Python AI

S. No.	Topic	Subtopic	Learning Outcome
1	**Introduction to Programming**	- Basics of Programming	- Understand fundamental programming concepts.
		- Programming Languages Overview	- Identify different programming languages and their use cases.
		- Introduction to Python	- Gain familiarity with Python syntax and structure.
2	**Python Programming Basics**	- Variables and Data Types	- Learn to declare variables and use different data types.
		- Operators	- Perform operations using arithmetic, comparison, logical, and bitwise operators.
		- Control Structures	- Implement decision-making using if-else, loops, and control flow structures.
		- Functions	- Understand function creation, arguments, return values, and scope.
3	**Data Structures in Python**	- Lists, Tuples, and Sets	- Master data manipulation with lists, tuples, and sets.
		- Dictionaries	- Use dictionaries for key-value pair operations.
		- Working with Strings	- Perform string operations and manipulations.
4	**Object-Oriented Programming**	- Classes and Objects	- Understand the principles of Object-Oriented Programming (OOP).
		- Inheritance, Polymorphism, Encapsulation	- Implement inheritance, polymorphism, and encapsulation in Python.
		- Exception Handling	- Manage errors and exceptions effectively.
5	**Python Libraries**	- NumPy	- Perform mathematical and statistical operations using NumPy.
		- Pandas	- Handle data manipulation and analysis with Pandas.
		- Matplotlib and Seaborn	- Create data visualizations using Matplotlib and Seaborn.
6	**Introduction to AI**	- What is AI?	- Understand the basics and history of Artificial Intelligence.
		- Machine Learning vs. AI	- Differentiate between Machine Learning and Artificial Intelligence.
		- AI Applications	- Explore real-world applications of AI in various industries.
7	**Machine Learning Basics**	- Supervised Learning	- Learn the concept of supervised learning and implement algorithms like Linear Regression and SVM.
		- Unsupervised Learning	- Explore unsupervised learning techniques like clustering and dimensionality reduction.
		- Neural Networks	- Introduction to neural networks and their role in AI.
8	**AI Tools & Frameworks**	- TensorFlow	- Gain hands-on experience with TensorFlow for building AI models.
		- Keras	- Learn to use Keras for building and training neural networks.
		- Scikit-learn	- Implement machine learning algorithms using Scikit-learn.
9	**Security Basics for Engineers**	- Importance of Cybersecurity	- Understand the fundamentals of cybersecurity and its importance in software development.
		- Threats and Vulnerabilities	- Identify common security threats and vulnerabilities.
		- Secure Coding Practices	- Learn best practices for writing secure code in Python.
10	**Security Implementation**	- Authentication and Authorization	- Implement secure authentication and authorization mechanisms in applications.
11	**Project Development**	- Project Planning	- Plan and design a software development project.
		- Coding and Implementation	- Apply the learned skills to develop a real-world application.
		- Testing and Debugging	- Test and debug the application to ensure its functionality and security.
12	**Final Project Presentation**	- Project Report	- Prepare a comprehensive report detailing the project development process and outcomes.
		- Presentation Skills	- Present the final project effectively, showcasing coding, AI, and security skills.