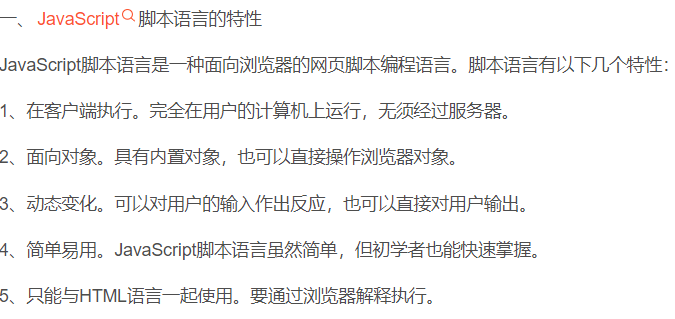
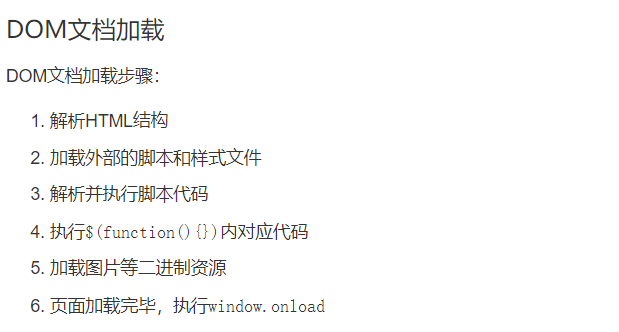
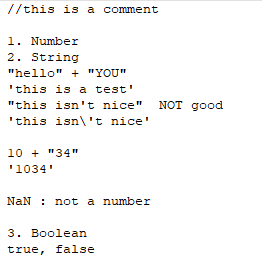
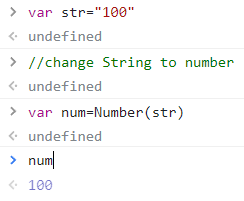
1/31/22, 2/17/22,3/31/22

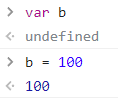
* **动态加载外部.js文件时候**
* **动态加载JS**

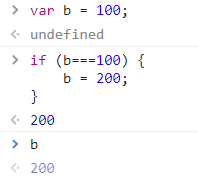
<https://blog.csdn.net/sgear/article/details/2043364?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_title~default-1.queryctrv2&spm=1001.2101.3001.4242.2&utm_relevant_index=3>

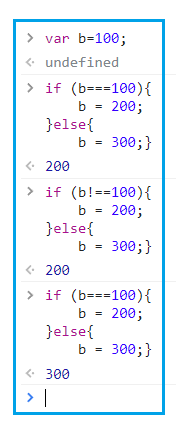


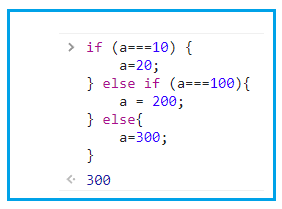


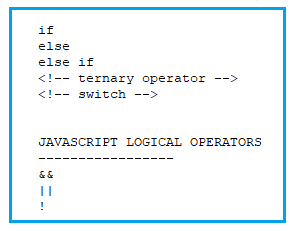






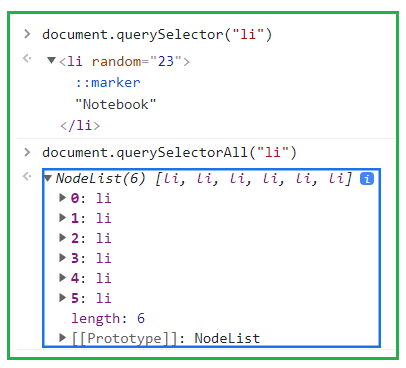
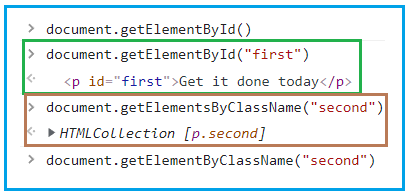


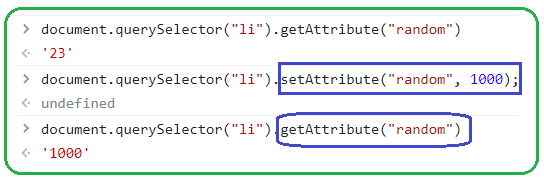


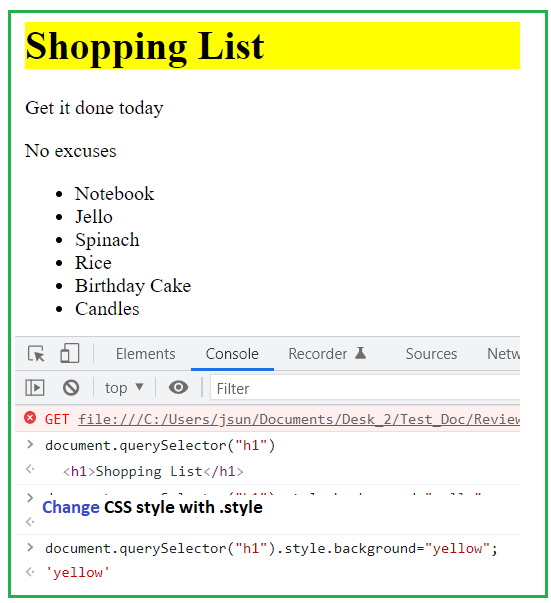


Javascript can

* Change all the **HTML elements** in the page
* Change all the **HTML attributes** in the page
* Change all the css styles in the page.
* Remove existing HTML elements and attributes.
* Add new HTML elements and attributes
* JavaScript can react to all existing **HTML events** in the page.
* JavaScript can create new HTML events in the page.







Events:

Keyboard and mouse

<https://developer.mozilla.org/en-US/docs/Web/events>

<https://developer.mozilla.org/en-US/docs/Web/API/Element#mouse_events>

In the previous video you saw something interesting:

Event listener syntax :

1. button.addEventListener("click", addListAfterClick);
2. input.addEventListener("keypress", addListAfterKeypress);

You didn't see the function being called like this as you may have expected:

1. button.addEventListener("click", addListAfterClick());
2. input.addEventListener("keypress", addListAfterKeypress(event));

This is something called a callback function. When that line of javascript runs, we don't want the addListAfterClick function to run because we are just adding the event listener now to wait for click or keypress. We want to let it know though that we want this action to happen when a click happens. So the function now automatically gets run (gets added the ()) every time the click happens. So we are passing a reference to the function without running it.

