

script.js

```
1 // 🖱️ Loops are used to repeatedly run a block of code.
2
3 // 🖱️ There are three types of loops in JavaScript: for, while and do-while.
4
5 // 🖱️ For Loop: We use for loop to run a piece of code until the set condition turns false.
6
7 // 🖱️ For example: 🖱️
8
9 console.log("For Loop");
10
11 console.log("Example 1:");
12
13 for (let i = 0; i < 2; i++) {
14
15     let Name = "PW Skills";
16
17     console.log(Name);
18 }
19
20 console.log("Example 2:");
21
22 for (let i = 2; i <= 20; i += 2) {
23
24     console.log(i);
25 }
26
27 // 🖱️ While Loop: This loop keeps on running as long as the condition is true.
28
29 // 🖱️ For example: 🖱️
30
31 console.log("While Loop");
32
33 console.log("Example 1:");
34
35 let i = 1;
36
```

```
37 while (i < 10) {
38
39     console.log(i);
40
41     i++;
42 }
43
44 console.log("Example 2:");
45
46 let j = 10;
47
48 while (j > 0) {
49
50     console.log(j);
51
52     j--;
53 }
54
55 // 🖱️ Do-While Loop: This loop will always run at least once, even if the condition is false.
56
57 // 🖱️ For example: 🖱️
58
59 console.log("Do-While Loop");
60
61 console.log("Example 1:");
62
63 let result = '';
64
65 let k = 0;
66
67 do {
68
69     k = k + 1;
70
71     result = result + k;
72
73 } while (k < 7);
74
```

```
75 console.log(result);
76
77 // 📌 When to use which loop?
78
79 // 📌 Use for loops when you know the number of iterations or the range of values.
80
81 // 📌 Use while loops when you don't know the number of iterations or where the termination condition is based on a dynamic condition.
82
83 // 📌 Use do-while loops when you want to execute a block of code at least once, even if the condition is false.
```