NAME : JAINAM SOLANKI

ENROLLMENT NO : 230203100010 4

SUBJECT : PROGRAMMING IN PYTHON

CLASS : 4_IT_B

TOPIC : INTRODUCTION OF DIFFERENT FRAMEWORKS IN PYTHON

# TABLE OF CONTENT



# INTRODUCTION

A framework in Python is a collection of pre-written code that provides a structure and common functionality for building applications. It acts as a foundation or skeleton that developers can build upon, helping them write applications more efficiently and with fewer errors.

# WEB DEVELOPMENT

- Django: A high-level web framework that promotes rapid development and clean, pragmatic design.
- Flask: A lightweight framework with a modular design, ideal for smaller applications or APIs.
- FastAPI: Known for its speed, FastAPI is great for building APIs with Python 3.7+ syntax and type hints.

# DATA SCIENCE AND MACHINE LEARNING

- NumPy: Fundamental package for numerical computing with Python, essential for array operations.
- Pandas: Data manipulation and analysis library, offering data structures like DataFrame.
- TensorFlow & PyTorch: Deep learning frameworks that facilitate building and training neural networks.

# GUI APPLICATIONS

- NumPy: Fundamental package for numerical computing with Python, essential for array operations.
- Pandas: Data manipulation and analysis library, offering data structures like DataFrame.
- TensorFlow & PyTorch: Deep learning frameworks that facilitate building and training neural networks.

# GAME DEVELOPMENT

- Pygame: A set of Python modules designed for writing video games.
- Pyglet: A cross-platform windowing and multimedia library for Python.

# ASYNCIO AND NETWORKING

- Asyncio: Python's standard library for writing concurrent code using the async/await syntax.
- Twisted: An event-driven networking engine written in Python, useful for implementing networked applications

# TESTING

- pytest: A popular framework for writing simple to complex functional tests.
- unittest: Python's built-in framework for organizing test cases into suites and running them.

# KEY FEATURES OF A PYTHON FRAMEWORK

Predefined structure (folders, files, classes, methods).

Predefined structure (folders, files, classes, methods).

Tools and libraries to handle common tasks (e.g., routing, database handling, form validation).

Consistency in coding standards and project layout.

# WHY USE A FRAMEWORK?

Saves time and effort – less code to write.

Improves productivity – built-in tools and features.

Promotes best practices – encourages clean, maintainable code.

Secure and scalable – often includes protection against common issues like SQL injection, CSRF, etc

# SUMMARY

A framework in Python is a set of tools, libraries, and conventions that helps developers build applications efficiently. It provides a predefined structure and reusable components so you don’t have to start from scratch.

# GITHUB LINK

# THANK YOU

123-456-7890 hello@reallygreatsite.com