PART 1 Theoretical Understanding

Q1: Explain the primary differences between TensorFlow and PyTorch. When would you choose one over the other?

1. TensorFlow is developed by Google and emphasizes deployment at scale (e.g., TensorFlow Lite, TensorFlow Serving), while PyTorch, developed by Facebook, is more popular for research due to its dynamic computation graph (eager execution).
2. Choose PyTorch for quick prototyping and research (more Pythonic and intuitive).
3. Choose TensorFlow for production-ready models and deployment on mobile, web, or large-scale cloud systems.

Q2: Describe two use cases for Jupyter Notebooks in AI development.

1. Experimentation & Prototyping: Jupyter Notebooks allow step-by-step testing of AI models, which is ideal for trying out preprocessing steps, tuning hyperparameters, or visualizing data.
2. Educational Demos & Tutorials: Instructors and students use Jupyter to explain concepts interactively, combining code, visualizations, and explanations in one place.

Q3: How does spaCy enhance NLP tasks compared to basic Python string operations?

1. spaCy provides advanced linguistic features such as tokenization, part-of-speech tagging, named entity recognition, and dependency parsing, which are not possible with basic string operations like .split() or .replace().

1. It uses optimized pipelines and pre-trained models for faster, more accurate NLP analysis compared to manual methods.

🔻 2. Comparative Analysis: Scikit-learn vs TensorFlow

Feature	Scikit-learn	TensorFlow
Target Applications	Classical ML (e.g., regression, SVM, KNN)	Deep Learning (e.g., CNNs, RNNs, transformers)
Ease of Use	Beginner-friendly, simple API	Steeper learning curve, especially for custom models
Community Support	Large and mature community; great for ML	Very active and growing community; great for deep learning and deployment