

Web Developer

2 Month Area of Focus

This outline is curated for Frontend Web Development. Having full grasp of the outlined concepts will enable you to create the interface specified at the end of this document.

Prospective Outline

1

How The Internet Works

- Browsing the web
- Breaking Google: Chrome Inspector → Breaking Google yourself
- Internet Backbone
- Traceroute
- Who is a Web Developer? Understanding the concepts Frontend Developer, Backend Developer and Full Stack Developer

2

HTML 5

- HTML Boilerplate
- Developers Fundamentals: Sourcing for solutions
- HTML Tags (h1 to h6, p, b or strong, i, or em, *generating dummy texts with lorem,
- lists(ol, ul, li), nested lists.
- Self closing HTML tags (br, hr, img) and Attribute.
- Anchor tag (a), href and *target attribute
 - Working with multiple pages

3

ADVANCED HTML 5

- HTML 5 Forms
 - label and attribute(for)
 - Input and it's attributes (id, type, name, value)
 - select (with attribute multiple) and option (with attribute value)
 - input - password with attribute min
 - submitting a form: concept GET/POST form attribute
 - Query strings
- HTML vs HTML 5 (header, nav, section etc.)
- Copy a website via Chrome developer option

4

CSS

- Working with CSS (external, internal and inline)
- CSS properties(css-tricks.com/almanac): Text CSS(font-size, font-weight etc.), Object CSS(background (image, size, position), border,)
- Color in CSS
- Internal and external CSS (selectors)
- class and id
- CSS Box model
- CSS Measuring units

5

ADVANCE CSS

- Critical Rendering (code minification)
- Flexbox (parent and items)
- CSS property position (relative, absolute, fixed, static) and position setters (top, right, bottom, left)
- CSS 3 features
 - transform & transitions
 - box-shadow
- Responsive Design

6

CSS GRID and LAYOUT

- Website Layout
- Grid vs Flexbox
- CSS GRID 1: grid on parent (gap, grid-template-columns, grid-template-rows, grid-template)
- CSS GRID 2: grid on parent (justify-items, repeat() function, size specifications (auto-fill et size, autofill et minmax(min, max))
- CSS GRID 3: grid on children (grid-column-start and -end, grid-column: a/b or span, grid-row, grid area: ra/ca/re/ce or item-name)
- CSS GRID 4: grid on children (justify-self, align-self)
 - noshit.xyz

7

JAVASCRIPT

- Introduction
 - JS datatypes
 - Numbers (arithmetic: +, -, %, *, /)
 - String (string, string concatenation, converting number to string and vice versa)
 - boolean & comparison (>, <, >=, <=, ==, ===, !==)
- Variables
 - prompt() & alert() get inputs and displaying input.
- Control flow
 - if, if...else, if...else if...else
 - Logical operators (AND, OR, NOT)
- JavaScript in website(internal and external)
 - Js in head and last thing in body (How the webpage loads)
- Functions
 - Pre-existing functions and their structure (alert(), prompt(), console.log()) and the concepts of arguments
 - Custom Functions: creating and calling
 - function declaration
 - The concept: parameters and arguments
 - why does function returns **undefined**
 - Control flow in a function to determine what to return
 - function expression
 - anonymous function
 - put it together to form a function expression

- Data Structures
 - Array
 - Creating an array
 - Accessing each item in an array (position and index position)
 - Changing the value of an item
 - Array with multiple datatypes as items
 - Nested array
 - Accessing items in a nested array
 - Array methods (shift, pop, push, unshift, length, concat, basic level sort, reverse, includes, join, slice and splice)
 - Objects
 - Creating an object
 - Property value pair
 - variable as property name
 - variable as property and value
 -
 - Accessing the element/item in an object
 - .property and ('property')
 - Change the value of the property
 - Adding new property with value
 - Create with different types(datatypes, function, array etc.) of values
 - List of objects
 - Accessing the objects
 - Accessing the values of properties in each object
 - Console as object
 - Creating an empty object with {} and null
 - {} vs null:
 - Putting this concept array and object together
 - Sample: Build a login logic
 - JavaScript Terminologies:
 - function declaration
 - function expression
 - calling or invoking a function
 - expression
 - function vs method
 - Loops
 - for loop - add ! to end of all strings in an array
 - while loop
 - do...while loop
 - forEach array method
 - JavaScript Keywords

8

DOM Manipulation

- The DOM
- DOM Selectors
 - `getElementByTagName("tag")`
 - `getElementById("id")`
 - `getElementsByClass("class")`
 - `querySelector("selector")`
 - `querySelectorAll("selector")`
 - `getAttribute("attr")`
 - `setAttribute("attr", "value")`
 - `parentElement`
 - `children`
 - `innerHTML`
 - Changing style
 - `style.property`
 - `className`
 - `classList`
 - `.add`
 - `.remove`
 - `.toggle`
- DOM Events
 - `addEventListener()`
 - creating a new html element in JS and adding to the HTML dynamically
 - `document.createElement("element")`
 - `elementSelected.appendChild(newElement)`
 - text - `document.createTextNode("text")`
 - Other DOM events
 - `keypress`, `hover`, `focus` etc

