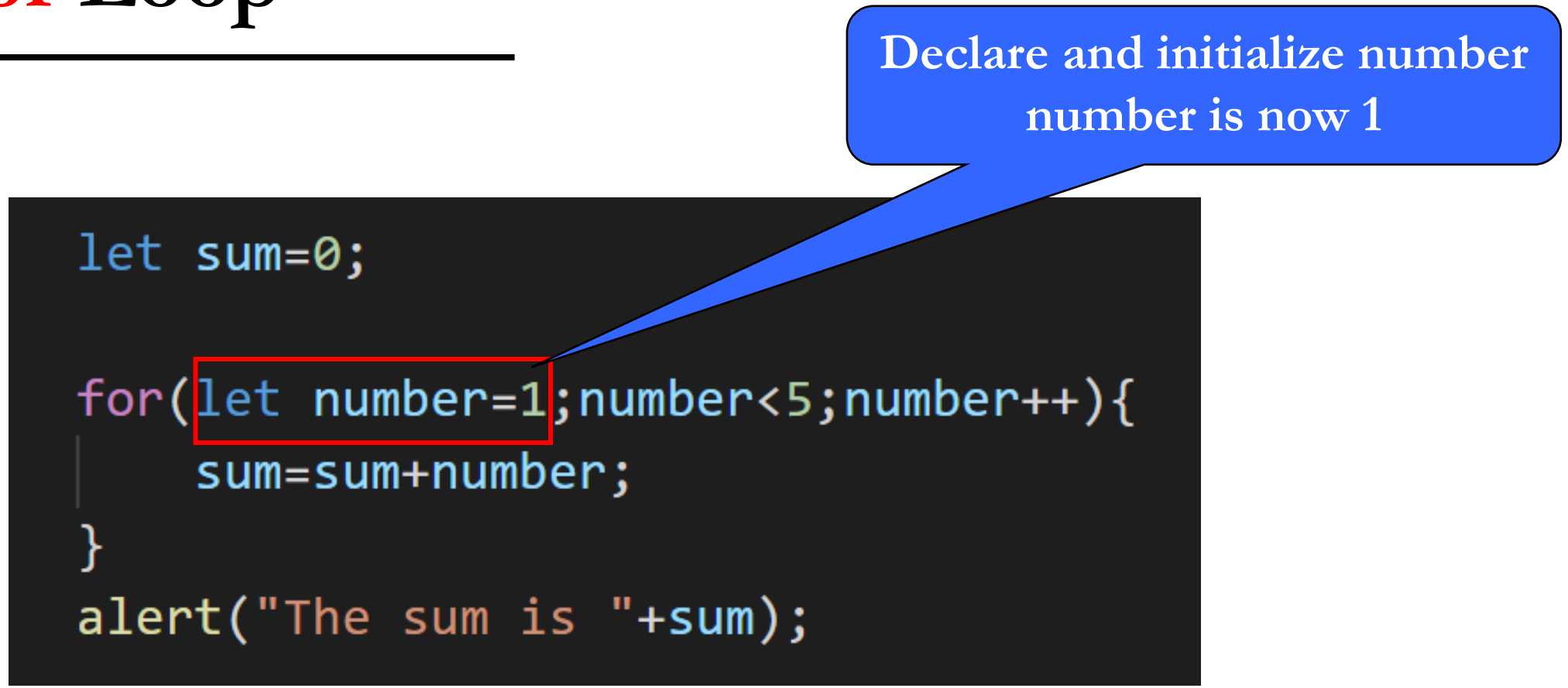
Trace for Loop

Declare and initialize sum
sum is now 0

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop



Trace for Loop

(number < 5) is true since number is 1

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Add the current value of sum (0) to current value of number (1), and assign it to sum. sum now is 1

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Execute adjustment statement number now is 2

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

(number < 5) is true since number is 2

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Add the current value of sum (1) to current value of number (2), and assign it to sum. sum now is 3

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Execute adjustment statement number now is 3

