launch_code

Codergirl - JavaScript

Class 14

May 26, 2021

Agenda

- Studio coding activity – end at 6:00 pm

- Lecture – end at 6:30 pm

- Exercise – end at 7:00 pm

Studio – Communication Log

You and a partner will begin by coding in tag-team shifts. By the end of the task you should have a good idea about how to have two people work on the same code at the same time. You will learn how to:

1. Quickly add code in pull + push cycles *(Important! This is the fundamental process!)*

2. Add a collaborator to a GitHub Project

3. Share *repositories* on GitHub

4. Create a *branch* in Git

5. Create a *pull request* in GitHub

6. Resolve merge conflicts (which are not as scary as they sound)

This lesson reinforces:

1. Creating repositories

2. Cloning repositories

3. Working with Git concepts: Staging, Commits, and Status√
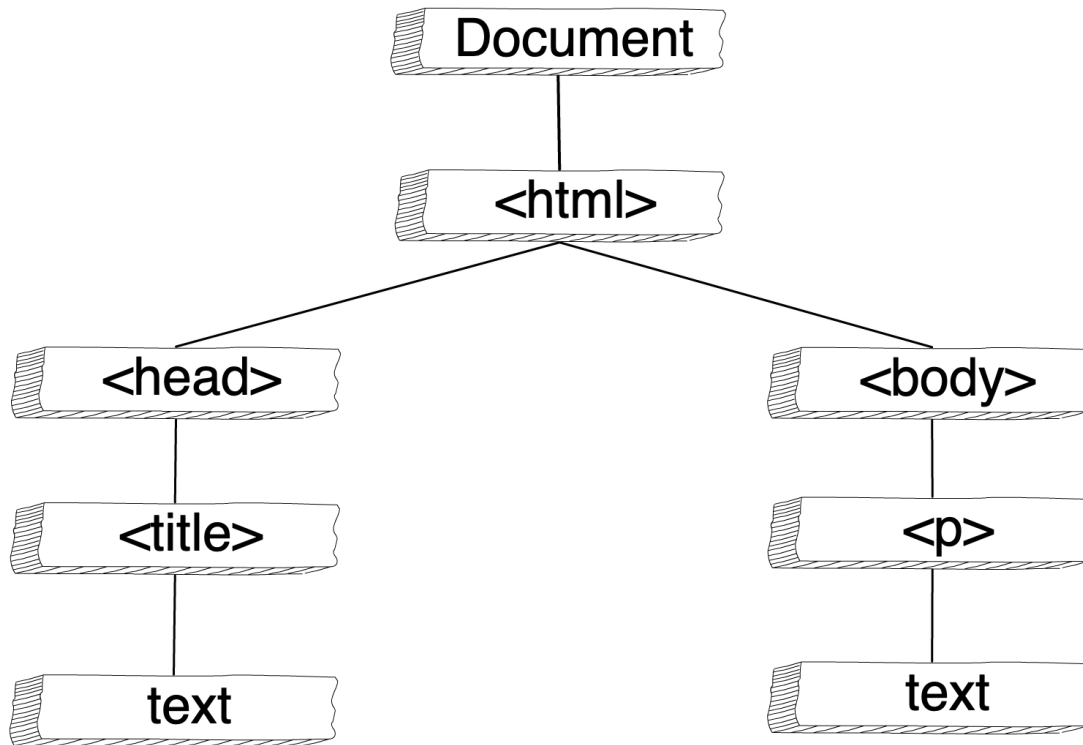

git merge

JS

- Static versus dynamic.
  - Static https://minimal-ceramics.xyz/
  - Mixed https://www.bombayfoodjunkies.com/
  - Likes or visitors on page, product info.

- **V8** is Google's open source high-performance JavaScript and Web Assembly engine, written in C++, Firefox uses SpiderMonkey, Mozilla uses Rhino

- https://caniuse.com/

Script tag

- <script> tag – JS executed by browser

- Inline

  <**script**> *// JavaScript code goes here!* console.log("Hello from inside the web page!"); </**script**>

- External

  - <**script src** = "myjs.js"></**script**>

The DOM

```
                    Document
                        |
                     <html>
                    /        \
             <head>            <body>
                |                 |
            <title>             <p>
                |                 |
             text              text
```

DOM – methods and properties

- JS can alter the DOM – re-render

- window – browser window or tab

Method or Property	Syntax	Description
alert	`window.alert("String message")`	Displays a dialog box with a message and an "ok" button to close the box.
confirm	`window.confirm("String message")`	Displays a dialog box with a message and returns **true** if user clicks "ok" and **false** if user clicks "cancel".
location	`window.location`	Object that represents and alters the web address of the window or tab.
console	`window.console`	Represents the debugging console. Most common and basic use is `window.console.log()`.

DOM – methods and properties

- document – html page

Method or Property	Syntax	Description
title	document.title	Read or set the title of the document.
getElementById	document.getElementById("example-id")	Returns a reference to the element that's id attribute matches the given string value.
querySelector	document.querySelector("css selector")	Returns the first element that matches the given CSS selector.
querySelectorAll	document.querySelectorAll("css selector")	Returns a list of elements that match the given CSS selector.

DOM – methods and properties

- element– html element

Method or Property	Syntax	Description
getAttribute	`element.getAttribute("id")`	Returns the value of the attribute.
setAttribute	`element.setAttribute("id", "string-value")`	Sets the attribute to the given value.
style	`element.style.color`	Object that allows reading and setting *INLINE* CSS properties.
innerHTML	`element.innerHTML`	Reads or sets the HTML inside an element.

DOM – events

- Triggered – sent

- Handled – receiving and performing action

- Events – JS event driven, handle user interaction, dynamic, state – loading, styling, displaying elements.

- Different event types – load, click, mouseover

DOM – events

- ## Inline Event handler.
    - *On click of the button, print a message to the console.*
    - <**button onclick**="console.log('you rang...');">Ring Bell</**button**>

- ## Event listener.
    - If the code *hears* the event, then the event is handled.
    - anElement.addEventListener("eventName", aFunction);
    - button.addEventListener("click", youRang);
    - A benefit of using addEventListener is that an *event* parameter can be passed to the event handler function. This event is an object instance of the Event class, which defines methods and properties related to events.

- ## Event Bubbling.

Examples and Exercises

Questions?

Studio time!