# Web Developer Roadmap [Advanced]

## 12 Week Plan Overview

### Things You Will Learn In This 12 Weeks

- Advanced HTML and CSS
- JavaScript deep dive
- ReactJS Fundamentals
- TypeScript
- Advanced Web Development Concepts
- Algorithms and Data Structures Fundamentals
- Git power user
- Interview Prep

# Table Of Content

# 12 Week Plan Details

## Week 1

- HTML
  - Refresh the tags
    - `HTML, Body, Head, Nav`
    - `Div, P, Section, Span`
    - `Anchor, Images`
    - `Lists`, `Tables`
  - What is DOM
  - How does DOM work
  - What is DOM hierarchy
- CSS
  - Box-model
  - Selectors - `Id, Class, Element-tag`
  - Properties - `Color, Background-color, Border`
  - Properties - `Margin, Padding`
  - Decoupling CSS and HTML
  - What and Why CDN
- JavaScript
  - Primitive data types - `Numbers, Boolean, String, etc`
  - Complex data types - `Array, Objects`
  - Variables - `var, let, const`
    - What is the difference between them
    - When to use which type of declaration
- Exercises
  - Write a function that takes in an array of mixed data types and returns the array in reverse order
  - Implement a clickable image that will take you to your favorite website

# Week 2

- HTML
  - Input and Button tags
    - Input types - `text, password, email, etc.`
    - `Submit` type input
  - `Form` element
    - Attributes of form
    - Submit/Post form
  - Attributes vs Properties
  - When to use Attributes vs Properties
  - Builtin and Custom attributes
  - Web storage, local storage, session storage
    - Difference between each storage type
    - Use cases for each storage type
- CSS
  - Parents, siblings, and other selectors
    - Play around with each type of selectors
    - Grab sibling selector
    - Grab sibling in only odd or even position
    - Can you select a parent from a child element
  - Pseudo-classes
    - And pseudo-elements
    - What and why do we use them?
  - CSS Gradients
    - Create beautiful backgrounds using gradients
    - Create beautiful buttons, hero components using gradients
- JavaScript
  - `Truthy and Falsy` values
    - List of such values
    - How to check if a variable is truthy or falsy
  - Conditionals - `if, else, switch, ternary`
    - How to write nested ternary
  - Event listeners
    - Create event listeners
    - Remove event listeners
  - Event handling
  - Event bubbling, delegation, propagation
    - What is event bubbling
    - What is event capturing
    - Different between target and currentTarget property
- Chrome Developer Tools
  - Debugging your function using console logs
  - What is JSON
  - `JSON.Stringify`

- Change JavaScript Object to JSON
- Change JSON to JavaScript Object

- Exercises
    - Create a registration form that will take username, password, email, telephone number
    - Hide an element on a click event anywhere outside of the element?
    - How to stop events bubbling in pure JavaScript?

## Week 3

- HTML
    - Accessibility
    - Unordered Lists
    - Ordered Lists
    - `Form` element
        - Form validation
        - Show/hide errors depending on error states
    - Radio buttons
        - Checkboxes
        - When to use Radio buttons vs When to use Checkboxes
    - What are cookies
        - How does e-commerce sites use cookies
        - How do ads companies use cookies
- CSS
    - Changing CSS with pure JavaScript
    - `Add, remove, and toggle classes` with JavaScript
- JavaScript
    - Loops
        - `for, while`
        - When to use which?
        - Are there any performance differences between them?
    - Arrays and collections
        - When to use arrays?
        - How to traverse arrays?
    - Scoping and Hoisting
    - The `"new"` keyword
        - How to create new objects in JavaScript
    - The `"this"` keyword
        - How is `this` different in JavaScript
        - Examples of how `this` is different depending on the context

- Bind, call, apply
- Chrome Developer Tools
    - Console tab - log, dir, group, table
    - Debugging using console statements
    - Tip and tricks of using dev tools
- Exercises
    - Create a TODO application using list elements and event listeners - pure HTML and JavaScript
        - Add todo item
        - Remove todo item
    - Perform form validation
        - Show/hide message toast-bar depending on the error state
    - Use Checkboxes to mark complete/todo task
    - Use a Radio button to set the priority of the task