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

(number < 5) is true since number is 3

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

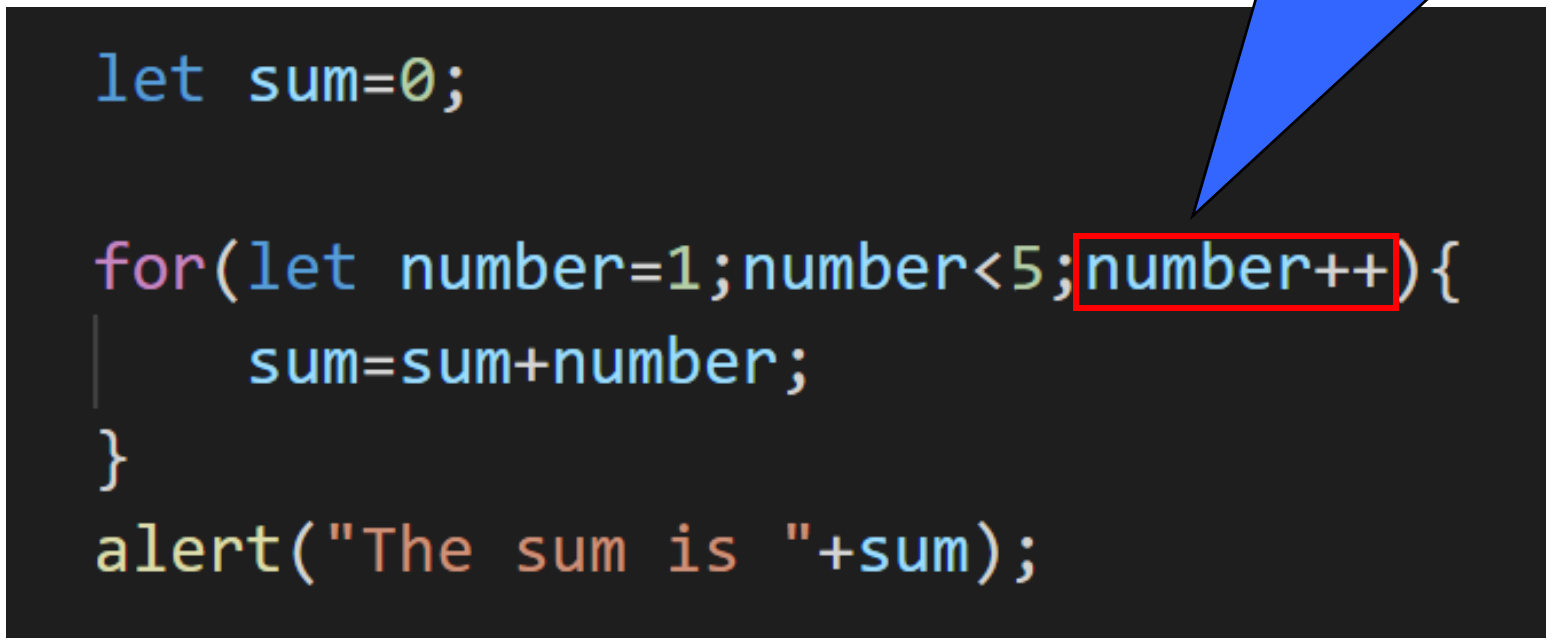
Trace for Loop

Add the current value of sum (3) to current value of number (3), and assign it to sum. sum now is 6

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop



Trace for Loop

(number < 5) is true since number is 4

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Add the current value of sum (6) to current value of number (4), and assign it to sum. sum now is 10

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

Execute adjustment statement number now is 5

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

(number < 5) is false since the number is now 5

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Trace for Loop

```
let sum=0;

for(let number=1;number<5;number++){
    sum=sum+number;
}
alert("The sum is "+sum);
```

Exit the Loop and display The sum is 10.

while Loop

Digital Product Management

FOX
MIS

Introducing while Loop

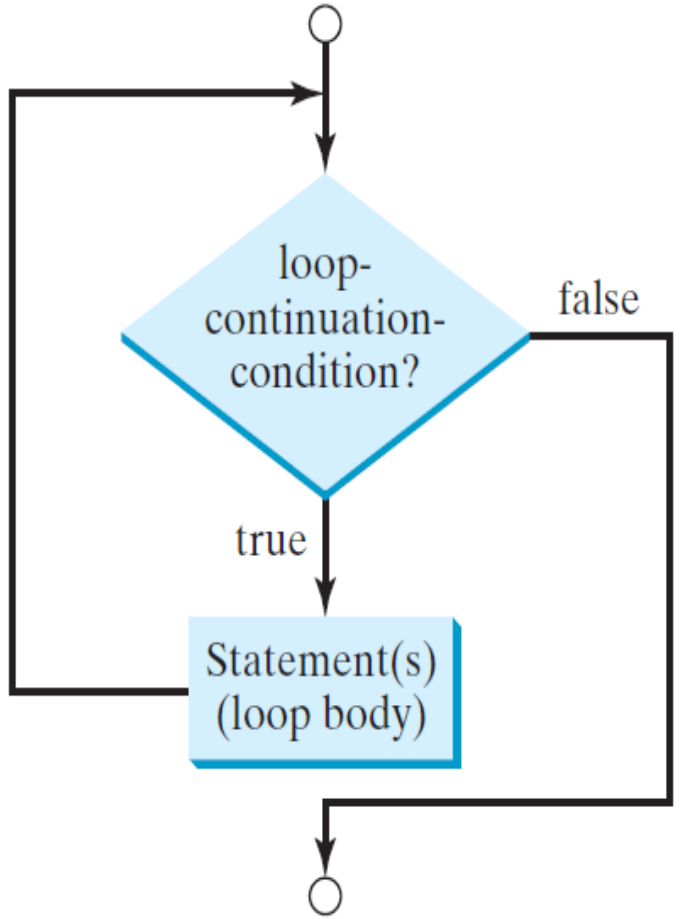
```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

The while loop will run until its looping condition evaluates to being false.

while Loop FlowChart

```
while (loop-continuation-condition)
{
  // loop-body;
  Statement(s);
}
```



Trace while Loop

Declare and initialize count
count is now 0

