Block 1:

The code initializes a variable b to the value of 1. Then, it declares a function someFunction that takes a single parameter number. Within someFunction, another function otherFunction is declared that returns the value of b. The value of b is then updated to 5 and otherFunction is returned. Next, someFunction is called with an argument of 9 and the result is assigned to firstResult. firstResult is a reference to the otherFunction that was created within someFunction. Finally, firstResult is called with an argument of 2 and the result is assigned to result. Since b was updated to 5 before otherFunction was returned, result will be 5.

Block 2:

The code initializes a variable a to the value of 1. Then, a function b2 is declared which assigns the value of 10 to a and returns nothing. However, b2 also declares a function a within its scope that does nothing. When b2 is called, it assigns the value of 10 to the local variable a, not the global variable a declared outside of b2. Finally, console.log(a) is called, which outputs the value of the global variable a, which is still 1. Therefore, the output will be 1.

Block 3:

The code initializes a variable i but does not give it a value. Then, a for loop is declared that will iterate three times. Within the loop, a constant log is declared and assigned a function that logs the value of i to the console. However, the log function is not called within the loop. Instead, a setTimeout function is called with log and a delay of 100 milliseconds as arguments. Since setTimeout is asynchronous, it will not execute the log function until after the loop has finished iterating. By the time the log function is executed, i will have a value of 3 because it was incremented three times during the loop. Therefore, the output will be three log statements of the value 3.