# Full Stack Web Application Development Curriculum

## Month 1: Python Fundamentals

### Week 1

- Introduction to Python 🐍: Variables, data types, basic operators.
- Control structures: Conditionals and loops.
- Weekly Project: Building a simple Python calculator.

### Week 2

- Functions: Defining functions, parameters, return statements.
- File Handling 📁: Reading from and writing to files.
- Weekly Project: Creating a program to analyze text files.

### Week 3

- Modules and Libraries 📚: Importing and using external libraries.
- Error Handling: Exception handling, try-except blocks.
- Weekly Project: Building a simple game using external libraries.

### Week 4

- Object-Oriented Programming (OOP) 🧪: Classes, objects, inheritance, and polymorphism.
- Weekly Project: Creating a simple object-oriented program.

## Month 2: Flask Framework

### Week 1

- Introduction to Flask 🌶️: Setting up a Flask project, creating routes.
- Templates and Rendering 🎨: Working with Jinja2 templates.
- Weekly Project: Building a simple Flask application.

### Week 2

- Form Handling with Flask 📝: Processing form data, validating inputs.

- Routing and URL Parameters 🛣️: Dynamic routing and handling URL parameters.
- Weekly Project: Creating a registration and login system using Flask.

**Week 3**

- Database Integration with Flask 🗃️: Connecting Flask with MySQL, executing queries.
- ORM (Object-Relational Mapping) 🗄️: Working with an ORM library such as SQLAlchemy.
- Weekly Project: Building a CRUD (Create, Read, Update, Delete) application.

**Week 4**

- Advanced Flask Concepts 🚀: Blueprinting, error handling, middleware, and request hooks.
- RESTful APIs 🌐: Designing and implementing RESTful APIs using Flask.
- Weekly Project: Developing a RESTful API with Flask.

**Month 3: HTML5 and CSS3**

**Week 1**

- Introduction to HTML5 🌐: Document structure, tags, attributes.
- Working with Text and Links 🔗: Formatting text, creating links.
- Weekly Project: Creating a simple HTML webpage.

**Week 2**

- Working with Images and Multimedia 🖼️: Inserting images, audio, and video elements.
- HTML Forms 📝: Creating forms and form controls.
- Weekly Project: Building a basic contact form using HTML.

**Week 3**

- Introduction to CSS3 🎨: Selectors, properties, and values.

- CSS Box Model 📦 : Margin, padding, border, and positioning.
- Weekly Project: Styling a webpage using CSS.

**Week 4**

- Layout and Flexbox 📏 : Creating responsive layouts using flexbox.
- CSS Transitions and Animations 🎉 : Adding motion effects to elements.
- Weekly Project: Designing a responsive portfolio webpage.

**Month 4: JavaScript and jQuery**

**Week 1**

- Introduction to JavaScript 🌟 : Variables, data types, control structures.
- Document Object Model (DOM) 🏰 : Accessing and manipulating HTML elements.
- Weekly Project: Creating an interactive quiz using JavaScript.

**Week 2**

- JavaScript Functions and Objects 🎯 : Defining functions, working with objects.
- Event Handling 🎉 : Capturing and handling user interactions.
- Weekly Project: Building a simple image slider using JavaScript.

**Week 3**

- Introduction to jQuery 💫 : DOM manipulation and event handling with jQuery.
- jQuery Effects and Animation 🌈 : Creating visual effects and animations.
- Weekly Project: Developing a dynamic photo gallery using jQuery.

**Week 4**

- Asynchronous JavaScript and AJAX ⏳ : Introduction to asynchronous programming and making AJAX requests.
- jQuery Plugins 🧩 : Using popular jQuery plugins for enhanced functionality.
- Weekly Project: Implementing a live search feature with AJAX and jQuery.

## Month 5: Bootstrap 5

### Week 1

- Introduction to Bootstrap 🎨: Setting up Bootstrap in a project, using Bootstrap classes.
- Bootstrap Grid System 📐: Creating responsive layouts with the grid system.
- Weekly Project: Building a responsive landing page using Bootstrap.

### Week 2

- Bootstrap Components 🧩: Navigation bars, buttons, forms, and cards.
- Bootstrap Utilities 🛠️: Helper classes and responsive utilities.
- Weekly Project: Designing a responsive blog using Bootstrap components.

### Week 3

- Customizing Bootstrap 🎨: Overriding default styles and creating custom themes.
- Bootstrap Modals and Alerts 💬: Displaying modals and alert messages.
- Weekly Project: Developing a modal-based image gallery with Bootstrap.

### Week 4

- Bootstrap Carousel and Scrollspy 🎠: Creating image carousels and scrollspy navigation.
- Optimizing Performance ⚡: Minifying CSS and JavaScript files, optimizing assets.
- Weekly Project: Enhancing a web application with Bootstrap components.

## Month 6: Advanced Flask and MySQL

### Week 1

- User Authentication and Authorization 🔒: Implementing registration, login, and access control.
- Security Best Practices 🛡️: Protecting against common security vulnerabilities.
- Weekly Project: Building a secure user authentication system with Flask.

**Week 2**

- Deployment and Hosting 🚀: Deploying Flask applications to a hosting platform.
- Scaling and Performance ⚙️: Caching techniques, database optimizations.
- Weekly Project: Deploying a Flask application to a cloud hosting platform.

**Week 3**

- RESTful APIs with Flask 🌐: Designing and implementing RESTful APIs using Flask.
- API Authentication and Rate Limiting 🔑: Implementing authentication mechanisms and rate limiting.
- Weekly Project: Creating a RESTful API for a mobile application.

**Week 4**

- Advanced Database Management 🗄️: Database migrations, indexing, and optimization.
- Full Stack Project Development 🚀: Bringing together all learned technologies to build a complete web application.
- Weekly Project: Developing a full-stack web application using Flask, Bootstrap, and MySQL.

Throughout the curriculum, I will be allocating time for practical exercises, coding challenges, and weekly projects to apply the concepts learned. Students are encouraged to document their progress 📝, and I will provide regular feedback and guidance to ensure their success. Let's embark on this exciting journey together! 🚀

Tobechi Ohaeri

Course Creator