```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

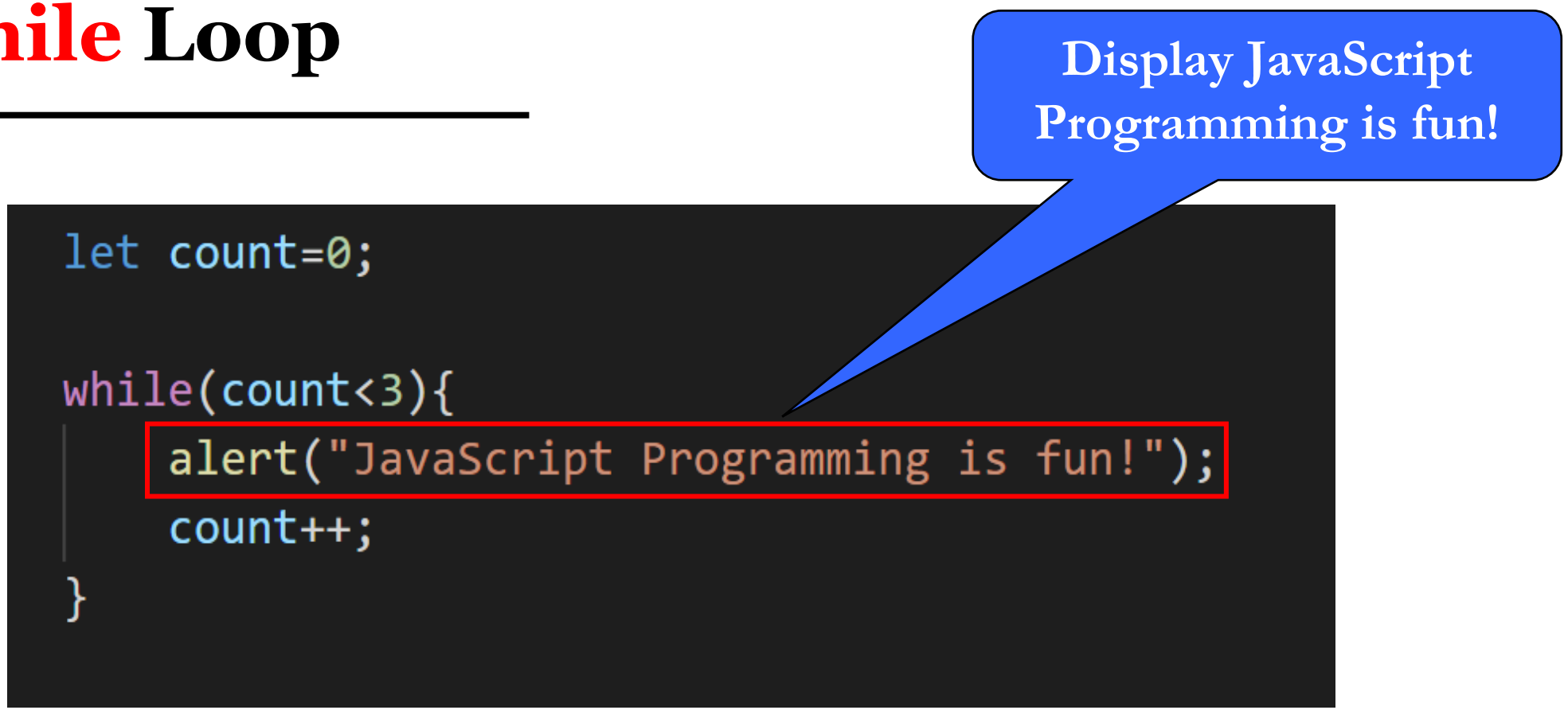
Trace while Loop

(count < 3) is true.
Since count is 0

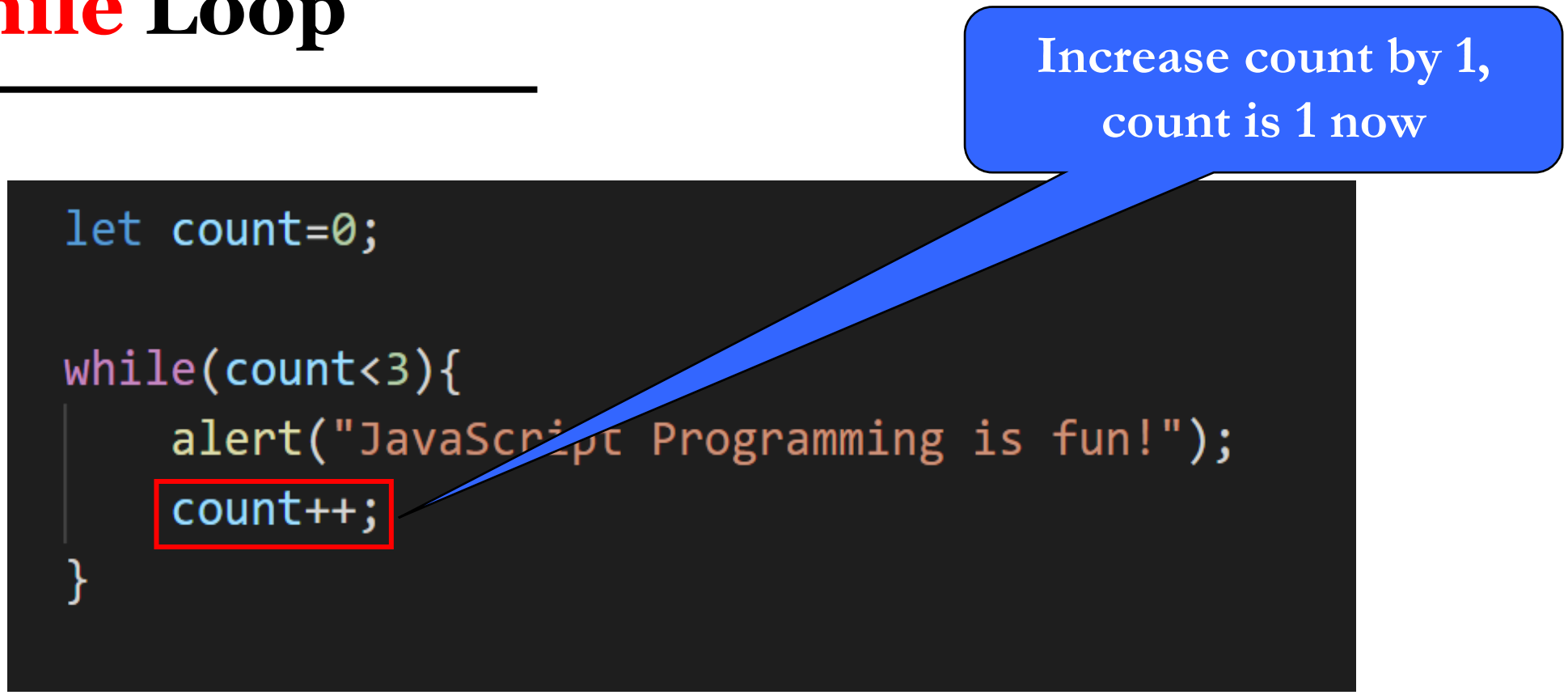
```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

