

The provided code will output the number 5 five times, each with a delay of 100 milliseconds. This behaviour might not be what you expect at first glance due to a common misunderstanding with JavaScript closures and the **setTimeout** function.

Let's break down what happens in the code:

1. The **for** loop runs from **i = 0** to **i < 5**.
2. In each iteration of the loop, a **setTimeout** function is called with a callback function that logs the value of **i** to the console after a delay of 100 milliseconds.
3. However, the **for** loop runs quickly, and by the time the first **setTimeout** callback is executed, the loop has already completed, and the final value of **i** is **5**.

Therefore, when the **setTimeout** callbacks are triggered after the delay, they will all log the final value of **i**, which is **5**.