**Node.js Module Types**   
  
In Node.js modules can be categorized in 3 types https://nodejs.org/api/modules.html#modules\_core\_modules,

1. Core Modules
2. Local Modules
3. Third Party Modules

**Node.js Core Modules**  
Node.js has several modules compiled into its binary distribution, and load automatically when the Node.js process starts, these are called the core modules. These core modules of node.js  are located within Node.js's source and are locate d inside **"lib"** folder. Some of the core modules are listed below.

* *http -*This module is used to create http server.
* *fs -*This module is used to perform file operations like reading, writing, appending and deleting files etc.
* *Crypto -*This module provides cryptographic functionalities like encryption, decryption, sign, verification, digesting etc.
* *Querystring -*This method includes methods to deal with querystring like unescapeBuffer, unescape, escape,  encode, stringify, decode and parse.
* *url -*This module includes methods for url resolutions,resolve, parsing, format etc.
* *path -*This module is used to deal with file paths when working with file system.

**How to load a module?**  
  
To load a module in your node application you can just use "require()" function. whose syntax is given below.

1. **var** module=require('module\_name');

There are several ways to reference modules, this depends on what type of module we are going to load.

**Loading** **core module**Core modules can be loaded as follows.

1. **var** http=require('http');

As I have already told you that code modules are loaded in "lib" directory, so in the above example http module will be loaded from lib folder.

**Local Modules**  
Local modules are user defined modules which are mainly used for specific projects and locally available in separate files or folders within project folders. These type of modules contain application specific functionality.  
 **Note**  
We can package locally created local modules and distribute them via NPM (Node Package Manager), which can be used by others and the node community.

**Third party module**  
  
The third party module can be downloaded by NPM (Node Package Manager). These type of modules are developed by others and we can use that in our project. Some of the best third party module examples are listed as follows: express, gulp, lodash, async, socket.io, mongoose, underscore, pm2, bower, q, debug, react, mocha etc.  
  
Third party modules can be install inside the project folder or globally.

**Loading third party modules**Third party Node.js module can be downloaded using NPM (node package manager) which you can download locally or globally. To download globally we use the following command.

1. npm install -g <module\_name>

here  we use -g to install package globally. If you want to install locally then use the following command.

1. npm install --save <module\_name>

Above command will download node package inside node\_modules folder and then you can directly use require function to load node module.

1. **var** module= require('module\_name');

**How thirdparty modules are installed.**

Npm install nodemon -g

Npm install chalk --save

Npm install express -g

Npm install body-parser –g

**Request Parameters**

**Retrieving HTTP GET Request Parameters**

There is two way

1. Named parameter: req.query.parameter
2. Positional parameter: req.params. parameter

**Retrieving HTTP POST Request Parameters**

req.body.nm

**Points of difference between Get and Post methods**  
  
**Get method**

* It is used when the url is sent to the server.
* The Get method is used to retrieve web pages from the server.
* It is the first method. This is the default method for many browsers.
* Get is also used to send user information from an online form on a web page to the server. Data is sent as a part of the URL in 'name-value' pairs.
* Form data are restricted to ASCII codes.
* There is a limitation on how much form data can be sent because URL lengths are limited.
* With the Get method, the browser appends the data onto the URL.
* All form data filled in is visible in the URL. Moreover, it is also stored in the user's web browsing history/logs for the browser.
* Get method is less secure.

**Post method**

* First request can't be Post.
* You send data with Post request i.e. form information. Data will appear within message body.
* It is sent in hidden (encrypted form)
* Post is safer than Get
* IT has large value sending option. Binary data, images and other files can all be submitted through METHOD=POST
* With the Post method, the data is sent as standard input.
* No default and should be Explicitly specified.

**Middleware in Express**

* static contents Middleware-  
   app.use(express.static(\_\_dirname + "/public"));
* body-parser middleware for Http POST requests  
   var bodyparser = require('body-parser');