Trace while Loop



Trace while Loop



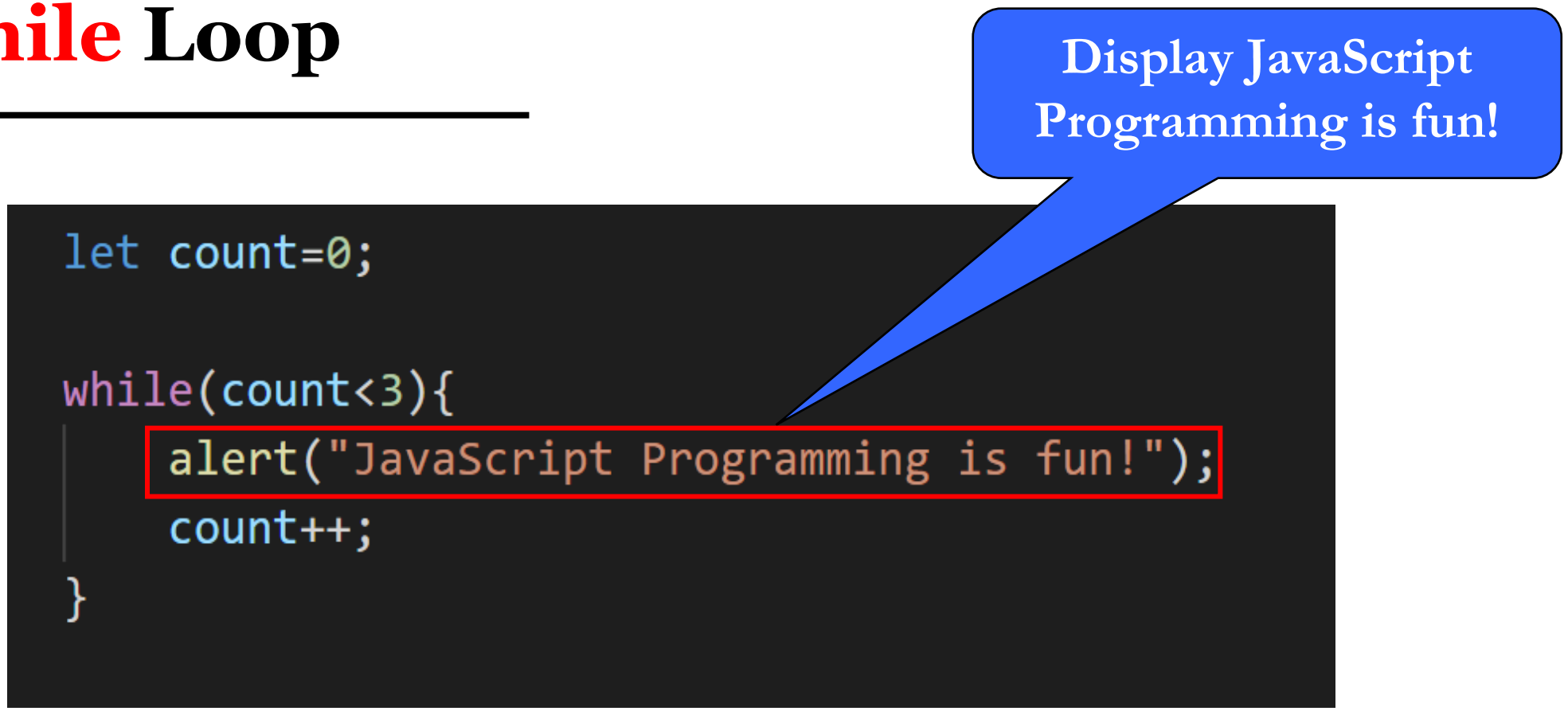
Trace while Loop

(count < 3) is true.
Since count is 1

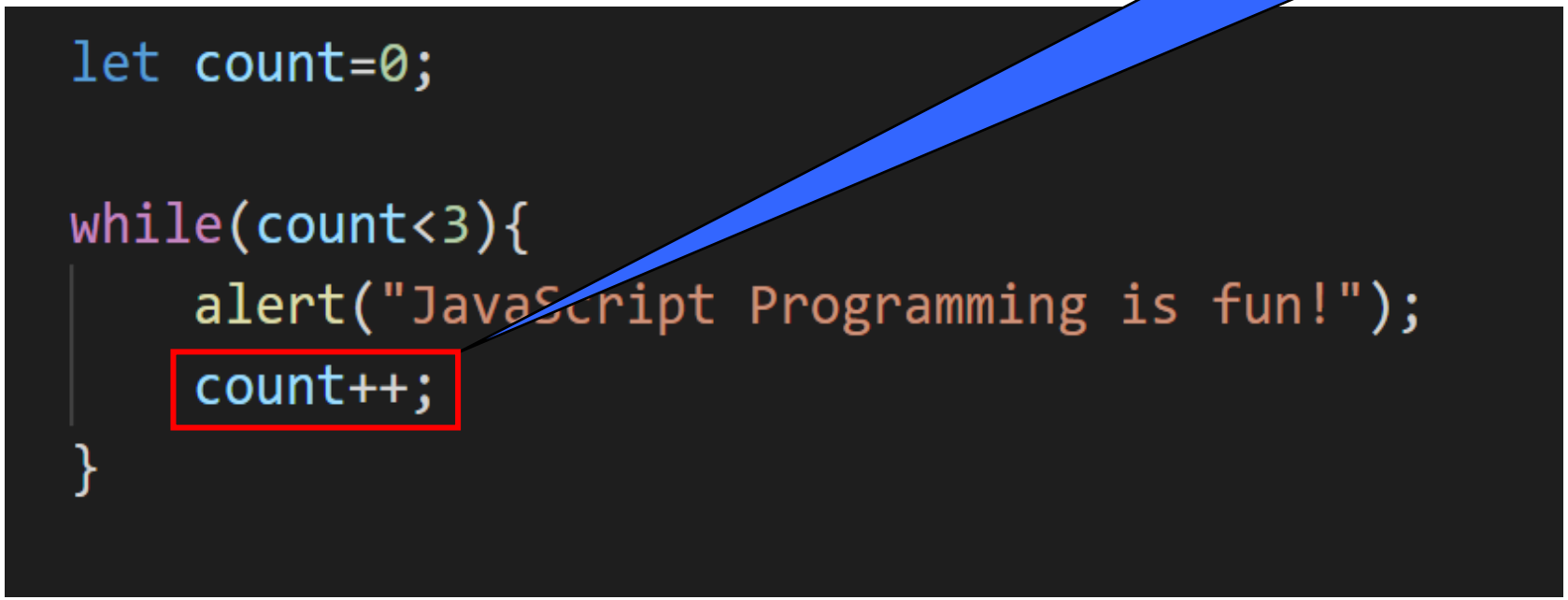
```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

