Índice.

1. Eventos en Node.

Muchos objetos en Node emiten eventos: un net.Server emite un evento cada vez que se establece una conexión, un fs.readStream emite un evento cuando se abre un archivo. Todos los objetos que emiten eventos son instancias de events.EventEmitter. Puede usar este módulo haciendo require ("events");

Puede adjuntar funciones a los objetos, para que se ejecuten cuando se genere un evento. Estas funciones se denominan oyentes.

2. Módulo de eventos

Los eventos están en un módulo separado que tenemos que requerir en nuestros programas creados con Node JS. Lo hacemos con la instrucción "require":

```
const events = require ('events');
```

Dentro de este o módulo dispones de una serie de utilidades para trabajar con eventos. Primero veamos el emisor de eventos, que se encuentra en la propiedad EventEmitter:

```
let EmisorEventos = events.EventEmitter
```

3. Definir un evento.

En "Node" existe un bucle de eventos, de manera que cuando se declara un evento, el sistema escucha a medida que ocurre, para luego ejecutar una función. Esta función se conoce como "callback" o como "event handler" y contiene el código que queremos que se ejecute cuando se produzca el evento al que lo hemos asociado.

Primero tendremos que instanciar un objeto de la clase EventEmitter, que hemos guardado en la variable EmisorEventos antes.

```
let ee = new EmisorEventos ();
```

Luego tendremos que usar el método on() para definir las funciones del manejador de eventos, o su equivalente addEventListener(). Para emitir un evento usamos el método emit().

Por ejemplo, vamos a emitir un evento llamado "datos", con este código.

```
ee.emit('datos', Date.now());
```

Ahora vamos a crear una función de controlador de eventos asociada con el evento definido en "datos".

```
ee.on ('datos', función (fecha) {

console.log (fecha);

});
```

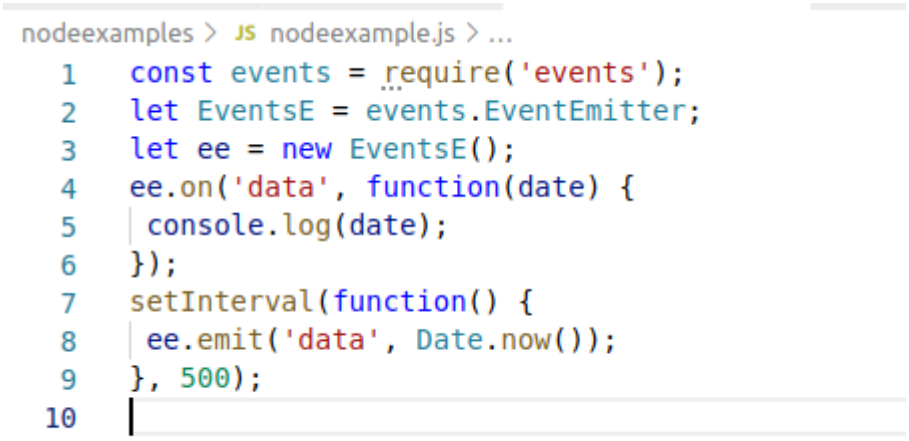
Si queremos aprovechar algunas de las características más interesantes de las aplicaciones de NodeJS, puede ser útil usar setInterval () para que podamos ir enviando datos de vez en cuando:

```
setInterval (función () {

ee.emit('datos', Date.now());

}, 500);
```

Con esto ya habremos construido un ejemplo de NodeJS totalmente funcional. El código completo sería el siguiente:



4. Asociación de eventos a objetos.

Creamos una clase que define a una persona y vamos a hacer que esta persona tenga emisores de eventos asociados a la herencia

```
const events = require('events');

class Person extends events.EventEmitter {
 constructor(name) {
 super();
 this.name = name;
 }
}

let manu = new Person('manu');
let boris = new Person('boris');
let people = [manu, boris];

people.forEach(function (littleperson) {
 littleperson.on('talk', function (message) {
 console.log(littleperson.name + ' has said ' + message)
 });
})

manu.emit(`talk`, `I hope you study node`);
boris.emit(`talk`, `I add that a lot`);
```