9

ADVANCE JAVASCRIPT

- Scope
 - function scope
- Advance Control flow
 - Ternary operator
 - `Switch...case...default`
- ES5 vs ES6
 - `var` to `const`, `let`
 - default arguments
 - arrow function
- Advance function
 - Closure
 - Currying
 - Compose
 - Avoiding side effects (logs in functions)
- Advance Arrays
 - `forEach`
 - `filter`
 - `find`

- `findIndex`
- `indexOf`
- `map`
- `reduce`
- `some`
- Advance Objects
 - Reference type
 - Context vs Scope
 - Instantiation: Object constructor
- Type Coercion
- Pass by value and Pass by reference
- ES7 (ESE6)
 - array method `includes()`
 - arrow function
- ES8 (ESE7)
 - `object.values`
 - `object.entries`
 - `object.keys`
- ES10 (ESE9)
 - array method:
 - `flat()`
 - `flatMap()`
 - `trim()` - `trimStart()`, `trimEnd()`
 - `Object.fromEntries()`
 - `try...catch` block
- Advance loops
 - Iterating arrays and strings: `for of`
 - enumerating object's properties
- ES2020
 - `BigInt`
 - optional chaining operator
 - Nullish coalescing operator
- ES2021
 - string and array method: `replaceAll()`
-
- Debugging
- How JavaScript works
- Modules
 - `export` and `import`

10

COMMAND LINE

- Using the terminal

11

GIT + OPEN SOURCE PROJECTS

- Interface walkthrough
- creating a repo
- git from terminal:
 - `git add`, `git commit`, `git clone`, `git push`
- Collaboration with git and contributing to open source projects

12

NPM + NPM SCRIPTS

- Introduction to NPM
- Setting up NPM
- Installing and using packages
- Updating Packages

GIT + OPEN SOURCE PROJECTS

- Interface walkthrough
- creating a repo
- git from terminal:
 - git add, git commit, git clone, git push
- Collaboration with git and contributing to open source projects

13

REACTJS

- Introduction to React
- Introduction to JSX and Babel
- Javascript Expressions in JSX & ES6 Template Literals
- JSX Attributes & Styling React Elements
- Inline Styling for React Elements
- React Components
- Javascript ES6 - Import, Export and Modules
- Local Environment Setup for React Development
- Keeper App Project - Part 1 Challenge
- React Props
- React DevTools
- Mapping Data to Components
- Refresh:
 - Javascript ES6 Map_Filter_Reduce
 - Javascript ES6 Arrow functions
- React Conditional Rendering with the Ternary Operator & AND Operator
- State in React - Declarative vs. Imperative Programming
- React Hooks - useState
- Javascript ES6 Object & Array Destructuring
- Event Handling in React
- React Forms
- Class Components vs. Functional Components
- Changing Complex State
- Javascript ES6 Spread Operator
- Managing a Component Tree
- React Dependencies & Styling the Keeper App

Possible Interfaces that can be built

- Newbie Level (HTML and CSS)
 - Challenge 1: Blog preview card <https://www.frontendmentor.io/challenges/blog-preview-card-ckPaj01lcS>
 - Challenge 2: Result summary interface : https://www.frontendmentor.io/challenges/results-summary-component-CE_K6s0maV
 - Challenge 3: Huddle landing page with a single introductory section: https://www.frontendmentor.io/challenges/huddle-landing-page-with-a-single-introductory-section-B_2Wvxgi0
 - Challenge 4: Social proof section: https://www.frontendmentor.io/challenges/social-proof-section-6e0qTv_bA
- Beginner Level (HTML, CSS, Basic JS)
 - Challenge 1: Manage landing page: <https://www.frontendmentor.io/challenges/manage-landing-page-SLXqC6P5>
 - Challenge 2: Age calculator app: <https://www.frontendmentor.io/challenges/age-calculator-app-df9DFFpj-Q>
 - Challenge 3: Interactive card details form: <https://www.frontendmentor.io/challenges/interactive-card-details-form-XpS8cKZDWw>
 - Challenge 4: News homepage: <https://www.frontendmentor.io/challenges/news-homepage-H6SWTa1MFI>
- Intermediate (HTML, CSS, JS)
 - Challenge 1: Crowdfunding product page: <https://www.frontendmentor.io/challenges/crowdfunding-product-page-7uvcZe7ZR>

REACTJS

- Newbie Level(Component) to Beginner level : ALL VANILLA JS COVERED TO REACTJS
- Intermediate level (Basic React + API)
 - Challenge 1: URL shortening API landing page: <https://www.frontendmentor.io/challenges/url-shortening-api-landing-page-2ce3ob-G>