



Python Frameworks: A Comprehensive Introduction

This presentation explores Python's rich framework ecosystem. We'll cover web, data science, and asynchronous frameworks. Understand their strengths and choose the best for your needs.

Name : Krupa Bhavsar

Enrollment : 2402031030093

Django: The High-Level Web Framework

Key Features

- ORM
- Templating Engine
- Security Features

Django is a robust, batteries-included framework. It accelerates web development with its comprehensive tools.

Use Cases

- Instagram
- Mozilla
- Pinterest

Popular for complex, database-driven websites. Django emphasizes reusability and rapid development.

Flask: The Microframework

Lightweight

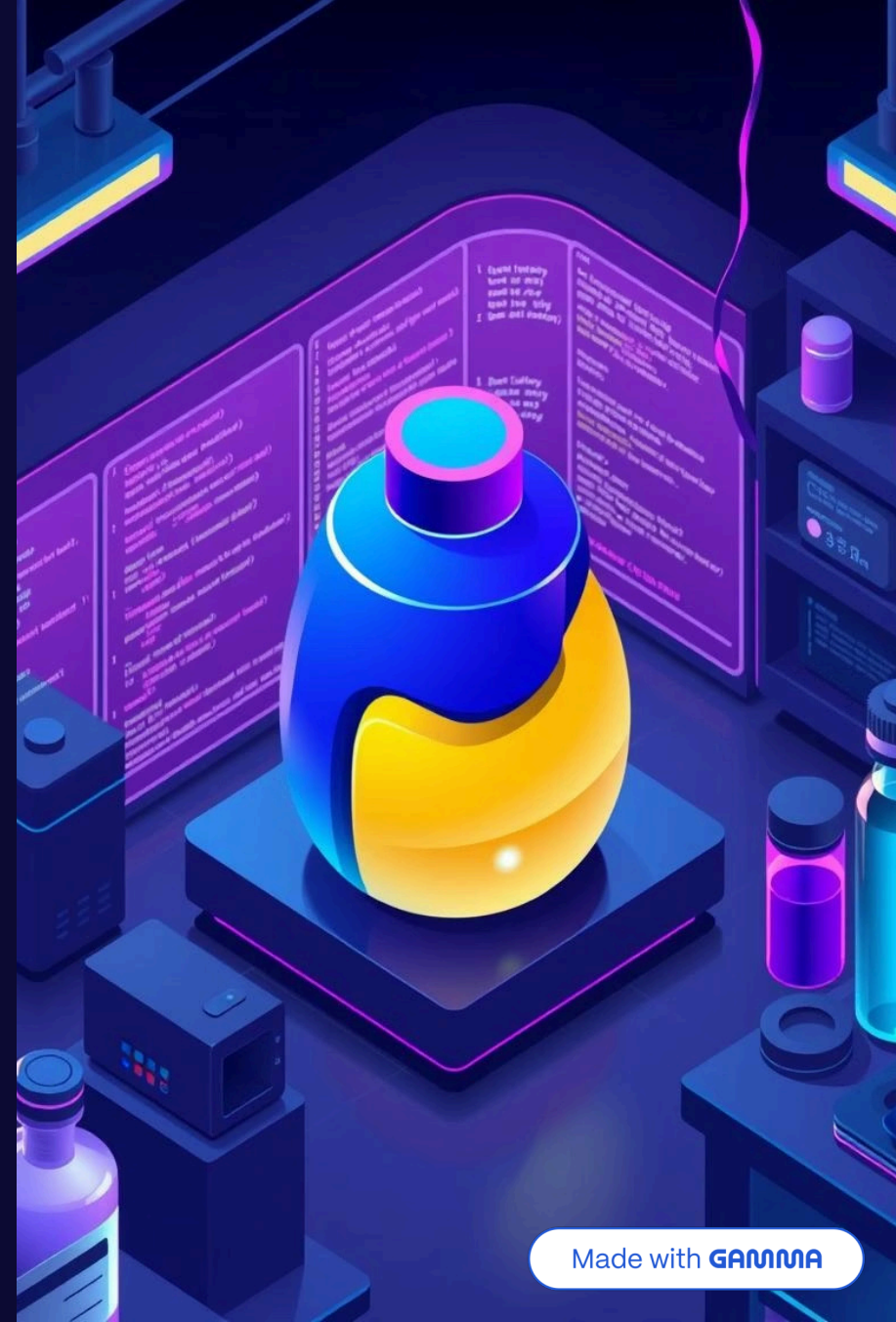
Flask is a microframework offering simplicity and flexibility. It is ideal for smaller applications.

