100 Days of Code Challenge –Full Stack Web Developer--

Days 1-10: Front-End Fundamentals

Day 1-3: HTML Basics

Learn about HTML structure, tags, and elements.

Build simple web pages with headings, paragraphs, lists, and links.

Day 4-6: CSS Basics

Understand CSS selectors, properties, and values.

Style your HTML pages with colors, fonts, margins, and padding.

Day 7-10: Intermediate CSS

Learn about layout techniques, positioning, and responsive design.

Build a responsive webpage using media queries.

Days 11-20: JavaScript Basics

Day 11-14: JavaScript Fundamentals

Learn variables, data types, operators, and control structures.

Practice writing basic scripts to manipulate the DOM.

Day 15-18: Functions and Objects

Understand functions, scope, closures, and the this keyword.

Learn about objects, methods, and prototypes.

Day 19-20: DOM Manipulation

Explore how to select and modify elements on a webpage using JavaScript.

Build interactive elements like buttons and forms.

Days 21-30: Advanced JavaScript & React Basics

Day 21-24: ES6+ Features

Dive into modern JavaScript features like arrow functions, destructuring, and classes.

Update your existing JavaScript code to use ES6+ syntax.

Day 25-28: Introduction to React

Learn about React components, JSX, and rendering.

Build a simple React application with components and props.

Day 29-30: React State and Events

Understand component state and how to handle user events.

Implement state changes and interactivity in your React app.

Days 31-40: React and Front-End Development

Day 31-34: React Hooks

Explore useState, useEffect, and other hooks for managing state and side effects.

Refactor your React app to use hooks.

Day 35-38: React Routing

Learn about React Router and implement navigation in your app.

Create different routes for different components.

Day 39-40: Styling in React

Explore CSS-in-JS libraries like styled-components.

Style your React components using these libraries.

Days 41-50: Back-End Development Basics

Day 41-44: Introduction to Node.js

Learn about Node.js, its asynchronous nature, and the event loop.

Create a basic server using Node.js.

Day 45-48: Express.js Framework

Understand the basics of Express.js, routing, and middleware.

Build API endpoints to handle HTTP requests.

Day 49-50: Integrating Front-End and Back-End

Connect your React front-end to your Express.js back-end.

Implement API calls from the React app to fetch data from the server.

Days 51-60: Databases and Advanced Front-End

Day 51-54: MongoDB Basics

Learn about MongoDB, collections, documents, and CRUD operations.

Set up a connection between your Express.js server and a MongoDB database.

Day 55-58: Advanced React Concepts

Explore more complex React patterns like context, higher-order components, and render props.

Apply these patterns to enhance your React application.

Day 59-60: State Management with Redux (Optional)

If you choose, learn about Redux for managing complex application state.

Implement Redux to manage global state in your React app.

Days 61-70: Authentication and User Management

Day 61-64: User Authentication

Learn about authentication methods like JWT (JSON Web Tokens).

Implement user registration and login functionality in your app.

Day 65-68: Protected Routes

Create protected routes that require authentication to access.

Implement access control for different user roles.

Day 69-70: User Profiles and Settings

Allow users to update their profiles and account settings.

Implement password reset functionality.

Days 71-80: RESTful APIs and Deployment

Day 71-74: Advanced API Design

Dive deeper into RESTful API design principles.

Implement pagination, sorting, and filtering in your API.

Day 75-78: Deployment Basics

Explore deployment options like Heroku, Netlify, and VPS.

Deploy your front-end and back-end applications.

Day 79-80: Error Handling and Validation

Implement error handling mechanisms in your back-end.

Validate user input on both the front-end and back-end.

Days 81-90: Real-Time Communication and Final Projects

Day 81-84: WebSocket Communication

Learn about real-time communication using technologies like WebSockets.

Implement a real-time chat feature in your application.

Day 85-88: Performance Optimization

Explore techniques to optimize your application's performance.

Minimize load times and enhance user experience.

