Full stack Web Development Bootcamp (12 weeks)

WEB DEVELOPMENT FUNDAMENTALS (One Week)

- Understand how websites work and how HTML, CSS and JavaScript contribute.
- Understand how the internet works and how websites are served.
- Learn to use git, GitHub and version control.
- Learn key troubleshooting and debugging skills to apply to your projects.

HTML 5

- Learn the anatomy of HTML syntax to structure your websites.
- Understand the HTML boilerplate and HTML doctypes.
- Understand indentation and nesting in HTML code.
- Learn to use HTML tags to structure headings, paragraphs.
- How to structure HTML lists to create unordered and ordered lists.
- How to insert images using HTML?
- How to create hyperlinks using anchor tags?
- Create multi-page websites. Learn HTML best practices.

CSS 3 (Two Weeks with Practical Projects)

- Understand what are cascading style sheets and how you can use it to style your website.
- How to use CSS selectors and properties.
- Learn about how to use inline, internal and external CSS.
- Understand CSS coding best practices.
- Learn about CSS specificity and inheritance.
- Learn and understand the CSS Box Model.
- Learn to use developer tools to inspect and diagnose CSS.
- Learn about CSS positioning and display properties.
- Learn about font styling using CSS and web safe fonts.
- Implement responsiveness using CSS media queries.
- How to use CSS float and clear properties.
- How combine CSS selectors and understand selector priority.

FLEXBOX

- Learn about why flexbox was introduced to CSS.
- Understand what happens when display is set as flex.
- Understand flex direction and how it affects layout.
- Learn to use align and justify for layout of child items.
- Understand how child items are sized and distributed in a flexbox and use the flex shorthand property

GRID

- Learn about how grid is different to flexbox and their strengths and weaknesses.
- Understand what happens when display is set as grid.
- Learn how grid sizing is done and how to create and arrange a grid.
- Learn to position items in a grid and how placement across rows and columns is done.
- Learn to use grid in combination with flexbox to create complex layout using native CSS.

BOOTSTRAP (One Week with Practical Projects)

- Understand the difference between native CSS tooling and external frameworks.
- Learn to install the Bootstrap framework into your website.
- Understand the Bootstrap 12 column layout system.
- Learn to use Bootstrap components such as buttons, carousels, cards and navigation bars.
- Accessing and incorporating designer icons from Bootstrap.
- Find and use bootstrap code snippets, examples and pre-built templates to quickly created beautiful mobile-first websites

WEB DESIGN (2 Days)

- Learn the fundamental principles of web design, UI and UX.
- Understand color theory and how to choose and use modern color palettes to make your website look professional.
- Understand modern typography and choose free-for commercial use typefaces to make your website user friendly and stylish.
- Learn to manage user attention using the key principles of User Interface (UI) design.
- Learn to design user friendly websites by understanding User Experience (UX) design
- DOCUMENT OBJECT MODEL (DOM)
- Learn the tree structure of HTML based websites. Traverse through the document using object notation. Separation of concerns and coding best practices. Manipulate and change the HTML elements using your understanding of the DOM.

JAVASCRIPT ES6 (4 Weeks with Practical Projects)

- Compare programming languages like JS with scripting and markup/stylesheet languages.
- Overview of JavaScript and its uses.
- Understand how to work with a code editor and IDEs.
- Basic syntax and data types in JavaScript, including: Variables and their scope (let, const)
- Data types (string, number, boolean, null, undefined, symbol)
- Operators and expressions, including arithmetic, comparison, and logical operators
- Control structures like if/else statements and loops,

including:

- The for loop and while loop
- The for...of loop to iterate over arrays
- The for...in loop to iterate over object properties
- Functions and their importance in JavaScript, including:
- Function declaration and expression
- Function scope and closures
- Arrow functions and their syntax
- Higher-order functions, including:
- Passing functions as arguments to other functions
- Returning functions from other functions
- The map, filter, and reduce methods on arrays and their use with higher-order functions
- Arrays and their methods, including:
- Creating and accessing arrays
- Adding and removing elements from arrays
- The slice method for slicing arrays
- The concat method for concatenating arrays
- The indexOf and lastIndexOf methods for finding elements in arrays
- The map, filter, and reduce methods for manipulating arrays
- The spread syntax ... to spread the elements of an array into a new array or function call Object-oriented programming in JavaScript, including:
- Creating objects with object literals Creating objects with constructor functions and the new keyword Creating objects with classes and the class keyword
- Adding and accessing properties and methods on objects
- The “This” keyword and how it works in object methods
- The super keyword for accessing parent class methods and properties
- Manipulating objects and arrays using methods and iteration, including:
- The Object. Keys and Object. Values methods for getting the keys and values of an object as arrays
- The for...in loop for iterating over object properties
- The Object. Assign method for copying properties from one object to another.

REACT.JS (10 days with Practical Projects)

- Learn front-end development with React.
- Understand when and how to use React Components.
- Learn to pass Props and work with them.
- Learn to write JSX and understand JSX syntax.
- Learn about the React DOM.
- Learn State Management in React.
- Learn about React Hooks.
- Learn about conditional rendering in React.

- Understand the difference between class and functional components

THE UNIX COMMAND LINE (One Week)

- How to use basic bash commands in a Unix/Linux Terminal.?
- How to manipulate files and folders without needing a graphical user interface.?
- How to download and install to your computer using command line.?

