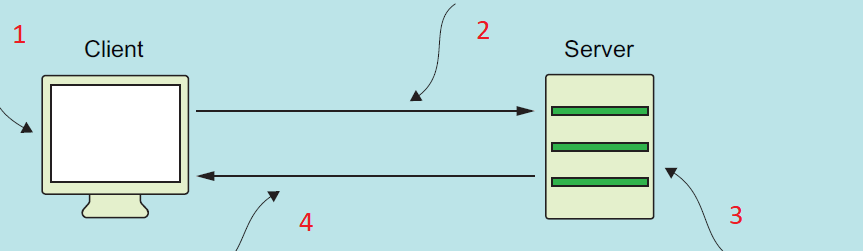
# Requerimientos

* [Postman](https://www.postman.com/downloads/)
* Visual Studio Code
* NodeJs

Introducción

[Node.js](https://nodejs.org/en/about/) es un programa asíncrono en tiempo de ejecución de JavaScript basado en eventos.

* NodeJS utiliza el motor de JavaScript
* NodeJS utiliza Chrome V8 Google
* Cuando se inicia una aplicación JavaScript, todo el código de esa aplicación se carga en memoria.
* JavaScript es asíncrono por naturaleza.



1. El cliente representa al navegador Web sobre el cual el cliente puede ver tu aplicación.
2. El cliente envía una petición al servidor, sabe que datos le gustaría enviar o recibir.
3. El servidor es donde tu aplicación se ejecuta y maneja algún dato enviado por el usuario.
4. El servidor responde al cliente por cargar una página o enviar datos.

Ejemplo HelloWorld

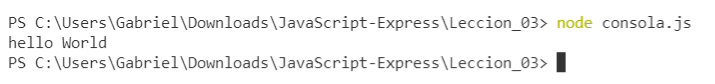
* Abre tu editor de texto favorito
* Ingresar el siguiente código:

console.log("hello World");

* Guardar el archivo como **helloWorld.js**
* Abrir una terminar Windows, navegar hasta donde se encuentra el archivo y ejecutar el siguiente comando:

**node helloWorld.js**

# Salida



Construyendo un Simple Servidor Web con NodeJs

* Abrir una terminal de Windows
* Crear una carpeta llamada **simple\_servidor**

**mkdir simple\_servidor**

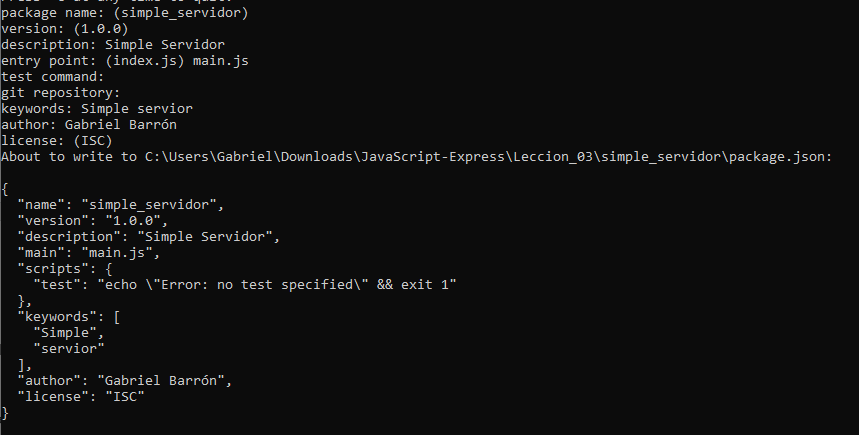
* Cambiarse a la carpeta recién creada mediante el siguiente comando

