Unit 10.1 Highlights

server - Physical hardware or software that takes requests from users and gives something back or completes a process

client - apps (i.e. browsers, desktop apps, etc.) which make requests to servers

request - Asking for something or some process to happen (i.e. HTTP GET, POST, PUT, DELETE)

response - The result of the request or what is returned after a request

client-server relationship - Describes the relation between the client and how it makes a service request to the server, and how the server can accept these requests, process them, and return the requested information to the client.

(Source: https://simple.wikipedia.org/wiki/Client-server)

(Source: https://www.w3schools.com/nodejs/nodejs_intro.asp)

Nodejs - Server-side JavaScript which uses asynchronous programming

Ex. 1 (invoking node from command line)

```
$ node <filename>
$ node something.js
$ node /Desktop/somefolder/something.js
```

(Source: https://nodejs.org/en/docs/guides/blocking-vs-non-blocking/)

synchronous (blocking) threading - Execution of additional JavaScript in the Node.js process must wait until a non-JavaScript operation completes

asynchronous (non-blocking) - Execution of additional JavaScript in the Node.js process doesn't wait until a non-JavaScript operation completes

---

(Source: https://nodejs.org/docs/latest/api/process.html)

process.argv - Returns an array containing the command line arguments passed when the Node.js process was launched

Ex. 4

```
$ node yournodefile.js one blue 16 four

//somewhere within yournodefile.js...

console.log(process.argv[0]); //<path to node>/bin/node
console.log(process.argv[1]); //<path to file>/yournode.js
console.log(process.argv[2]); // one
console.log(process.argv[3]); // blue
console.log(process.argv[4]); // 16
console.log(process.argv[5]); // four
```

(Source: https://www.w3schools.com/jsref/jsref_forin.asp)

for/in Loop - iterates over all enumerable properties of an object (i.e., the object equivalent of a 'for loop' iterating through all items with an array)

Ex. 5

```
var string1 = "";
var object1 = {a: 1, b: 2, c: 3};

for (var property1 in object1) {
  string1 = string1 + object1[property1];
}

console.log(string1);
// expected output: "123"
```

(Source: https://www.w3schools.com/nodejs/nodejs_modules.asp)

require() - Includes a module within a javascript file with the name of the module

Ex. 5

```
//file1.js

var services = {
      power: “electricity”,
      water: “H2O”,
      internet: “cable”,
      trash: “waste disposal”
}
module.exports = services;



//file2.js

var services = require(‘./file1.js’);
console.log(services.water); // “H2O”
```