Trace while Loop



Trace while Loop



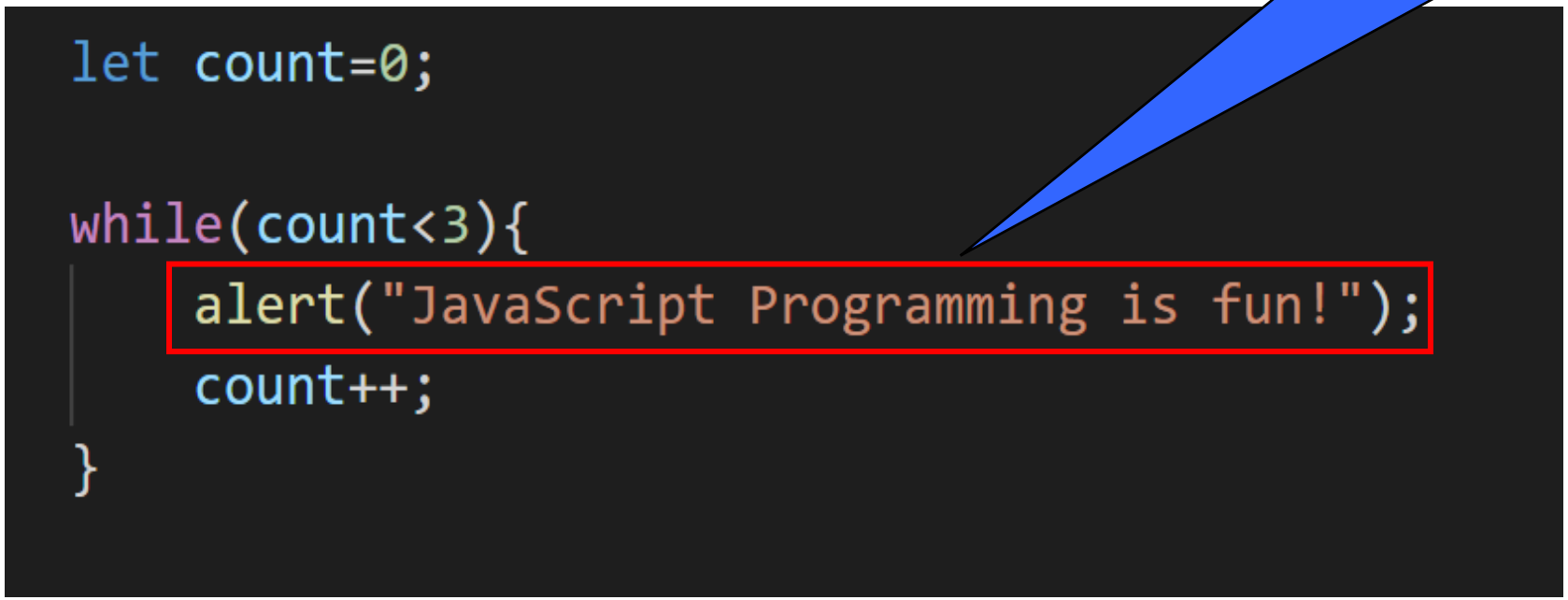
Trace while Loop

(count < 3) is true.
Since count is 2

```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

Trace while Loop



Trace while Loop

Increase count by 1, count is 3 now

```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

Trace while Loop

(count < 3) is false.
Since count is 3

```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

Trace while Loop

