

## Node.js Events

Node.js is perfect for event-driven applications.

### Events in Node.js

Every action on a computer is an event. Like when a connection is made or a file is opened.

Objects in Node.js can fire events, like the `readStream` object fires events when opening and closing a file:

#### Example:

```
var fs = require('fs');

var readStream = fs.createReadStream('./demofile.txt');

/*Write to the console when the file is opened:*/
readStream.on('open', function () {
  console.log('The file is open');
});
```

#### output

The file is open

## Events Module

Node.js has a built-in module, called "Events", where you can create-, fire-, and listen for- your own events.

To include the built-in Events module use the `require()` method. In addition, all event properties and methods are an instance of an `EventEmitter` object. To be able to access these properties and methods, create an `EventEmitter` object:

```
var events = require('events');
var eventEmitter = new events.EventEmitter();
```

### The EventEmitter Object

You can assign event handlers to your own events with the `EventEmitter` object.

In the example below we have created a function that will be executed when a "scream" event is fired.

To fire an event, use the `emit()` method.

Example:

```
var events = require('events');  
var EventEmitter = new events.EventEmitter();
```

```
//Create an event handler:  
var myEventHandler = function () {  
  console.log('I hear a scream!');  
}
```

```
//Assign the eventhandler to an event:  
eventEmitter.on('scream', myEventHandler);
```

```
//Fire the 'scream' event:  
eventEmitter.emit('scream');
```

**Output:**

```
  I hear a scream!
```

Example:

```
const EventEmitter = require('events');  
const emitter = new EventEmitter();
```

```
// Register a listener  
emitter.on('messageLogged', function(){  
  console.log('Listener called');  
});
```

```
// Raise event  
emitter.emit('messageLogged');
```

**Output**

```
  Listener called
```