**Q1:**

**Write the necessary Node script to make this code work for all arrays: [1,2,3,4,5,6,7,8].even(); // [2,4,6,8] [1,2,3,4,5,6,7,8].odd(); // [1,3,5,7] Test your code in Node.JS CLI**

 Array.prototype.even = function () {

                return this.filter((x) => x % 2 === 0);

            };

            Array.prototype.odd = function () {

                return this.filter((x) => x % 2 !== 0);

            };

            console.log(`Even output: ${[1, 2, 3, 4, 5, 6, 7, 8].even()}`);

            console.log(`Odd output: ${[1, 2, 3, 4, 5, 6, 7, 8].odd()}`);

**Q2:**

1. **Explain why do we want sometimes to use setImmediate instead of using setTimeout?**

setImmediate is called during the check handler phase of the event loop, while setTimeout is called after the provided delayed time and also block I/O. We use setImmediate, because it allows use to schedule a callback within the I/O cycle thus guarantees that our callback will run first regardless of any other timer and their delay time. Also, setImmediate won't block the event loop, as event recursive call is executed only on the next event loop iteration.

1. **Explain the difference between process.nextTick and setImmediate?**

Event loop has multiple phases. Process.nextTick is called at the start of every event loop and in between each event phases. While, setImmediate is called during the check handler phase of the event loop. In fact process.nextTick has got high priority than setImmediate.

1. **Does Node.js has window object?**

Even though node.js is build on top of javascript (v8 engine), it does not have the window object. Window object is a global object in browser. Node, on the other hand runs in the server and hence does not have Window as its global objects