**cd simple\_servidor**

* Inicializar el proyecto

**npm init**

Cerciorarse que el punto de entrada sea main.js; al final confirmar los cambios



* Ingresar el comando para abrir Visual Studio Code

**code .**

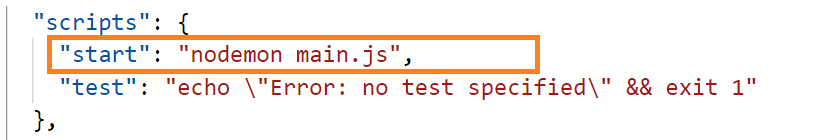
* Observa el archivo package.json que indica los datos de inicialización del proyecto, dependencias del proyecto.
* Crear dentro de la carpeta el archivo **main.js**, que es el punto de entrada de nuestra aplicación
* Instalar siguiente módulo mediante línea de comandos

npm i http-status-codes -S

* Instalar modulo nodemon

**npm i nodemon --save**

* Agregar la instrucción en archivo package.json



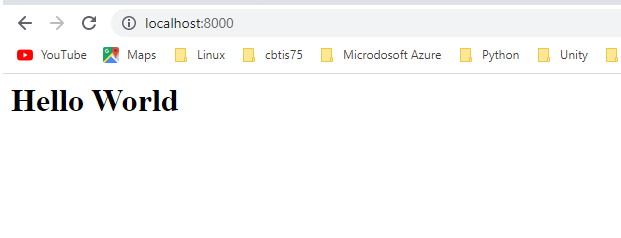
* Modificar el archivo **main.js** e ingresar el siguiente código

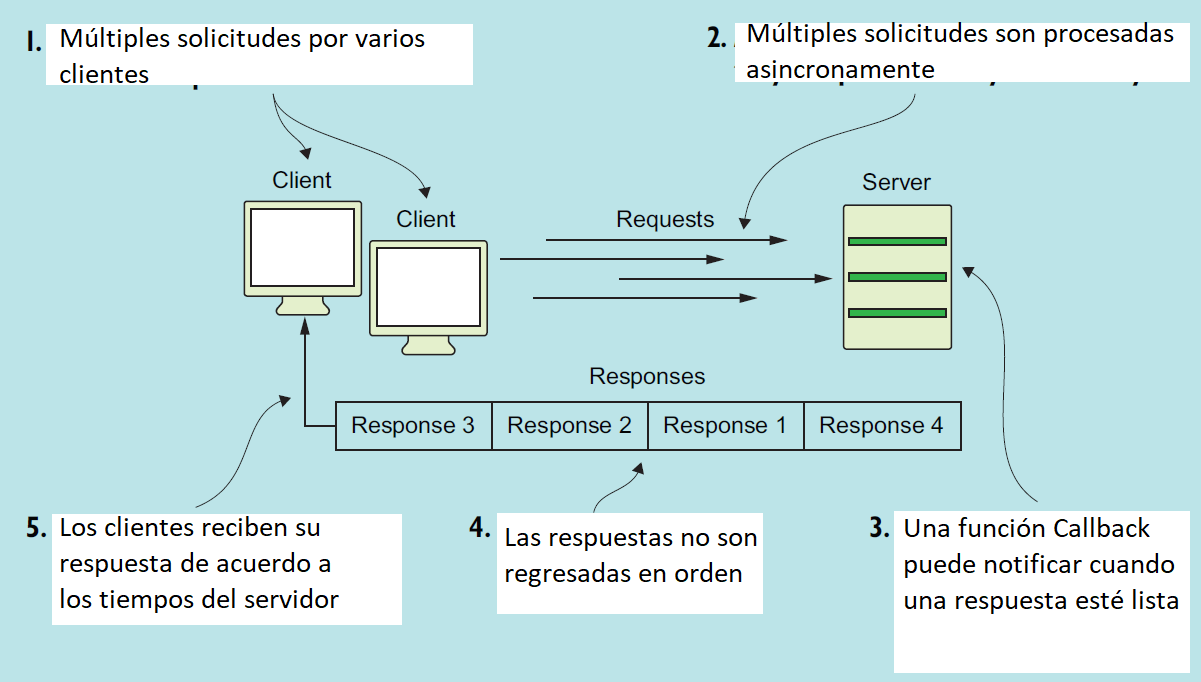
|  |
| --- |
| const http = require('http');  const puerto = 8000;  const app = http.createServer(function(request, response) {      console.log('Atendiendo Peticion');      response.write('<h1>Hello World </h1>');      response.end();  });  app.listen(puerto);  console.log("Iniciando servidor sobre puerto " + puerto); |

