**Node.js**

Task1

**Event loop**

The event loop is a way off operation, that JavaScript as a programming language follow to execute code. JavaScript to begin with is a single threaded language, which means that it only executes one single line of code at a time, with the priority to the line written first.

Although JavaScript executes code in a specific order, this order may change if the function was an asynchronous function, this is where the event loop method works. For a function set with time delay (SetTimeout) is recognized as an asynchronous function, hence is not ready to be executed, and as a result the priority is given to the functions ready to be executed. These asynchronous functions are sent to the browser’s API to be queued until it is ready to be executed, however it will only be executed after all synchronous functions are executed.

1. console.log(‘*Hello’*);
2. setTimeout(()=>{

console.log(‘*Second statement*’),2000

};

1. setTimeout(()=>{

console.log (‘*Third statement*’),0

};

1. console.log(‘*Goodbye’*);

**Console**

1. Hello
2. Goodbye
3. Third statement
4. Second statement

As shown in this example, “Hello” and “Goodbye” were executed first then the “Third statement” after finishing the time set, and finally the “Second statement:”. This shows that the time given to the functions affected its priority to be executed among the asynchronous functions only, making the first function to be ready for execution gets the highest priority.

**References**

The event loop - JavaScript: MDN (2022) JavaScript | MDN. Mozilla Foundation. Available at: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/EventLoop>.