## Week 4

- CSS
    - Introduction - What and Why Flexbox
    - Flexbox properties
        - `flex-direction`
        - `flex-wrap`
        - `just-content`
        - `align-items`
- Web Development
    - HTTP fundamentals
        - What is HTTP
        - How does internet work?
    - How does browser work?
    - What happens behind the scenes when you visit any site?
        - Search about this on `ngninja.com`
    - What is DNS
        - How is IP address resolved
    - Difference between IP and Mac address
- JavaScript
    - What are Callback functions
    - Objects and Prototypes
        - What are prototypes?
        - What is prototype chaining?

  - Prototypal Inheritance
    - How is this different than Class-based Inheritance?
  - Fun with array methods
    - `map, reduce, filter`
  - More Fun with array methods
    - `find, concat, reverse`
  - Bind, call, apply
    - Difference between them
    - When to use which method
  - Function as first-class citizens
    - What do first-class citizens mean?
- Git
  - What and why Git
  - Install Git locally
  - Merging `remote` branch X into your `local` branch Y
  - Deleting a local branch
  - Create a Git repo
  - What are branches
  - Committing, pushing, and pulling changes
    - What are the commands for these operations?
    - What flags do they take
- Exercises
  - Create a navigation bar using Flexbox
  - Create a Git repo with two branches. Commit changes to one of the branches. Merge those commits to the other branch.

## Week 5

- CSS
  - `“Initial”, “Inherit” and “Unset”` Properties
  - Normalizing and Validating your CSS
    - Why to do it
    - How to do it
  - Difference between `em, rem, px, %`
    - When to use which
  - Media queries
    - Target mobile, tab, and desktop resolutions
  - Writing responsive CSS for all the major platforms
- JavaScript

  - Closures
  - Composition
  - Intervals and Timers
  - Async programming
    - Promises, Observables, `async-await`
  - Recursion
    - Why to use recursion?
    - Design patterns of recursion
    - Tree data-structure
      - DFS and BFS algorithm using recursion
- ReactJS
  - What is React
  - Why React
  - React vs Vue vs Angular
  - What are SPAs
    - Single Page Applications
    - Why SPAs are better than traditional web apps
- Exercises
  - Create a blog
    - Page for list of all the posts
    - Page for the post
    - Make it responsive for mobile, tablet, and desktop
    - Make the images responsive too
  - Make the navigation bar responsive
    - Show `hamburger` icon with a dropdown for mobile platforms

## Week 6

- JavaScript
  - Immutable Data Structures
  - JavaScript as Functional Programming language
  - Currying
- ReactJS
  - Installing ReactJS
  - What is JSX
  - What is a React component
  - Why do you need components
  - Creating your first React App
    - using `create-react-app` utility

- Git
    - Cherry-picking a particular commit
    - Rebasing your branch with the other branch
    - Creating pull requests
    - Squashing multiple commits into one single commit
- Chrome Developer Tools
    - Networks tab - All, XHR, JS
    - Networks tab - Inspect XHR query request
    - Networks tab - Inspect XHR query response
- Exercises
    - Create a react application
    - Create a React component that displays details of a TODO task
        - task name, description, due dates, priority
    - Demonstrate cherry-picking a particular commit
        - Commit changes to branch X
        - Goto branch Y and cherry pick the last commit from branch X and push it to the branch Y

## Week 7

- HTML
    - Adding multimedia on webpages
        - Adding videos
        - Adding sounds
        - `<object>` Element
        - Embedding YouTube Videos
- CSS
    - 2D and 3D transformations
    - Animations
    - Keyframes
- JavaScript
    - JavaScript Classes
        - What are classes
        - How is JavaScript class different than other language class like Java
    - Arrow functions
    - Block scope
        - How to achieve blog scope for variables
- ReactJS
    - Function Components