Day 89-90: Finalizing Projects

Spend time polishing and refining your projects.

Implement any remaining features or improvements.

Days 91-100: Portfolio and Beyond

Day 91-94: Create a Portfolio

Build a personal portfolio website showcasing your projects and skills.

Deploy your portfolio and link to your GitHub repositories.

Day 95-98: Continuous Learning

Stay updated with the latest trends in web development.

Explore additional technologies like GraphQL, Docker, etc.

Day 99-100: Reflection and Future Goals

Reflect on your journey and how much you've learned.

Set goals for your future as a Full Stack Web Developer.

PYTHON Interview questions.......

Basic Python Concepts:

- What is Python? What are its key features?
- Explain the difference between a list and a tuple in Python.
- How do you comment out multiple lines of code in Python?
- What is the purpose of the `__init__` method in a Python class?
- How are exceptions handled in Python? Explain the try-except block.
- What is a virtual environment in Python and why is it used?

Data Types and Operators:

- Discuss the differences between mutable and immutable objects in Python.
- Explain the differences between `==` and `is` operators in Python.
- How does garbage collection work in Python?
- Describe the purpose and usage of the `enumerate()` function.
- What is list comprehension? Provide an example.

Object-Oriented Programming (OOP):

- Define encapsulation, inheritance, and polymorphism in the context of OOP.
- How do you create a new instance of a class in Python?
- Explain the difference between instance variables and class variables.
- What is method overloading? Does Python support it?
- Describe the concept of method overriding with an example.
- What is the difference between `__str__()` and `__repr__()` methods?
- How does multiple inheritance work in Python? Mention any issues associated with it.

Functions and Modules:

- Explain the purpose of the `*args` and `**kwargs` in function definitions.
- What is a lambda function? Provide an example of its usage.
- How are modules used in Python? How do you import a module?
- What is the purpose of the `if __name__ == "__main__":` block in a Python script?

File Handling:

- Explain the difference between reading a file using `read()`, `readline()`, and `readlines()`.
- How do you open and write to a file in Python?
- What is the purpose of the `with` statement in file handling?

Advanced Concepts:

- Describe the Global Interpreter Lock (GIL) in Python and its impact on multithreading.
- What are decorators in Python? Provide an example of how they are used.
- Explain the concept of generators and the use of the `yield` keyword.
- How can you handle and raise custom exceptions in Python?
- What is a closure in Python? Provide an example.
- Discuss the differences between list comprehension and generator expression.

NAEEM

Python Developer

📞 +91-9598489890   @ nn9412466@gmail.com   📍 LIG Colony, Kurla West , Mumbai 400070

OBJECTIVE

Energetic and motivated fresher with a strong passion for Python programming and a desire to contribute to innovative projects. Seeking an opportunity to apply my coding skills and learn from experienced developers.

EDUCATION

Bachelor of Science (B.Sc)

Allahabad University Family

📅 2021 - 2023

SKILLS

Programming Languages: Python

Web Development: HTML, CSS

Version Control: Git

Problem Solving and Logical Thinking

Microsoft word   Microsoft excels.   Typing Speed: 40-45wpm

PROJECTS

1. Snake, Water & Gun Game Program.

2. Kon Banega Karorpati (KBC) Program.

3. Python Calculator Program.

4. Library Management System Program (OOP).

5. Merge The PDF Program.

6. Random Password Generator.

7. News API in Python program.

LANGUAGES

Hindi
Native   ●●●●●

NAEEM

Python Developer

📞 +91-9598489890   @ nn9412466@gmail.com   📍 LIG Colony, Kurla West , Mumbai 400070

OBJECTIVE

Energetic and motivated fresher with a strong passion for Python programming and a desire to contribute to innovative projects. Seeking an opportunity to apply my coding skills and learn from experienced developers.

