**Testing and Debugging**

**Errors, Exceptions, and Warnings**

* Errors are caused when something goes wrong in a program.
* They are caused by: System error, programmer error, user error.

**Exceptions**

* An exception is an error that produces a return value that can then be used by the program to deal with the error

**Warnings**

* A warning can occur if there’s an error in the code that isn't enough to cause the program to crash.

**Use Strict**

* Strict mode encourages a better quality of JavaScript to be written that befits a programmer, so its use is recommended.
* Tu use to use it in the line of a JavaScript file: **‘use strict’;**

**Debugging Tools**

* Most modern browsers also have a debugging tool that allows you to set breakpoints in your code that will pause it at certain points.

**try, catch,** and**finally**

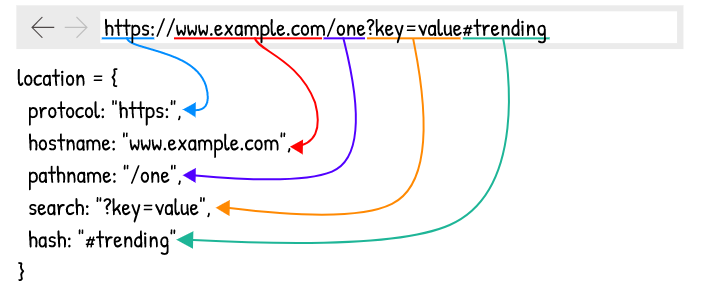
* The code inside the catch block will only run if an exception is thrown inside the try block. The error object is automatically passed as a parameter to the catch block.
* Code **try { …code } catch { …code } finally { …code }**

**Jest**

* [Jest](https://facebook.github.io/jest/) is a TDD framework, created by Facebook, that has gained a lot of popularity recently. It makes it easy to create and run tests by providing helper methods for common test assertions.
* Install with **npm install -g jest**

**How Single-Page Applications Work**

* **SPA** = Single Page Application
* Website that re-renders its content without making a request to the server to fetch new HTML.
* **SPA** use **window.location** this allows you to interact with the different parts of the URL without having to parse it yourself.

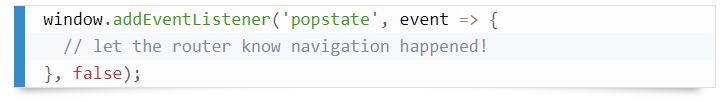


**Route Matching**

* Single-page application generally rely on a router.
* These can be static (/about) or dynamic (/album/:id, where the value of :id can be any number of possibilities) paths

**Detecting back/forward button navigation**

* When the back and forward buttons are clicked (as well as when history.go() is called), the browser emits a popstate event. In order to detect these, we can add an event listener to the window object.



**The HTML5 <template> element.**

* Reusable fragment of HTML that can be manipulated just like you would the contents of the document itself, but without the overhead of actually updating the DOM or having to compile and parse strings of HTML.
* The HTML5 <template> element is actually very well supported.