

### 1. Discuss on what is an Event Loop.

The event loop is an important component or mechanism in Node.js that lets it efficiently run programs or operations simultaneously and asynchronously. It enables Node.js to handle a high number of concurrent connections and it constantly checks if there are tasks that need to be executed asynchronously and processes them properly. The event loop has many phases. It makes use of callbacks, timers, I/O operations, and many more others that processes tasks sequentially. The cycle will repeat indefinitely while Node.js is running

### 2. Explain the difference between Core Modules, Local Modules and Third-party modules used in Node.js.

- Core Modules – These are built-in modules that are provided by Node.js once it is installed. They can be loaded in the program by using the *required* function. Important core modules include http, url, querystring, path, fs, and util which has different functions/purpose.
- Local Modules – These modules are created by the users themselves and are locally stored in the project. It also needs the require function and its relative path must be specified in order to be used.
- Third-party Modules – These are the modules created by other developers that need to be installed globally which are hosted and installed using the NPM or Node Package Manager. Some popular third-party modules include express, angular, and react.

### 3. What is the advantage of using exports or module.exports in a Node.js application?

The exports or module.exports is a special object which exposes whatever was assigned to it as a module. These are a useful part of Node.js as by separating different parts of the code, it can make it easier to understand and maintain. By exporting specific functions, objects, or variables, developers can reuse code across different modules or projects which can greatly reduce redundancy and improving development efficiency. By using the require function and specifying the path, you will be able to access or connect separate files that are encapsulated, which also enables modularity in the Node.js application.