MASTERING THE AI TOOLKIT

- Primary difference between Tensor Flow and PyTorch
    - Tensor Flow uses a static computation graph which requires you to define the entire model before running it, making it better for deployment and production environments. (choose it for large-scale, production-ready systems)
    - PyTorch uses dynamic computation graph which builds the graph as code executes making it easier to debug and experiment with. (choose it for research rapid prototyping and flexible model design)
- Two uses of Jupyter Notebooks in AI development
    - Interactive Experimentation: Data scientists can write, test and visualize AI code in real time making it easier to debug and analyze result
    - Documentation and Reporting: Jupyter Notebook combine code, outputs and explanations in one document, useful for tutorials, research reports or presentation.
- How SpaCy enhance NLP tasks compared to basic Python string Operations
    - ✓ SpaCy provides pre-trained language models that understand models that understand grammar, entities and dependencies enabling advanced NLP tasks like tokenization, POS tagging and named entity recognition.
    - ✓ Basic Python string operations (like .split () or .replace ()) only handle plain text and lack linguistic understanding.
    - ✓ Thus, spaCy offers faster, more accurate and scalable text processing for NLP applications.

COMPARATIVE ANALYSIS

Compare Scickit-Learn and Tensor Flow in terms of:

Aspect	Scikit-Learn	Tensor Flow
Target Applications	Classical machine learning(e.g. regression,classification,clustering)	Deep learning and neural networks
Ease use for Beginners	Very easy to learn simple APIs	Steeper learning curve, more complex setup
Community Support	Large and active for traditional ML	Large and growing especially in AI/deep learning fields