Python Topics Agenda

Basic Python Topics:

- Introduction to Python:
  - Python syntax, keywords, and structure
  - Installation and environment setup (IDEs, virtual environments)
- Data Types and Variables:
  - Numbers, Strings, Booleans, Lists, Tuples, Sets, Dictionaries
  - Type conversions
- Basic Operators:
  - Arithmetic, Comparison, Logical, Assignment, Bitwise
- Control Flow:
  - If-else statements
  - While and For loops
  - Break, Continue, Pass
- Functions:
  - Defining and calling functions
  - Function parameters and return values
  - Default arguments and variable-length arguments (*args, **kwargs)
- Basic Data Input/Output:
  - Input from the user
  - Reading from and writing to files
- Error Handling:
  - Try-except blocks
  - Common exceptions

Intermediate Python Topics:

- Object-Oriented Programming (OOP):
  - Classes and Objects
  - Inheritance, Encapsulation, Polymorphism
  - Dunder (Magic) Methods (e.g., __init__, __str__)
- Modules and Packages:
  - Importing modules
  - Creating your own modules
  - Using pip to install packages
- Working with Files:
  - Context Managers (with statement)
  - Reading and writing different file formats (CSV, JSON)

- Decorators and Generators:
  - Understanding function decorators
  - Yield and generators for efficient looping
- Comprehensions:
  - List, dictionary, and set comprehensions
- Regular Expressions (Regex):
  - Pattern matching and substitution using re module
- Lambda Functions and Functional Programming:
  - Map, Filter, and Reduce functions
  - Anonymous functions (lambda)
- Error Handling (Advanced):
  - Custom exceptions
  - Raising exceptions
- Unit Testing:
  - Introduction to testing in Python (unittest, pytest)
  - Writing test cases

Advanced Python Topics:

- Advanced OOP Concepts:
  - Metaclasses
  - Multiple inheritance and Method Resolution Order (MRO)
  - Property decorators (@property)
- Concurrency and Parallelism:
  - Multithreading and Multiprocessing
  - Asyncio for asynchronous programming
- Working with Databases:
  - Connecting to SQL databases using sqlite3, SQLAlchemy
  - CRUD operations with databases
- Data Science and Machine Learning Libraries:
  - NumPy, Pandas for data manipulation
  - Matplotlib, Seaborn for data visualization
  - Introduction to machine learning with Scikit-learn
- Web Development Frameworks:
  - Introduction to Flask and Django
  - Building REST APIs
- Handling Large Files and Data:
  - Working with CSVs, JSON, and XML in memory-efficient ways
  - Streaming data processing
- Advanced Modules:
  - Logging and Debugging
  - Contextlib for advanced resource management
  - Threading and Queues for task management

- Packaging and Deployment:
  - Creating Python packages and distributing them
  - Version control and CI/CD integration for Python projects
- Working with APIs:
  - Sending requests with the requests library
  - Consuming and building APIs (REST, GraphQL)
- Performance Optimization:
  - Profiling Python code
  - Using Cython or PyPy for performance boosts