```
let count=0;

while(count<3){
    alert("JavaScript Programming is fun!");
    count++;
}
```

The loop exits. Execute the next statement after the loop.

do-while Loop

Digital Product Management

FOX
MIS

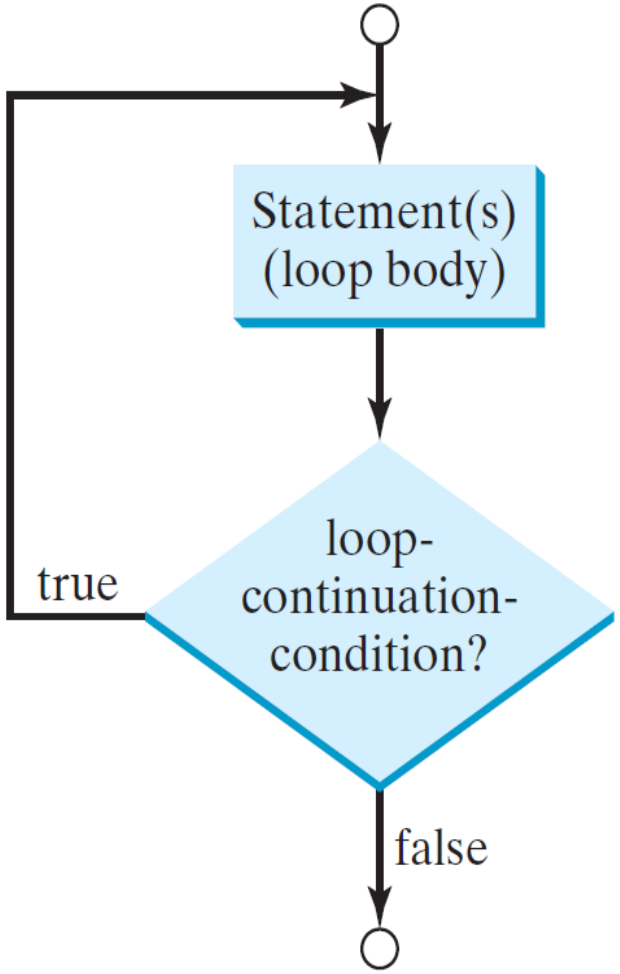
Introducing do-while Loop

```
let count=0;

do{
    alert("JavaScript Programming is fun!");
    count++;
} while(count<3);
```

do-while Loop FlowChart

```
do {
 // Loop body;
  Statement(s);
} while (loop-continuation-condition);
```



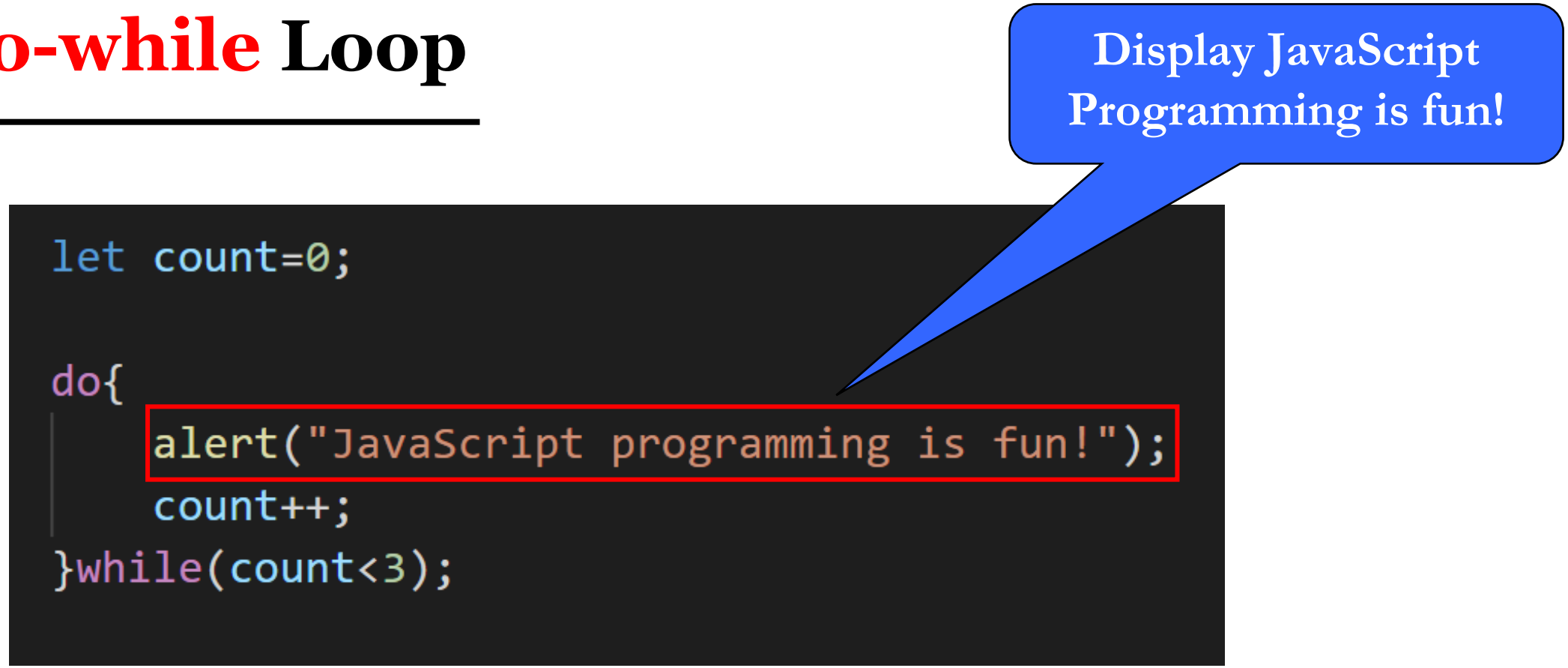
Trace do-while Loop

Declare and initialize count
count is now 0

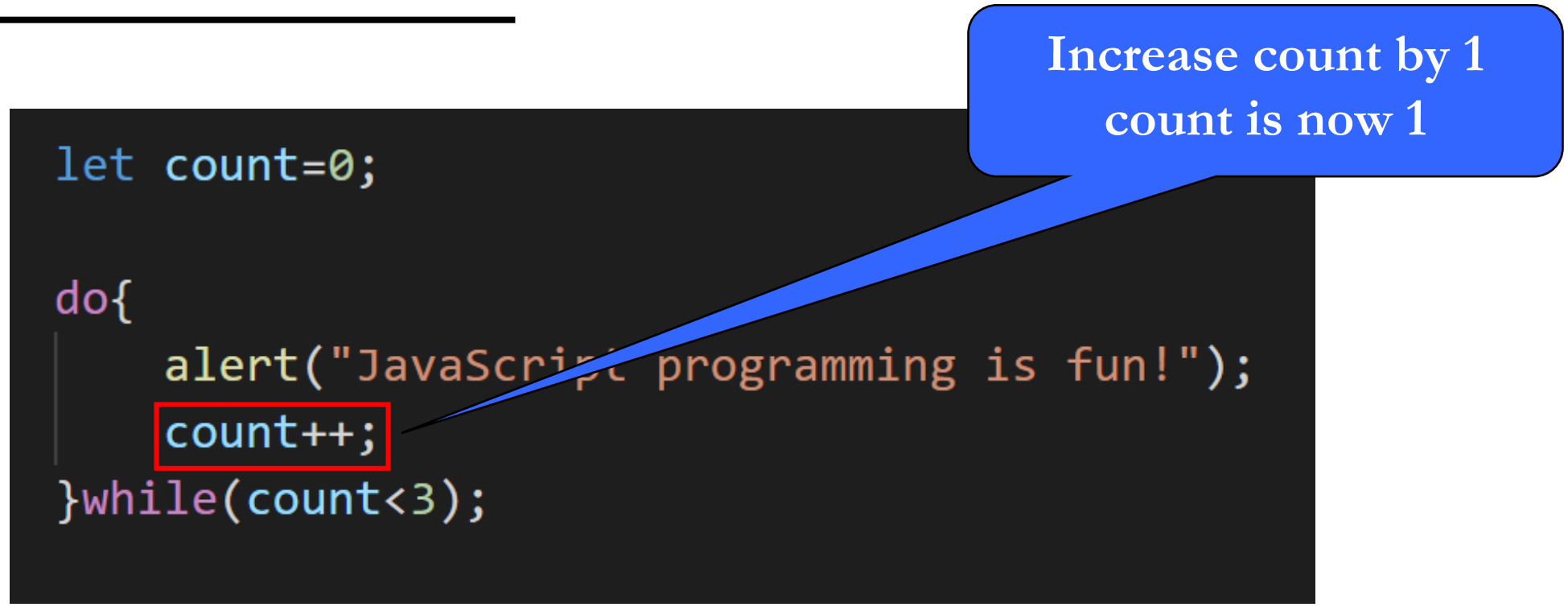
```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