GIT GITHUB AND VERSION CONTROL

- Using git for version control and collaboration
- Git forking, branching and cloning
- Using GitHub as a remote repository
- Checkout and rolling back changes with git.
- Using git and GitHub with Xcode

NODE.JS (Two Weeks)

- Explore the components of back-end development, working with an MVC framework.
- Apply concepts like data types, objects, methods, object oriented programming, and classes in the context of back end development.
- Server-Side JavaScript
- Using Node on the command line
- NPM
- JavaScript Build Processes
- Event Loop and Emitters
- File System Interaction
- Modules
- Native Node drivers

EXPRESS.JS (One Week)

- Understand how to install and use express in Node applications
- Creating Node and Express based servers
- RESTful Routing with Express
- Understand and use middleware for Node applications

APPLICATION PROGRAM I NTERFACES (APIS) (3 days)

- Understand what APIs are and how they work.
- HTTP in Depth
- Calling APIs
- Reading API documentation

- Basic API Authentication
- Server to server communication
- JSON vs. XML, sending data over the wire

EJS

- Understand what EJS does and how to use it with Node and Express.
- Templating with EJS
- Running code in EJS templates
- Passing data from server to template and vice versa
- Creating layouts/partials with EJS

DATABASE FUNDAMENTALS (2 Weeks)

- Data Relationships
- Designing a Data Model
- Relational Databases
- Alternative Databases
- Entity Relationship Modelling (ERM) and Object Relational Mapping (ORM)

SQL DATABASES

- Working with Database Schemas
- Create-Read-Update-Destroy (CRUD)
- Database Joins
- Querying SQL databases

POSTGRES DATABASES

- PostgreSQL queries
- Create-Read-Update-Destroy (CRUD)
- Postgres Best Practices
- Installing Postgres
- Relational Databases
- Implementing One to Many, One to One and Many to Many Relationships.
- Working with Postgres in Node with the pg Package.

DEPLOYMENT (One Week)

- Understand hosting and deployment.
- Hosting static websites with GitHub Pages.
- Deploying server based applications with Heroku.
- Deploying Databases with Mongo Atlas.

BUILDING RESTful APIs

- Understand REST and guiding principles behind API design.
- Learn to work with a MongoDB GUI Robo 3T
- Implementing GET, POST, PUT, PATCH and DELETE by creating a public API from scratch.
- Understand and use chained route handlers from Express.

AUTHENTICATION and SECURITY

- Understand the need for authentication and keeping user details secure.
- Learn about Encryption and use encryption to keep your database secure.
- Learn and implement Hashing and Salting with bcrypt
- Using Sessions and Cookies to persist user log in sessions.
- Setting up local authentication from scratch.
- Implementing Passport to authenticate users quickly and effectively.
- Understand and use environment variables to keep secret keys secure.
- Understand and use OAuth 2.0 to log in users using Google and Facebook

WEB3 BASICS AND BLOCKCHAIN DEVELOPMENT

- Understand Web3 and block chain development.
- Understand a Canister.
- Understand a Motoko actor.

DEPLOY YOUR CV WEBSITE ONTO A BLOCKCHAIN

- Set up a cycles wallet
- Obtain free cycles from Dfinity
- Understand how to manage the cycles balance of a canister
- Build and deploy a static, personal CV website to the Internet Computer
- Check how many cycles were used

WORK WITH A REACT FRONTEND AND BLOCKCHAIN BACKEND

- Use a React frontend in combination with the Internet Computer
- Allow users to create their own notes and persist these in a canister
- Learn to use CRUD style operations on a canister from the front end
- Learn to use the Candid interface to test the app.

CREATE YOUR OWN CRYPTO TOKEN

- Understand the role of tokens in dapps and open internet services
- Create a token modelled on Ethereum's ERC-20 and deploy it on the Internet Computer
- Create a faucet contract that dispenses small amounts of the token.
- Understand and work with principal IDs to identify users and smart contracts
- Use the Plug wallet to track token balances
- Learn how to transfer the token between user accounts.

CREATE AN NFT AND MARKETPLACE

- Create a non-fungible token modelled on ERC-721 and deploy it to the Internet Computer
- Build a marketplace website like OpenSea to allow users to mint, buy and sell NFTs.
- The owner of the NFT can interact with the canister to transfer the ownership of the .jpeg to another account

FAQs

1. **What's the difference between your online and in-person bootcamp?**

   The online bootcamp on Brains College has the same curriculum as our in-person bootcamp. Our in-person bootcamp only runs once per year and is wait-list based. The Brains's online course can be taken at anytime, on your schedule.

2. **How long does the bootcamp take to complete?**

   We teach the course in person over 12 weeks. But we have seen students online complete the course as quickly as 3 weeks. It depends if you are going to work full-time or part-time. It can also be helpful to read around the topics and do more self-directed research and experimentation.

3. **I don't have any prior experience or a technical background, can I join the bootcamp?**

   The course is designed for students of all levels. The youngest person we've had on the course is 11, the oldest 86. The most inexperienced student didn't know what a website is, the most experienced works for Twitter as a developer. The most important thing is perseverance and willingness to learn. If you have that then you'll do great.