

COURSE OUTLINE

Topic: HTML, CSS, Bootstrap & JavaScript

Trainer: Joy Kuroko | Platform: Coursera

COURSE INFORMATION

Course Title	Web Design: HTML, CSS, Bootstrap & JavaScript
Course Level	Beginner to Intermediate
Duration	8 Weeks (48 Hours Total)
Format	Live Online Classes + Recorded Sessions + Project Work
Schedule	Mon & Wed: 6:00 PM – 8:00 PM Saturday: 10:00 AM – 12:00 PM
Prerequisites	No prior experience required — just a laptop and internet access
Certification	Certificate of Completion issued upon passing final project assessment

COURSE DESCRIPTION

This comprehensive course is designed to take complete beginners through the full journey of building modern, professional, and responsive websites. Starting from the very basics of HTML structure, learners will progressively master CSS styling, Bootstrap framework for responsive design, and JavaScript for interactivity. By the end of this course, students will have built multiple real-world projects and a personal portfolio website they can proudly showcase to employers or clients.

LEARNING OUTCOMES

By the end of this course, learners will be able to:

- Build well-structured web pages using semantic HTML5 elements
- Style websites professionally using CSS3 including Flexbox and Grid layouts
- Create fully responsive websites using Bootstrap 5 framework
- Add dynamic interactivity to web pages using JavaScript (ES6+)
- Manipulate the DOM to create interactive UI components
- Handle user events, forms, and input validation with JavaScript
- Build and deploy a complete multi-page responsive website
- Apply best practices in web accessibility and performance

WEEKLY CURRICULUM BREAKDOWN

WEEK 1: Introduction to Web Development & HTML Fundamentals		6 Hours
• How the Web Works	Overview of browsers, servers, HTTP, and how websites are delivered to users.	
• Setting Up Your Environment	Installing VS Code, extensions (Live Server, Prettier), and browser DevTools intro.	
• HTML Document Structure	DOCTYPE, html, head, body tags; understanding the anatomy of a web page.	
• Text & Semantic Elements	Headings (h1–h6), paragraphs, spans, strong, em, and semantic HTML5 tags (header, main, footer, section, article, nav).	
• Links & Images	Anchor tags, relative vs absolute paths, image optimization and alt attributes.	
• Lists & Tables	Ordered, unordered, and definition lists; creating structured data tables.	
• HTML Forms	Input types, labels, textarea, select, radio, checkbox, and form submission basics.	
■ Project: Build a personal bio page with navigation, sections, and a contact form.		
■ Tools: VS Code, Chrome DevTools, GitHub (intro)		
WEEK 2: CSS Fundamentals & Styling		6 Hours
• CSS Syntax & Selectors	Tag, class, ID selectors; specificity, inheritance, and the cascade.	
• The Box Model	margin, padding, border, content; how spacing works in CSS.	
• Typography & Colors	Font families, Google Fonts, font-size, line-height, color, and backgrounds.	
• CSS Units	px, em, rem, %, vh/vw — when and why to use each.	
• Positioning & Display	static, relative, absolute, fixed, sticky; block vs inline vs inline-block.	
• Pseudo-classes & Pseudo-elements	:hover, :focus, :nth-child; ::before and ::after for decorative content.	
• CSS Variables & Custom Properties	Defining and reusing design tokens for consistent styling.	
■ Project: Style the Week 1 bio page with custom fonts, colors, spacing, and hover effects.		
■ Tools: Chrome DevTools CSS panel, Google Fonts		
WEEK 3: Advanced CSS — Flexbox & Grid		6 Hours
• CSS Flexbox	flex-direction, justify-content, align-items, flex-wrap, gap; building 1D layouts.	
• Flexbox Practical Patterns	Navbar, card rows, centering elements, and footer layouts using Flexbox.	
• CSS Grid	grid-template-columns/rows, grid-area, auto-fill, minmax; building 2D layouts.	
• Grid Practical Patterns	Image galleries, dashboards, and magazine-style layouts.	
• Responsive Design with Media Queries	@media breakpoints (mobile-first), adapting layouts for phone, tablet, desktop.	
• CSS Transitions & Animations	transition property, @keyframes, animation for smooth UI effects.	
■ Project: Build a fully responsive multi-section landing page (no framework — pure CSS).		
■ Tools: CSS Grid Generator, Flexbox Froggy (practice)		

WEEK 4: Bootstrap 5 — Responsive Framework

6 Hours

- **Bootstrap Introduction** What is Bootstrap, CDN setup, and understanding the utility-first approach.
- **Bootstrap Grid System** Containers, rows, columns; breakpoints (sm, md, lg, xl, xxl).
- **Bootstrap Utilities** Spacing (m-, p-), colors, display, text alignment, and flex utilities.
- **Bootstrap Components** Navbar, cards, buttons, badges, alerts, modals, and accordions.
- **Bootstrap Forms** Form controls, input groups, validation states, and form layouts.
- **Customizing Bootstrap** Overriding Bootstrap with custom CSS; using Bootstrap Icons.
- **Responsive Navigation** Building a collapsible hamburger navbar for mobile devices.

