কমপ্লিট পাইথন রোডম্যাপ

Rashedul Alam Shakil

Works at Siemens, Germany
Founder of aiQuest Intelligence
Founder of Study Mart
Master in Data Science at FAU Erlangen

Watch Video

❖ Basic Syntax and Structure

❖ Control Flow

❖ Functions

❖ Data Structures & Algorithms

❖ Modules and Packages

❖ File Handling

❖ Error Handling

❖ Object-Oriented Programming (OOP)

❖ Working with Data

❖ Advanced Concepts

❖ Libraries and Frameworks

❖ Testing and Debugging

❖ Version Control

❖ Best Practices

1. Basic Syntax and Structure:
- ❖ Variables and Data Types
- ❖ Operators (arithmetic, comparison, logical, etc.)
- ❖ Comments and Docstrings

2. Control Flow:
- ❖ Conditional Statements (if, elif, else)
- ❖ Loops (for, while)
- ❖ Loop Control (break, continue, pass)

3. Functions:
- ❖ Defining and Calling Functions
- ❖ Arguments and Parameters
- ❖ Return Values
- ❖ Lambda Functions
- ❖ Scope (local, global)

4. Data Structures & Algorithm:

- ❖ Lists
- ❖ Tuples
- ❖ Dictionaries
- ❖ Sets
- ❖ List Comprehensions
- ❖ Dictionary Comprehensions

- ❖ Time Complexity
- ❖ Space Complexity
- ❖ Sorting algorithms
- ❖ Linked list
- ❖ Stack
- ❖ Queue

5. Modules and Packages:

- ❖ Importing Modules
- ❖ Standard Library Modules
- ❖ Installing and Using Third-Party Packages (pip)

6. File Handling:

- ❖ Reading and Writing Files
- ❖ Working with File Paths

7. Error Handling:
- ❖ Exceptions (try, except, finally)
- ❖ Raising Exceptions
- ❖ Custom Exceptions

8. Object-Oriented Programming (OOP):
- ❖ Classes and Objects
- ❖ Methods and Attributes
- ❖ Inheritance
- ❖ Polymorphism
- ❖ Encapsulation

9. Working with Data:
- ❖ Strings (manipulation and formatting)
- ❖ Regular Expressions
- ❖ Dates and Times

10. Advanced Concepts:
- ❖ Decorators
- ❖ Generators and Iterators
- ❖ Context Managers
- ❖ Metaclasses

11. Libraries and Frameworks:
- ❖ Common Libraries (e.g., NumPy, pandas, matplotlib)
- ❖ Web Development (Flask, Django)
- ❖ Web Scraping (BeautifulSoup, Selenium, Scrapy)
- ❖ Data Science and Machine Learning (scikit-learn, TensorFlow, PyTorch)

12. Testing and Debugging:
- ❖ Writing Tests (unittest, pytest)
- ❖ Debugging Tools and Techniques

13. Version Control:
- ❖ Using Git for Version Control

14. Best Practices:
- ❖ Code Style (PEP 8)
- ❖ Writing Clean and Readable Code
- ❖ Documentation

Learning Resources Links:

- ➢ Official Docs: [Visit](#)
- ➢ Become a Python Developer: [Visit](#)
- ➢ 60 Days of Python: [Visit](#)