| **Feature** | **Python (Pandas/Native)** | **PySpark (Spark API in Python)** |
| --- | --- | --- |
| **Best For** | Small to medium datasets | Big data / distributed processing |
| **Execution** | Local / single machine | Distributed across Spark cluster |
| **Memory handling** | In-memory, RAM-bound | Handles data across cluster nodes |
| **Ease of use** | Simple and intuitive | More complex setup but scalable |
| **Performance** | Fast for small datasets | Designed for scale, optimized for TBs+ |
| **Data volume** | MBs to low GBs | 10GB to petabyte-scale |
| **Integration** | Excellent for scripting, APIs, lightweight | Excellent for batch/streaming and data lakes |
| **Debugging** | Easier (runs locally) | Harder (jobs distributed) |
| **Parallelism** | Manual (threads/multiprocessing) | Built-in via Spark |
| **Deployment** | Lightweight (cron, Airflow, scripts) | Requires Spark infra (EMR, Databricks, etc.) |