EDUCATION

Bachelor of Science (B.Sc)

Allahabad University Family

📅 2021 - 2023

SKILLS

Programming Languages: Python

Web Development: HTML, CSS

Version Control: Git   **Problem Solving and Logical Thinking**

Microsoft word   **Microsoft excels.**   **Typing Speed: 40-45wpm**

LANGUAGES

Hindi
Native   ●●●●●

English
Intermediate   ●●○○○

PROJECTS

1. Snake, Water & Gun Game Program.

2. Kon Banega Karorpati (KBC) Program.

3. Python Calculator Program.

4. Library Management System Program (OOP).

5. Merge The PDF Program.

6. Random Password Generator.

7. News API in Python program.

NAEEM

Python Developer

📞 +91-9598489890   @ nn9412466@gmail.com
🔗 www.linkedin.com/in/naeem-s-9032b5289   📍 LIG Colony, Kurla West , Mumbai 400070

OBJECTIVE

Energetic and motivated fresher with a strong passion for Python programming and a desire to contribute to innovative projects. Seeking an opportunity to apply my coding skills and learn from experienced developers.

EDUCATION

Bachelor of Science (B.Sc)
Allahabad University Family
📅 2021 - 2023

FIND ME ONLINE

in   **LinkedIn**
www.linkedin.com/in/naeem-s-9032b5289

SKILLS

| Programming Languages: Python |

| Web Development: HTML, CSS |

| Version Control: Git | Problem Solving and Logical Thinking |

| Microsoft word | Microsoft excels. | Typing Speed: 40-45wpm |

LANGUAGES

Hindi
Native   ●●●●●

English
Intermediate   ●●○○○

PROJECTS

1. Snake, Water & Gun Game Program.

2. Kon Banega Karorpati (KBC) Program.

3. News API in Python program

4. Library Management System Program (OOP).

5. Merge The PDF Program.

6. Random Password Generator.

Naeem

Python Developer

📞 **+91-9598489890**   @ **nn9412466@gmail.com**

📍 **Mumbai, Kurla West , LIG Colony 400070**

SUMMARY

Energetic and motivated fresher with a strong passion for Python programming and a desire to contribute to innovative projects. Seeking an opportunity to apply my coding skills and learn from experienced developers.

EDUCATION

Bachelor of Science

Allahabad University Famiy

📅 2023

SKILLS

| Programming Languages: PYTHON |

| Web Development: HTML, CSS |

| Version Control: Git |

| Problem Solving and Logical Thinking |

| Typing Speed: 35-40wpm |

PROJECTS

Kon Banega Karorpati (KBC)

- What was a successful outcome of your work? (e.g. Raised $3,000 for the charity)

Python Calculator

- Created a command-line calculator application in Python.
- Implemented arithmetic operations and error handling to ensure accurate calculations.
- Technologies: Python

NAEEM

Python Developer

📞 **+91-9598489890** @ **nn9412466@gmail.com** 🔗 **www.github.com/9598Naimshaikh**
🔗 **www.linkedin.com/in/naeem-s-9032b5289** 📍 **LIG Colony, Kurla West , Mumbai 400070**

OBJECTIVE

Energetic and motivated fresher with a strong passion for Python programming and a desire to contribute to innovative projects. Seeking an opportunity to apply my coding skills and learn from experienced developers.

EDUCATION

Bachelor of Science (B.Sc)

Allahabad University Family

📅 2020 - 2023

FIND ME ONLINE

in **LinkedIn**
www.linkedin.com/in/naeem-s-9032b5289

SKILLS

Programming Languages: Python

Web Development: HTML, CSS

Version Control: Git & GitHub **Problem Solving and Logical Thinking** **Microsoft word**

Microsoft excels. **Typing Speed: 40-45wpm**

LANGUAGES

Hindi
Native ●●●●●

English
Intermediate ●●○○○

PROJECTS

1. Snake, Water & Gun Game Program.

2. Kon Banega Karorpati (KBC) Program.

3. News API in Python program

4. Library Management System Program (OOP).

5. Merge The PDF Program.

6. Random Password Generator.