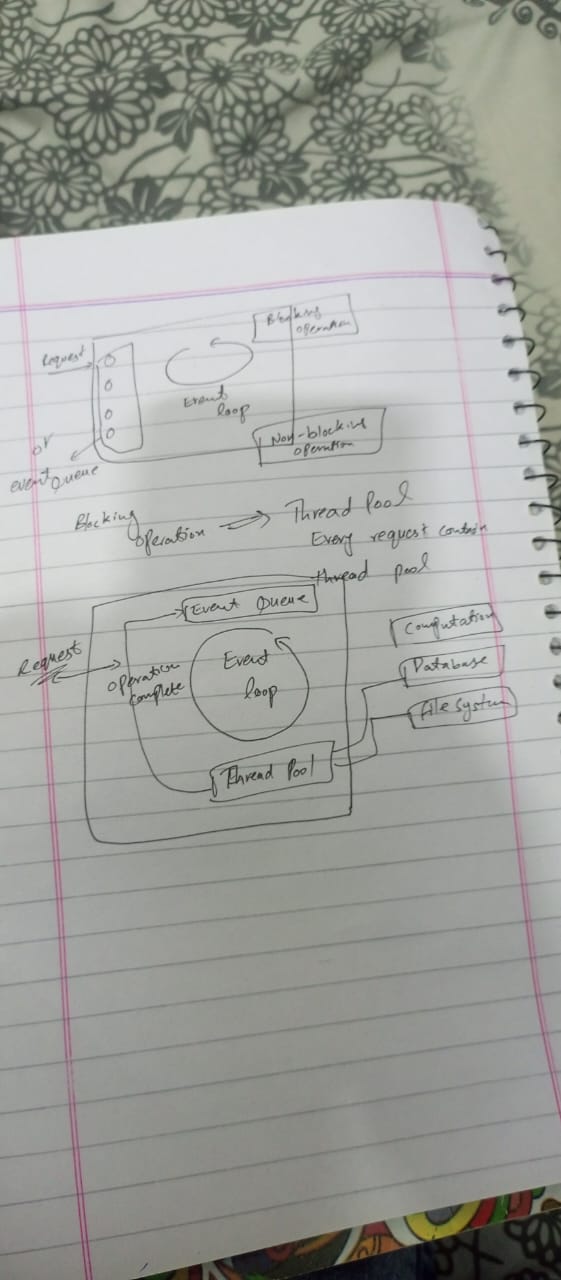
Task 1:



Event loop allows NodeJs to perform non-blocking I/O operations like javascript acts as a single thread. When the event loop starts it initializes the event loop and processes the provided input script. Node.js can handle multiple operations simultaneously by offloading tasks to the system kernel whenever possible.  
  
console.log('Start of script');  
// Timers  
setTimeout(() => {  
console.log('Timer callback executed');  
}, 1000);   
  
// Pending Callbacks  
process.nextTick(() => {  
console.log('process.nextTick callback executed');  
});  
  
// Poll (simulating I/O)  
const fs = require('fs');  
fs.readFile('example.txt', 'utf8', (err, data) => {  
if (err) {  
console.error('Error reading file:', err.message);  
} else {  
console.log('File content:', data);  
}  
});  
  
//setImmediate  
setImmediate(() => {  
console.log('setImmediate callback executed');  
});  
  
console.log('End of script');