**Week 04**

**Debugging**

Debugging is the process of finding and fixing errors within a script, it's possible to do this through developer tools in "Sources", there are three parts: File Navigator, Code Editor, JavaScript Debugging.

File Navigator is where the files stay; Code Editor part is where we see the code and JavaScript Debugging is where have the information about our debugging. In code editor part we can put one or more breakpoints where the debugger will automatically pause the JS execution. The features available to debug are in JavaScript Debugging and they are:



1 - Resume: Resumes the execution.

2 - “Step over”: run the next command, but *don’t go into a function*

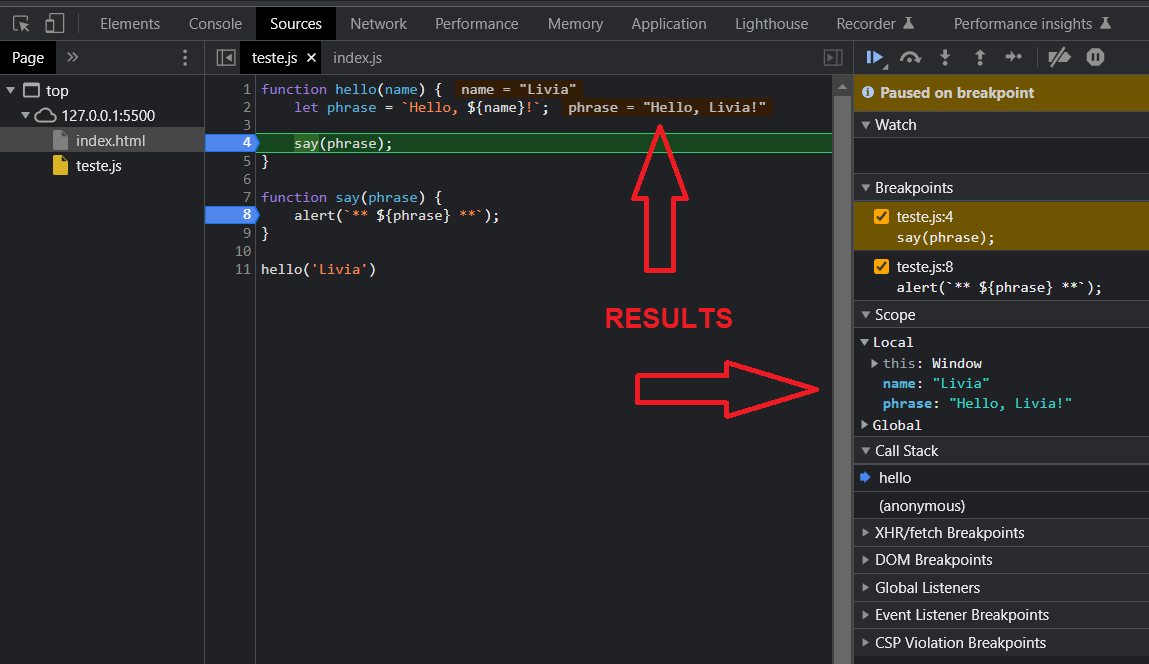
*3 -* “Step into”: *go into a function*

*4 -* “Step out”: continue the execution till the end of the current function

5 - “Step”: run the next command

6 - Enable/Disable all breakpoints

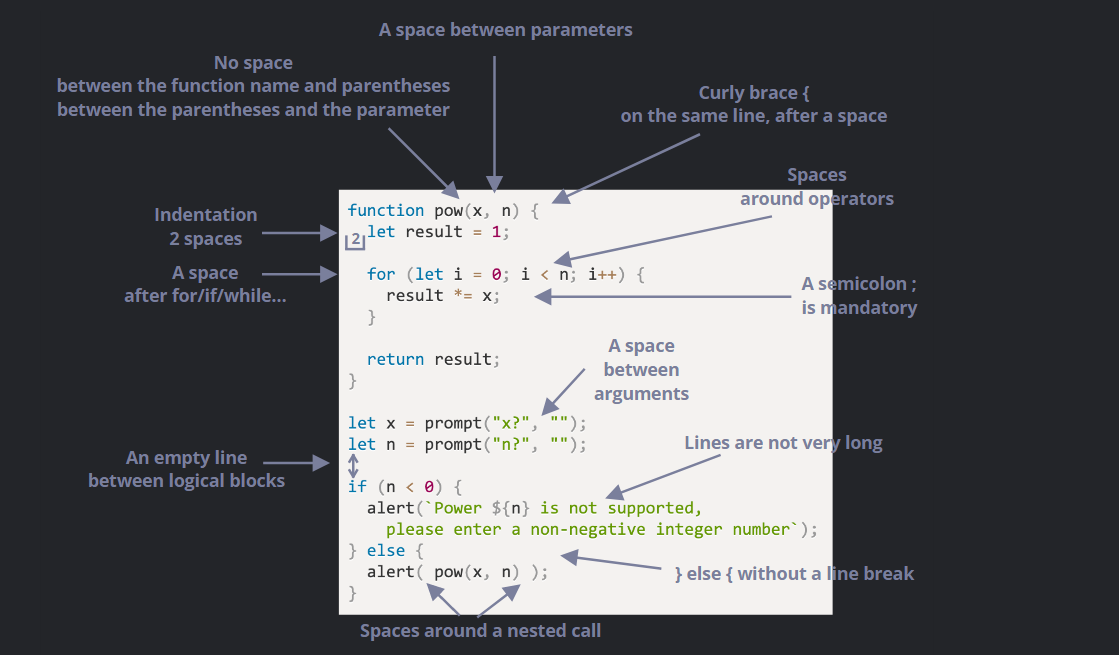
7 - Enable/Disable automatic pause in case of an error



The debugging results can be seen in Code Editor, JavaScript Debugging as picture above.

**Coding Style**

Syntax:



**Observations:**

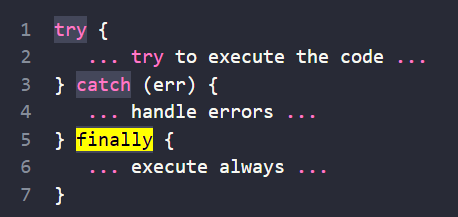
* About length, the line usually has 80 or 120 characters.
* The limit for there not to be a vertical indent is nine lines, after that the indent must exist.