Extensible

Highly customizable, allowing developers to choose components. Flask offers WSGI toolkit and Jinja2 templating.

Use Cases

Perfect for APIs, microservices, and medium-sized apps. Examples include Netflix and Reddit.





Pyramid: The Adaptable Framework

Focus on Quality

Pyramid prioritizes adaptability and testability. It suits both simple and complex projects.

Minimal Boilerplate

Start small and finish big, with extensive configuration options.

Flexible

A general-purpose framework. It is known for minimal boilerplate and quality.

NumPy: Numerical Computing Foundation



Arrays

NumPy provides multi-dimensional array objects. These are essential for numerical operations.



Functions

Extensive mathematical functions are available.



Optimized

Optimized C code ensures high performance. NumPy is the base for pandas and scikit-learn.





Pandas: Data Analysis Powerhouse



DataFrames

Pandas introduces DataFrames and Series. These facilitate data manipulation.



Cleaning

Data cleaning and wrangling become easier.



Integrate

Integration with Matplotlib and Seaborn is seamless. Visualize your insights effortlessly.

Scikit-learn: Machine Learning Toolkit



Classification

Scikit-learn offers various classification algorithms.



Regression

It supports regression for predictive modeling.



Clustering

Clustering algorithms help in data segmentation.

asyncio: Concurrent Programming

Event Loop

asyncio features an event loop for managing tasks.



Coroutines

It introduces coroutines for efficient concurrency.

Asynchronous I/O

Asynchronous I/O operations improve performance.

Tornado: Asynchronous Networking



Non-blocking I/O

Tornado excels with non-blocking I/O.



WebSocket Support

It supports WebSockets for real-time communication.

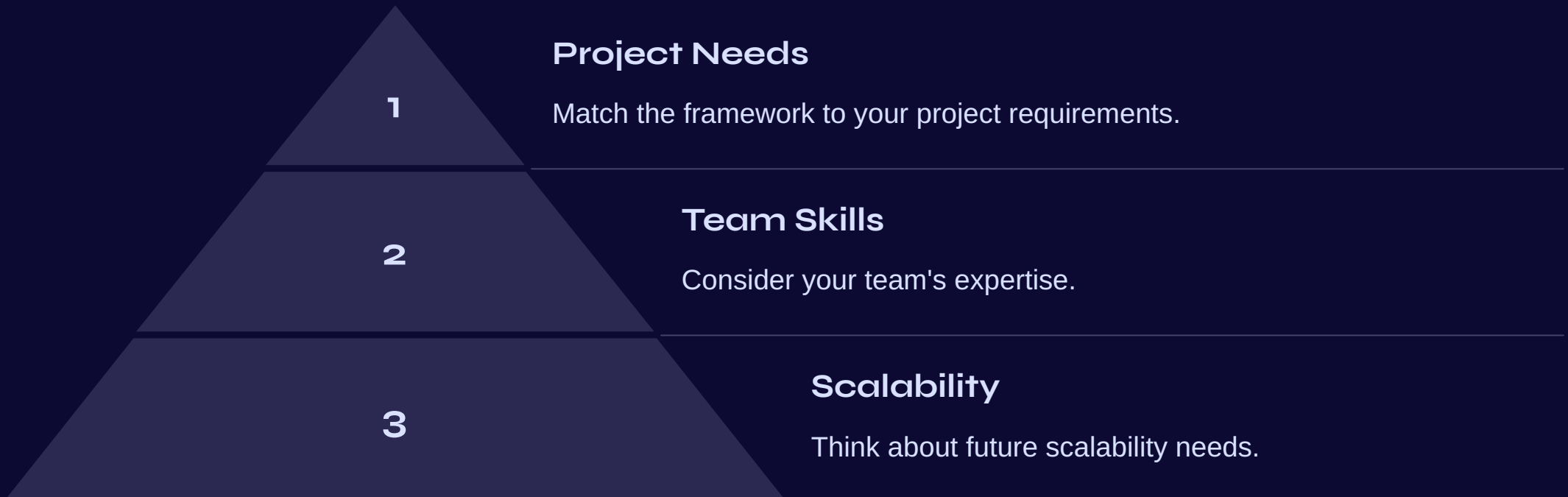


Scalability

Handle 10,000+ open connections easily.



Choosing Your Framework Wisely



Refereance

GitHub Repository : <https://github.com/Krupal989/2402031030029-Python>

Thank You
Any Questions?