Python Vs Java and Compilers

Friday, April 25, 2025 1:59 PM

Python Vs Java
Python Compilers
Installation python Training in QA
Basics

1.0 → Cpython

Aspect	Java	Python
Type	Compiled (JIT compiler)	Interpreted (bytecode + PVM)
Speed	faster due to JIT compiler	slower due to interpretation
Execution model	JVM	PVM
Typing	bytecode → MC int a, char b Static Type	a = 10 a = [ ] b = " " → dynamic type

Python vs Java

What is the Problem?

Python → slower (interpreted + length) → PVM → bytecode execute line by line

How is rectified? Python 3.x

Pypy → compiler → JIT compiled version
→ interprets code at first (Cpython)
→ Traces which code is executed
→ compiles the hot code into m
code at Runtime (JIT)
→ Run optimized machine code much

Pypy can run 4x to 10x faster than for certain Programs

Virtual environment in Python

.exe → install → folder (C:\, D:\)

Python, Pandas, matplotlib → stored diffr location Globally Python 3.1

2 Projects
AI → flask, pandas, scikit → Project 1 (Python 3.11) → Venv 1
Projects 2 (Python 3.13) → Pandas, Python, Pytest ← ETL Testing
Venv 2
→ to avoid conflicts b/w Packages diff

Why use venv

1) keep your Project dependencies isolated
2) Avoid version conflicts

3) makes Project more portable
4) ensures reproducibility