    - Class components
        - Difference between different component types
    - How to reuse components inside other components
    - `states and props`
        - What and why state variables
        - What and why prop variables
- Exercises
    - Embed your favorite YouTube video on a webpage
    - Create a bouncing ball that bounces from floor to ceiling forever

## Week 8

- HTML
    - Implementing drag and drop using HTML5
- CSS
    - What is Shadow DOM?
    - Why is Shadow DOM important?
- JavaScript
    - The Lexical `this`
        - What is it?
        - How is it different from the regular `this` in JavaScript
    - Spread operator
        - What is it?
        - What are it's usecases?
    - Rest operator
        - What is it?
        - What are it's usecases?
    - What is destructuring
- ReactJS
    - How to do two-way binding in React
    - Adding styles to your react component
        - External styles
        - Inline styles
        - Object notation styles
- Git
    - Writing good commit messages
        - Format/Standard to follow
        - Why writing good commit messages is important?
        - Commitizen

    - Branching strategies
        - Different options available
        - Pros and cons of branching strategies
        - How to manage dev, QA, and production branches
- Exercises
    - Write a function to clone a JavaScript object using ES6+ features
    - Implement drag and drop feature to upload a file by dropping it on a web page
    - Create a React component that displays a list of TODO tasks

## Week 9

- HTML
    - HTML5 Web Workers
        - What is a Web Worker?
        - Implementing web worker
        - Does given browser support web worker?
- JavaScript
    - Map + Set + WeakMap + WeakSet
        - What are those?
        - What are the differences between them?
        - When to use which data structure?
    - What is a call stack
    - What is a task queue
    - What is an event loop in JavaScript
- ReactJS
    - Rendering content conditionally
    - Rendering `async` data in your React component
        - Get the data
        - Display it on your React app
    - Suspense react component
        - What does it do?
        - How to use it?
        - Show a loader when you are retrieving the data
- Exercises
    - Given an array which contains duplicate elements write a function which returns an array of unique elements
    - Fetch a data in your React application and use `Suspense` feature to gracefully show the empty state loaders and the data once fetched

# Week 10

- ReactJS
    - React hooks
        - What are hooks
        - Why do you need hooks
        - What problem does it solve?
    - Different React hooks - commonly used ones
        - `useState` hook
        - `useEffect` hook
        - `useRef` with React hooks
    - What is Context API
        - How to use it?
    - `useContext` hook
- JavaScript
    - What is Execution Context
        - How does JavaScript code and functions get executed?
        - How does `setTimeout` and `setTimeInterval` work under the hood?
    - All about memory leak
        - What causes memory leaks in JavaScript
        - How to avoid them?
        - How to debug memory leaks?
    - How-to Garbage Collection
        - What is Garbage Collection
        - How does Chrome handle Garbage collection
    - The JavaScript Engine
        - React about the Chrome's V8 JavaScript Engine
- Mini project
    - Build a Tic-tac-toe game with ReactJS
        - Create a border for the tic-tac-toe game
        - Add click-handler that will render X on odd click numbers
        - Add click-handler that will render O on even click numbers
        - At the end reveal the winner in GREEN color -> Even Or Odd

## Week 11

- JavaScript
    - Debugging in JS
        - All the tricks to debug JS code
        - How to attach debugger in VSCode
    - What is debounce
        - How to implement it?
        - What are its use cases?
    - What is throttle?
        - How to implement it?
        - What are it's usecases?
    - JavaScript good practices
    - JavaScript bad practices
- ReactJS
    - Adding Typescript to your application
    - Why Typescript?
    - What problems does Typescript solve?
- Real-world project 1
    - Random Challenge App For Bored People
        - A button that will randomly suggest from a database of activities, brain-teasers, or simple games
        - Win virtual coins after the user completes the challenge

## Week 12

- Real-world project 2
    - Build a clone of Instagram app
        - Use ReactJS framework
        - Use the Instagram public API to fetch the data from
        - Implement the timeline view
        - Implement the profile view
        - Bonus: Solve any one major pain-point that you struggle with when using the Instagram application