Trace do-while Loop



Trace do-while Loop



Trace do-while Loop

```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

(count < 3) is true.
Since count is 1

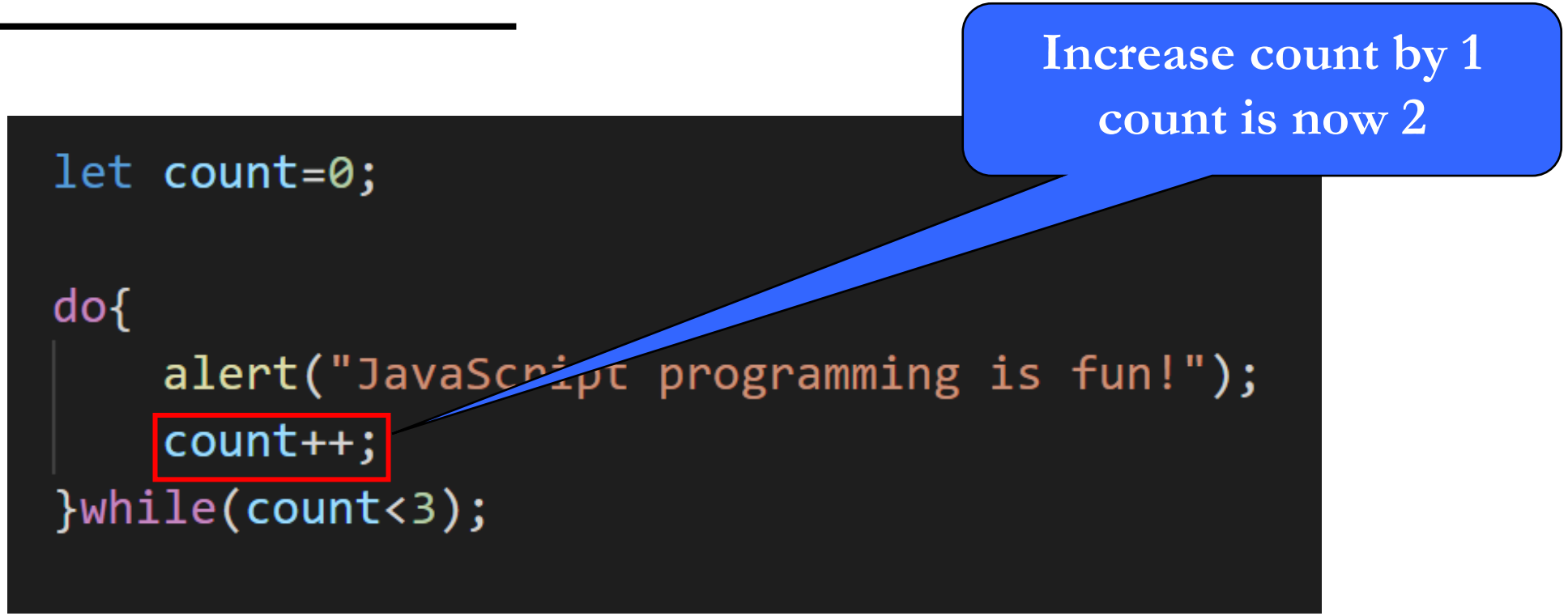
Trace do-while Loop

Display JavaScript Programming is fun!

