* JavaScript Performance

⟹ . performance.now()

This method returns a high resolution timestamp in millisecond. It represents the time elapsed

* How to speed up your JS code?

→ Reduce activity in loops

→ Reduce DOM access

→ Reduce DOM size

→ Avoid unnecessary variables

→ Delay JS loading

* Reflow

It means re-calculating the position and geometries of elements in the document, for the purpose of re-rendering part or all of the document.

* Repaint

It is nothing but the repainting elements on the screen as the skin of element

change which affects the visibility of an element but do not affects layout.

Example.

i) changing visibility of an element

ii) changing background.

Repaint is a relatively faster process than the reflow.

The less the number of reflow and repaint the faster JS will be.

Let say if we are working with 100 paragraphs the there will be reflow and repaint 100 time which makes JS slower.

To avoid this we use ble below concept.

* DocumentFragment

It is a lightweight version of Document that stores a segment of a document structure comprised of nodes just like a standard document.

If we add 100 paras here then there will be no reflow and repaint.

The only reflow and repaint will happen when this fragment will be added to the document.

Fragment creation   .createDocumentFragment();

→ JavaScript is a single threaded language.

→ It follows asyncronous approach.

* Call Stack

A call stack is a mechanism for an interpreter to keep track of its place in a script that calls multiple functions — what a function currently being run and what functions are called from within that function, etc.

* Event Loop

An event loop is a mechanism in JS that allows it to handle multiple events and execute code asynchronously. It's a core part of the JS runtime environment and is responsible for managing the exception of code, handling events and maintaining the event queue.

For more details refer to "Philip Robert" video.

* Set setTimeout()

It allows you to schedule a function to be executed after a specified amount of time

Syntax ⇒ setTimeout (function, delay);