**HTTP**

The http global core object is used to transfer data over the Hyper Text Transfer Protocol. Using this object, Node.js can handle incoming HTTP requests, and send out responses. This can be used as the backbone to a Restful API; handling different incoming requests and giving out desired responses.

One can do this using createServer() and listen(). CreateServer() is used to tell the server how to handle incoming requests and what kind of responses to return. Listen() specifies the port in which we will listen for requests.

**FileSystem**

The filesystem core object is used to manipulate/read from files on the host computer.

Traditionally, JavaScript is ran from a web browser and thus would have core objects that allowed it to interact with the webpage such as the DOM (Document Object Model). Since JavaScript ran from node.js isn’t tied to a webpage, it does not have direct access to the Document Object Model. In its place, we can interact with the machine running a Node.js file itself; one such way is the file system.

With the file system object, we can read files, update existing files, create new files, and delete old files using JavaScript code. When combined with the above, it can be used to serve html files.

**OS**

In addition to reading files from the host computer, we can also get system information from the host computer. The OS core object allows us to retrieve information such as operating system and hardware information. In a real application, we can use this information to scale the quality/performance of the program based on the computer running it.