Asba HAZEM

Phase 1: Foundations

1. Learn Programming Basics

- **Languages**: Start with JavaScript (focus on ES6+ features).
- **Key Concepts**:
    - Variables, Data Types, Operators
    - Functions, Loops, Conditionals
    - Arrays, Objects, and JSON
    - Basic Problem Solving

2. Version Control

- **Tool**: Git and GitHub
- **Topics to Cover**:
    - Setting up repositories
    - Commit, push, pull, and branching
    - Collaborating on projects

3. Command Line Basics

- **Skills**:
    - Navigating directories
    - File manipulation
    - Running scripts

Phase 2: Front-End Development

1. HTML & CSS

- **HTML**:
  - Semantic HTML tags (e.g., `<header>` , `<footer>`)
  - Forms and validation
- **CSS**:
  - Box Model, Flexbox, Grid
  - Media queries for responsiveness
  - Basic animations and transitions
- **Tools**:
  - Chrome DevTools

2. JavaScript for the Web

- DOM Manipulation
- Event Handling
- Fetch API and Promises

3. Modern Front-End Framework

- **Framework**: React
- **Topics to Cover**:
  - Functional Components and Hooks
  - State Management (useState, useReducer, useContext)
  - React Router for navigation
  - Basic testing with React Testing Library

4. CSS Frameworks and Tools

- Bootstrap or TailwindCSS
- CSS-in-JS tools (e.g., Styled-Components)

Phase 3: Back-End Development

1. Basics of Back-End Programming

- **Language**: Node.js with JavaScript or TypeScript
- **Concepts**:
    - HTTP and RESTful APIs
    - Handling requests and responses
    - Middleware (e.g., Express.js)

2. Databases

- **SQL**:
    - Relational Databases (PostgreSQL, MySQL)
    - Writing queries, Joins, Transactions
- **NoSQL**:
    - MongoDB basics

3. Authentication & Security

- Password hashing (e.g., bcrypt)
- JWT Authentication
- Security best practices (e.g., OWASP Top 10)

4. Advanced Topics

- Real-time communication (e.g., WebSockets, Socket.io)
- GraphQL basics
- File uploads and management

Phase 4: DevOps and Deployment

1. Build and Deployment Basics

- Using npm/yarn for package management

- Build tools (e.g., Webpack, Vite)

2. Hosting Platforms

- Deploying with services like Vercel, Netlify, or AWS

3. Containerization

- Learn Docker basics
- Containerizing full-stack applications

Phase 5: Advanced Front-End

1. TypeScript

- Types and Interfaces
- Generics and Enums
- Migrating JavaScript projects to TypeScript

2. State Management

- Redux Toolkit
- Zustand or Recoil (for lighter state management needs)

3. Testing

- End-to-End Testing with Cypress or Playwright

Phase 6: Advanced Back-End

1. Advanced Frameworks

- NestJS (for TypeScript developers)
- Golang for highly performant back-end systems

2. Microservices

- Learn how to build and manage microservices

- Understand message brokers like RabbitMQ or Kafka

3. Advanced Database Concepts

- Indexing and Query Optimization

- Database migrations

Phase 7: Cloud and Scaling

1. Cloud Services

- AWS, Azure, or Google Cloud basics

- Setting up and using S3, EC2, Lambda

2. CI/CD

- Automate testing and deployments using GitHub Actions or Jenkins

3. Monitoring and Logging

- Tools like Prometheus, Grafana, and ELK Stack

Phase 8: Project Building

- Build real-world projects to solidify your learning:

  1. A Blog Platform

  2. An E-commerce Website

  3. A Real-time Chat App

  4. A Task Management System

Recommended Resources

Websites:

- freeCodeCamp

- MDN Web Docs

- W3Schools

Books:

- "Eloquent JavaScript"

- "You Don't Know JS" series

Courses:

- The Odin Project

- Full-Stack Open

Tools:

- CodePen, Replit, or GitHub Codespaces for practice