## **Automated Linters**

Linters are tools that can automatically check the style of your code and make improvement suggestions and can also find bugs too, some of these tools are: JSLint, JSHint, ESLint.

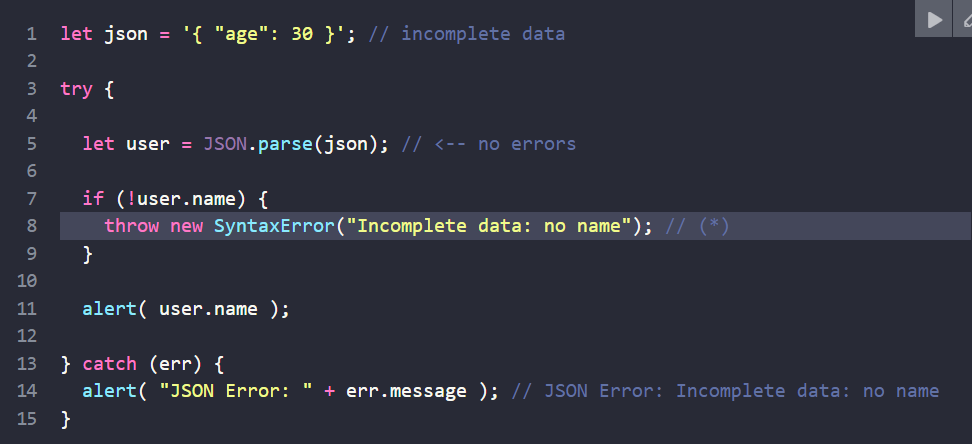
# **Error handling, "try...catch"**

The *try...catch...finally* statements are the way to handle errors and exceptions. In the *try* block is where we place the code that will be executed, if it does not contain errors or some type of exception, in case of an error, what was coded in the *catch* block will be triggered, thus being able to have a less "scary" view for the user. In the *finally* block it means that it will be executed regardless of errors, but this block is optional.



An exception can be defined by the developer using *throw* and when this exception happens the code will be stopped and stop the *catch* block.

Example:



**Questions**

*Style:*

According to material about coding style semicolon is mandatory, is it? For I had read something that was optional.

*Debugging:*

While doing the example in the material something unexpected happened, regardless of the command I used (step over, step into, etc.) after doing the debugging and pressing the command again one new file was opened, but I haven't this file. Why did this happen?

