Node.js Event Emitter Cheat Sheet

This cheat sheet covers the basic principles around Node.js event emitters.

EVENT EMITTER

Node.js allows us to create and handle custom events easily by using the events module. The Event module includes the EventEmitter class which can be used to raise and handle custom events.
Some modules using the EventEmitter class:

- Http
- Streams

Event Emitter Class

You can get the EventEmitter class from the Node.js built in events module.

```
const { EventEmitter } = require('events');

const myEmitter = new EventEmitter();
```

EventEmitter.prototype.emit()

Emits an event to all subscribed listeners with some data.

```
const myEmitter = new EventEmitter();

myEmitter.emit('eventOne', { msg: 'This is a message...' });
```

EventEmitter.prototype.on()

Listens to a specific event and its data.

```
myEmitter.on('eventOne', (data) => console.log(data.msg));
```

EventEmitter.prototype.addListener()

Same as the myEmitter.on() method.

```
myEmitter.addListener('eventOne', (data) => console.log(data.msg));
```

EventEmitter.prototype.once()

Sometimes you want your application to respond to an event (or type of event) only one time (i.e., the first time the event occurs).

```
myEmitter.once('eventOne', () => console.log('This callback triggers only once!'));
```

EventEmitter.prototype.removeListener()

Remove a listener from the listener array for the specified event.

```
const callback = () => {
  console.log('someone connected!');
};

myEmitter.on('connection', callback);

myEmitter.removeListener('connection', callback);
```

EventEmitter.prototype.removeAllListeners()

Removes all listeners, or those of the specified event.

```
myEmitter.removeAllListeners(['eventOne']); // receives an array of events
```

EventEmitter.prototype.setMaxListeners()

Sets the maximum allowed listeners for a specific event. Node.js will print a warning if more than 10 listeners are added for a specific event. setMaxListeners allows you to modify this limit.

```
myEmitter.setMaxListeners(25);
```

EventEmitter.prototype.listeners()

Returns all the listeners subscribed to the specified event.

```
myEmitter.on('eventOne', (data) => console.log(data.msg));

myEmitter.listeners('eventOne'); // returns a list of functions
```

EventEmitter.prototype.getMaxListeners()

Returns the current max listener value for the EventEmitter instance.

```
myEmitter = new EventEmitter();

myEmitter.getMaxListeners() // defaults to 10
```

EventEmitter.listenerCount()

Returns the number of listeners for a given event.

```
const myEmitter = new EventEmitter();

myEmitter.on('eventOne', (data) => console.log(data));

EventEmitter.listenerCount(myEmitter, 'eventOne'); // output: 1
```

Events.defaultMaxListeners()

Sets the default max listens for all EventEmitter instances. However, calling emitter.setMaxListeners(n) still has precedence over events.defaultMaxListeners.

```
const { defaultMaxListeners } = require('events');

defaultMaxListeners(25);
```