```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

Trace do-while Loop



Trace do-while Loop

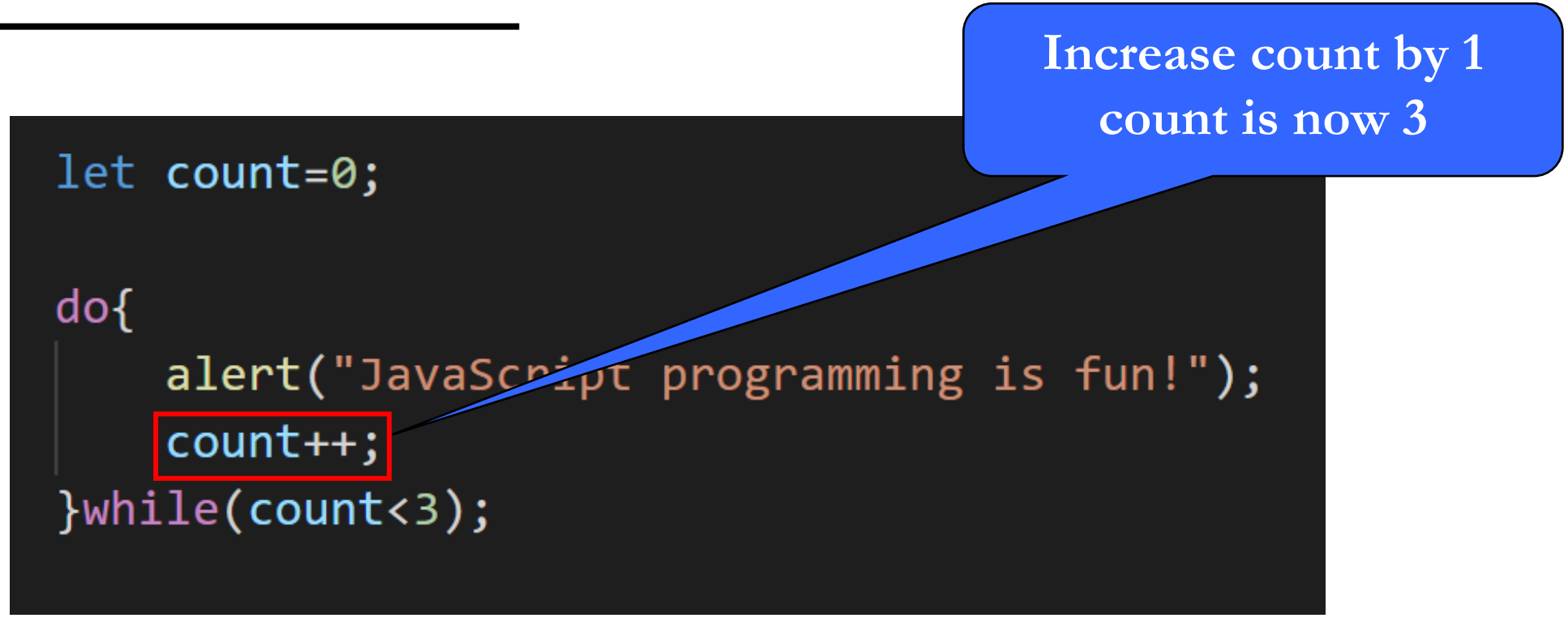
```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

(count < 3) is true.
Since count is 2

Trace do-while Loop

Trace do-while Loop



Trace do-while Loop

```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

(count < 3) is false.
Since count is 3

Trace do-while Loop

```
let count=0;

do{
    alert("JavaScript programming is fun!");
    count++;
}while(count<3);
```

The loop exits. Execute the next statement after the loop.

Time for “Challenges”!

Challenges

- BugCollector
- CaloriesBurned
- BudgetAnalysis
- DistanceTraveled
- C2FTable
- GuessANumber

Homework

- **Write the 1st function for Riley's Ranking Calculator:**
  - function totalAssets()
- **Write the 2nd function for Riley's Ranking Calculator:**
  - function totalDebt()

Diamond Peer Teacher
Quinten Powers

Bug Collector Intro Walkthrough

FOX
MIS

Diamond Peer Teacher
Lauren Quinn

Debugger Video 4 (Developer Tools Breakpoints)

FOX
MIS

Diamond Peer Teacher
Anna Boykis

Guess A Number Walkthrough

FOX
MIS

Diamond Peer Teacher
Lauren Quinn

Sum of Numbers Walkthrough

FOX
MIS

Diamond Peer Teacher
Quinten Powers

Average Rainfall Walkthrough

FOX
MIS

Diamond Peer Teacher Quinten Powers

Pennies For Pay Walkthrough

FOX
MIS

More to Come

Prepare with Readings & Videos before our next class!!!