

# Fabric vs Databricks: A Self-Assessment for Power BI Developers

Choosing between Microsoft Fabric and Databricks is not a tool comparison. It is a **starting-point decision**. Both platforms support SQL, Spark, notebooks, and BI. The difference is what **each platform assumes about you on day one**. This checklist is designed for **Power BI developers** to self-assess honestly. Not where you want to be in 2 years. Where you are **today**. Use it to decide which platform reduces friction, not which one sounds more powerful.

Area	Ask yourself honestly	Fabric	Databricks
Starting point	What do you start with today?	Power BI, Excel, business reporting	Engineering, backend, platforms
Coding background	Have you written production code before?	No, or very little	Yes, regularly
PySpark exposure	Have you ever used PySpark?	Never used it	Already used or learning
PySpark expectation	Is Spark required on day one?	No, optional	Yes, expected
SQL comfort	How do you work with data today?	Mostly SQL	SQL + code
Learning pressure	How fast must you deliver value?	Immediately	After ramp-up
Transformation style	How do you prefer building logic?	SQL, visual tools, assisted notebooks	Code-first pipelines
Power BI usage	How important is Power BI?	Primary output	One of many outputs
Data modelling	Where do you add most value?	Semantic models, measures	Data structures, pipelines
Scale today	What data size do you handle now?	Small to medium	Large to massive
Platform decisions	Who owns infra decisions?	Platform abstracts it	Engineers own it
Failure handling	Who thinks about retries, idempotency?	Platform handles most	You handle it
DevOps	How mature is your CI/CD?	Basic or none	Advanced
Team type	Who are your peers?	Analysts, BI Dev	Data Engineers
AI usage	How do you plan to use AI?	Embedded AI in analytics, Copilot-style assistance	Custom ML, feature engineering, model pipelines
Career motion	What do you want next?	Grow from BI to analytics engineering	Deepen data/platform engineering

## How to use this checklist:

- If most of your answers fall under **Fabric**, start there
- If most fall under **Databricks**, you're ready for it
- If you're split, Fabric first and Databricks later is a common and safe path

## Important clarification:

- Starting with Fabric does **not** limit you. It buys you time to strengthen fundamentals like SQL, data modelling, and end-to-end thinking.
- Databricks rewards strong engineering habits early. Fabric allows you to build them progressively. Choose the platform that matches your **current leverage**, not your ego.