* Iniciar la aplicación Web con el siguiente comando

**nodemon main.js**

* Abrir una nueva pestaña en el navegador e ingresar la siguiente URL <http://localhost:8000/>





# Desafío

* Crear una nueva aplicación web.
* Llamar al archivo de entrada **index.js**
* Cambiar el número de puerto de la aplicación Web
* Instalar módulo **nodemon** y agregar el script Start en package.json.
* Usar la consola para indicar cuando se inicie la aplicación y cuando se recibe la petición
* Ejecutar la aplicación y observar en la consola de Windows se observe que se ejecuto.

Creando un segundo Servidor

* Crear un nuevo proyecto llamado **segundo\_servidor** y cambiarse a la carpeta recién creada.
* Inicializa el proyecto con el comando **npm init**
* Instala la dependencia nodemon **npm i nodemon –save**
* Agregar el script nodemon para su ejecución.
* Agregar el script main.js y observa el ejemplo de simple\_servidor y replica la instrucción

|  |
| --- |
| const http= require('http');  const app = http.createServer();  app.on("request", (req, res) =>      console.log("Recibiendo peticiones");      res.writeHead(200, {          'Content-Type':'text/html'      });      let mensaje = "<h1>Welcome back</h1>";      res.end(mensaje);  });  app.listen(3000);  console.log("Se ha iniciado el servidor"); |

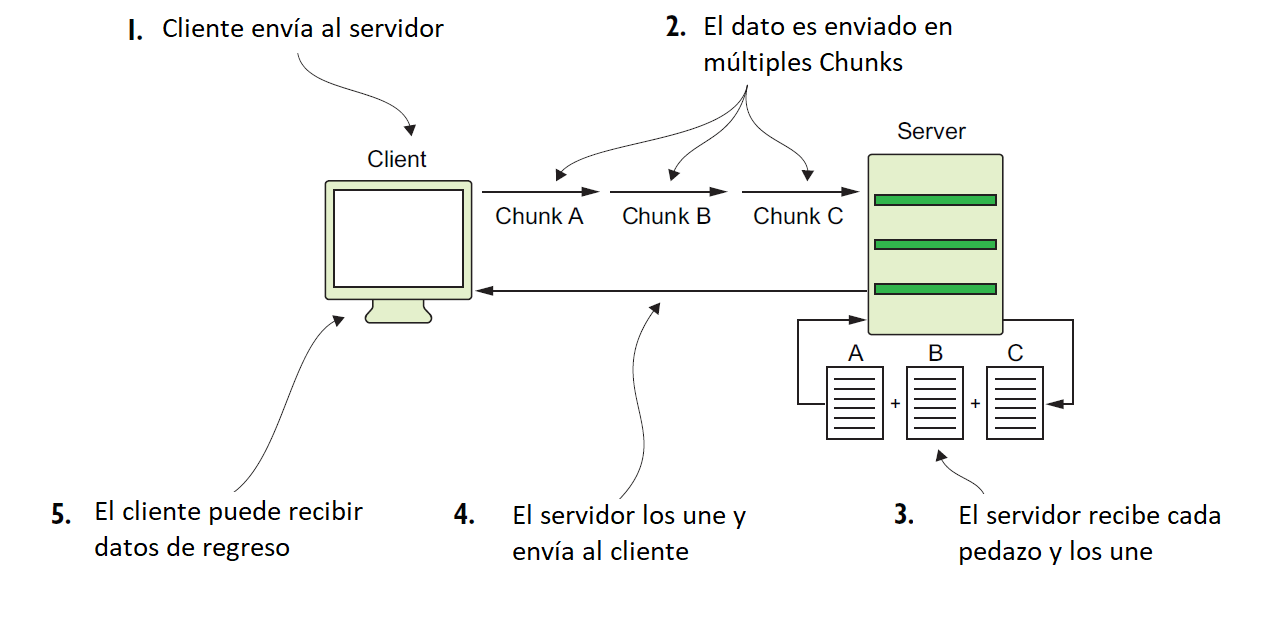
* Agregar información de la petición en salida mediante la función console

|  |
| --- |
| console.log("Método peticion " + req.method);      console.log("URL " + req.url);      console.log("Headers " + req.headers); |

* Para desplegar un arreglo o información JSON utiliza la función JSON.

