**100DaysOfCodeALXSE**

**My schedule for 100 Days**

**Day 1-10: Python Basics**

* **Days 1-3**: Install Python and set up your development environment. Learn about basic syntax, variables, and data types in Python.
* **Days 4-6**: Dive deeper into control flow statements like if, elif, and else. Practice using loops such as for and while loops.
* **Days 7-8**: Explore Python's built-in functions and standard library modules for tasks like string manipulation, math operations, and working with dates and times.
* **Days 9-10**: Familiarize yourself with data structures in Python including lists, tuples, dictionaries, and sets.

**Day 11-30: Intermediate Python**

* **Days 11-15**: Learn about functions in Python, including defining and calling functions, passing arguments, and returning values. Practice writing reusable and modular code.
* **Days 16-20**: Deepen your understanding of object-oriented programming (OOP) concepts such as classes, objects, inheritance, and polymorphism.
* **Days 21-25**: Explore file handling in Python, including reading from and writing to files, working with CSV and JSON files, and error handling with try-except blocks.
* **Days 26-30**: Start working on small Python projects to apply what you've learned so far. Projects could include building a simple calculator, a todo list app, or a text-based game.

**Day 31-70: Advanced Python and Projects**

* **Days 31-40**: Dive deeper into more advanced topics such as regular expressions, working with APIs, and asynchronous programming.
* **Days 41-50**: Explore web development with Python using frameworks like Flask or Django. Learn about routing, templates, and interacting with databases (e.g., SQLite, PostgreSQL).
* **Days 51-60**: Learn about data manipulation and analysis with libraries like NumPy and Pandas. Practice working with datasets, cleaning data, and performing basic data analysis.
* **Days 61-70**: Work on medium-sized projects that challenge your skills and creativity. Projects could include building a blog, a weather app, or a basic machine learning model using libraries like scikit-learn.

**Day 71-100: Specialization and Final Projects**

* **Days 71-80**: Choose a specialization within Python that interests you. This could be web development, data science, machine learning, automation, etc. Dive deep into related topics and projects.
* **Days 81-90**: Refine and expand your knowledge in your chosen specialization. Work on projects that showcase your skills and solve real-world problems.
* **Days 91-100**: Work on a larger, more complex project that integrates multiple Python libraries and technologies. This could be a personal portfolio website, a full-stack web application, or a machine learning project with real-world applications.

**Tips:**

* **Practice Daily**: Make sure to code every day, even if it's just for a short period of time.
* **Utilize Resources**: Take advantage of online tutorials, documentation, and community forums like Stack Overflow and Reddit.
* **Build Projects**: Hands-on projects are crucial for solidifying your skills and gaining practical experience.
* **Review and Refactor**: Regularly review and refactor your code to improve readability, efficiency, and maintainability.
* **Stay Curious**: Python has a vast ecosystem, so don't hesitate to explore new libraries, frameworks, and tools that catch your interest.