■ **Project: Rebuild the Week 3 landing page using Bootstrap — compare effort and result.**

■ *Tools: [Bootstrap 5 Docs](#), [Bootstrap Icons](#), [CDN](#)*

WEEK 5: JavaScript Fundamentals

6 Hours

- **JS Basics & Syntax** Variables (var, let, const), data types, operators, and comments.
- **Control Flow** if/else, switch statements, ternary operators, and logical operators.
- **Loops** for, while, do-while, for...of, for...in — iterating over data.
- **Functions** Function declarations, expressions, arrow functions, parameters, and return values.
- **Arrays & Objects** Creating, accessing, modifying arrays and objects; array methods (map, filter, reduce, forEach).
- **ES6+ Features** Template literals, destructuring, spread/rest operators, and default parameters.
- **Error Handling** try/catch/finally, throwing custom errors, and debugging with console.

■ **Project: Build a JavaScript quiz app with score tracking and result display.**

■ *Tools: [Browser Console](#), [VS Code Debugger](#)*

WEEK 6: DOM Manipulation & Events

6 Hours

- **The DOM Explained** What is the Document Object Model? How JS interacts with HTML.
- **Selecting & Traversing Elements** getElementById, querySelector, querySelectorAll, parentElement, children.
- **Modifying the DOM** innerHTML, textContent, createElement, appendChild, removeChild, classList.
- **Event Listeners** addEventListener, click, keydown, submit, mouseover events and event objects.
- **Event Delegation** Handling events on dynamically created elements efficiently.
- **Form Handling & Validation** Capturing form input, preventing default, real-time input validation.
- **Local Storage** Saving and retrieving data from the browser's localStorage API.

■ **Project: Build an interactive To-Do List app with add, complete, delete, and local storage persistence.**

■ *Tools: [Browser DevTools](#), [MDN Web Docs](#)*

WEEK 7: JavaScript Advanced Concepts & APIs6 Hours

• Asynchronous JavaScript	Callbacks, Promises, async/await — understanding the event loop.
• Fetch API	Making HTTP GET/POST requests, handling responses with .json(), error handling.
• Working with JSON	Parsing, stringifying, and displaying JSON data from APIs.
• Third-Party APIs	Using a free public API (e.g. OpenWeather, REST Countries) to display live data.
• ES Modules	import/export syntax; organizing code into reusable modules.
• Introduction to Version Control	Git commands: init, add, commit, push, pull; working with GitHub repositories.

■ Project: Build a Weather App that fetches live data from a public API and displays it dynamically.

■ Tools: Fetch API, OpenWeatherMap API, Git, GitHub

WEEK 8: Final Project — Build & Deploy a Complete Website6 Hours

• Project Planning	Wireframing, choosing a project type (portfolio, business site, or landing page).
• Development Sprint	Applying HTML, CSS, Bootstrap, and JavaScript to build the complete site.
• Code Review Session	Peer review and trainer feedback on code quality, structure, and best practices.
• Performance & Accessibility	Image optimization, semantic HTML audit, color contrast, and ARIA labels.
• Deployment	Deploying the project live using Netlify or GitHub Pages — step by step.
• Project Presentation	Each learner presents their project — what they built, challenges faced, and what they learned.
• Career Guidance	Building a portfolio, writing a developer CV, freelancing basics, and next steps (React, backend).

■ FINAL PROJECT: Complete personal portfolio or business website — responsive, interactive, and live on the web.

■ Tools: Netlify, GitHub Pages, Lighthouse (performance audit)

ASSESSMENT & GRADING

Weekly Mini-Projects (7 projects)	35%
Class Participation & Assignments	15%
Mid-Course Quiz (Week 4)	20%
Final Capstone Project (Week 8)	30%
	100%

TOOLS & RESOURCES PROVIDED

- VS Code (free) with recommended extensions list
- All source code and starter files shared via GitHub repository
- Recorded session replays accessible for 3 months
- WhatsApp / Slack community for peer support and Q&A;
- Curated reading list: MDN Web Docs, CSS Tricks, JavaScript.info
- Certificate of Completion upon successful project submission

ABOUT THE TRAINER

Joy Kuroko is a Full-Stack Software Engineer and experienced Web Development Trainer holding an M.Sc in Computer Science from Benson Idahosa University. She has led training programs at Bigt Infotech Solutions, mentoring students through real-world projects in HTML, CSS, JavaScript, React, and Node.js. As a developer at Lions Gate Academy, she continues to build production-grade web and mobile applications while delivering structured, hands-on training that bridges theory and practice.

Email	jkuroko@gmail.com
Phone	+234 907 208 7504
Portfolio	joykuroko-portfolio.netlify.app
GitHub	github.com/MizMelly

Submitted by Joy Kuroko | CertMart Trainer Application | Course: Web Design HTML, CSS, Bootstrap & JavaScript | 2025