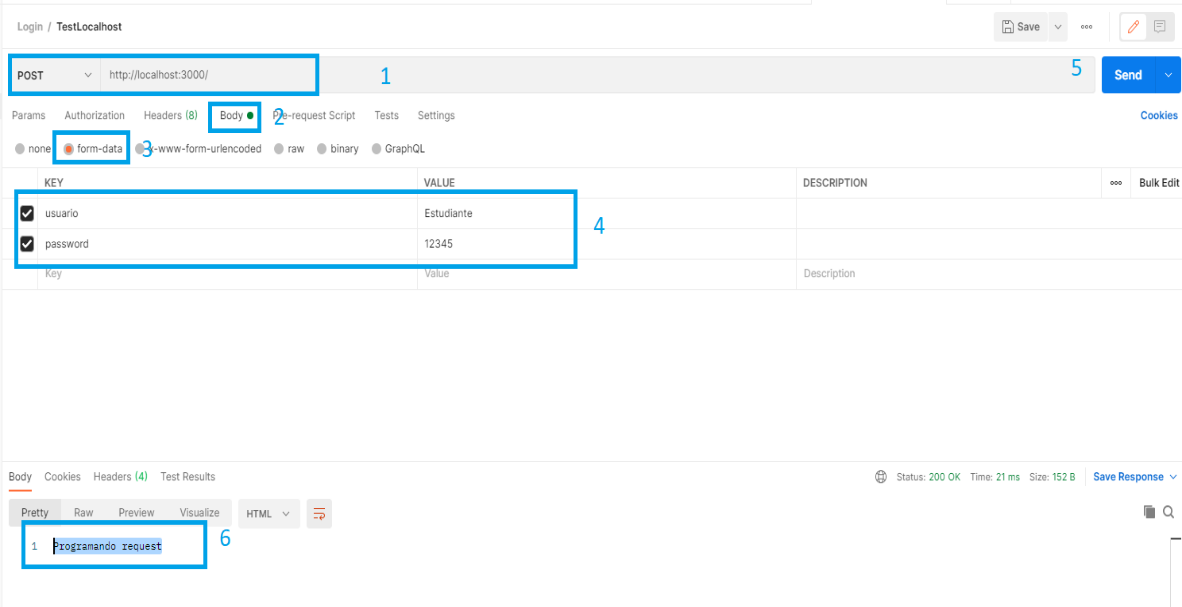
|  |
| --- |
| console.log("Headers " + JSON.stringify(req.headers, null, 2)); |

Vamos a entender como se envía la información



|  |
| --- |
| const http= require('http');  const app = http.createServer();  app.on("request", (req, res)=>{      console.log("Recibiendo peticiones");      var arrDatos = [];      req.on("data", (bodyData) => {          arrDatos.push(bodyData);      });      req.on("end", () => {          arrDatos = Buffer.concat(arrDatos).toString();          console.log("Contenido de los datos " + arrDatos);      });      res.writeHead(200, {'Content-Type': 'text/html'});      res.end('Programando request');      console.log("Método peticion " + req.method);      console.log("URL " + req.url);      console.log("Headers " + JSON.stringify(req.headers, null, 2));      let mensaje = "<h1>Welcome back</h1>";      res.end(mensaje);  });  app.listen(3000);  console.log("Se ha iniciado el servidor"); |

* Hacer la petición con Postman



* Salida en la consola



# Desafío

* Envía parámetros de información como: nombre, apellidos, carrera
* Despliega la información enviada mediante la consola, almacena la información enviada mediante un arreglo.
* Despliega la información enviada mediante la página HTML