Zadanie 2

Wybór technologii - składowanie danych w Databricks

- Porównanie technologii **DeltaLake** i **Iceberg**
- Podpowiedz klientowi w jakich scenariuszach każda z technologii będzie bardziej optymalna.

Delta Lake vs Apache Iceberg: porównanie technologii składowania danych w Databricks

Kategoria	Delta Lake	Apache Iceberg
Wydajność	Bardzo dobra dla małych i średnich zbiorów, szybki start na Databricks	Przewaga przy gigantycznych zbiorach, lepsze zarządzanie metadanymi
Time Travel (historyczność)	Tak, wygodne przeglądanie wersji	Też wspierane, choć z innymi mechanizmami
ACID	Pełna zgodność, transakcje jak w bazach relacyjnych	Tak, wsparcie dla transakcji ACID
Integracja z Azure	Bezproblemowa, natywne wsparcie Databricks i Azure	Wymaga doinstalowania connectorów lub dodatkowych narzędzi
Wsparcie BI	Popularne narzędzia BI wspierają Delta Lake bez problemów	Zależne od narzędzia, czasem wymaga dodatkowej konfiguracji
Konfiguracja	Bardzo prosta, szczególnie w środowisku Databricks	Zazwyczaj wymaga więcej ustawień, trudniej na start

Kiedy wybrać którą technologię?

- **Delta Lake:** najlepszy wybór, gdy zależy nam na ścisłej integracji z Databricks i Azure, prostym wdrożeniu i szybkim efekcie. Sprawdzi się przy projektach, gdzie ekosystem Microsoft jest priorytetem, a skala danych nie przekracza setek terabajtów.
- **Iceberg:** idealny przy bardzo dużych (petabajtowych) zbiorach i jeśli zależy Ci na pracy w różnych silnikach (np. Spark, Trino, Flink) lub na łatwej migracji między platformami. Lepiej skaluje się przy naprawdę wielkich projektach.

Zadanie 3

What is the medallion lakehouse architecture? - Azure Databricks | Microsoft Learn

Napisz krytykę architektury medalionu, może nie trzeba tworzyć medalionu, jakie ma błędy, wady ….

20 punktów i krótki opis do każdego.

Krytyka architektury medallion (Bronze-Silver-Gold)

Medallion architecture, czyli popularne podejście trójwarstwowe w Azure Databricks (Bronze = dane surowe, Silver = dane przetworzone, Gold = dane agregowane/analityczne), ma wiele zalet, ale też **nie jest uniwersalnym rozwiązaniem**. Warto być krytycznym – poniżej lista **20 wad i ryzyk** (każda krótko opisana):

1. **Rozbudowana złożoność:** Więcej warstw = więcej kodu, pipeline'ów i potencjalnych błędów.
2. **Opóźnienia:** Przepływ przez każdą warstwę wydłuża czas dostarczenia danych końcowych.
3. **Koszt składowania:** Te same dane są magazynowane wielokrotnie (każda warstwa = nowe pliki).
4. **Diagnozowanie błędów:** Trudniej znaleźć źródło problemu, bo błąd może powstać na każdym etapie.
5. **Nieoptymalność dla małych projektów:** Dla prostych rozwiązań niepotrzebna nadmiarowość.
6. **Duplikacja logiki:** To samo przetwarzanie może być powtarzane w kilku miejscach.
7. **Nie każda analiza wymaga Gold:** Często dashboardy lub analityka mogą korzystać już z warstwy Silver.
8. **Ryzyko niespójności:** Różne wersje danych mogą istnieć równolegle.
9. **Potrzeba automatyzacji:** Bez solidnego automatycznego zarządzania pipeline'y łatwo się psują.
10. **Duże zużycie zasobów:** Każda warstwa to osobne przetwarzanie – rosną koszty obliczeń.
11. **Versioning trudny do kontrolowania:** Zarządzanie wersjami między warstwami jest kłopotliwe.
12. **Granice warstw są płynne:** W praktyce trudno jasno oddzielić, gdzie kończy się jedna warstwa, a zaczyna druga.
13. **Problemy ze skalowaniem przy zmianach schematów:** Dynamicznie zmieniające się dane mogą rozbijać architekturę.
14. **Batch > streaming:** Medallion jest projektowany głównie do batch processing, streaming jest doklejany.
15. **Bariera wejścia:** Nowe zespoły muszą długo się wdrażać, żeby zrozumieć całość.
16. **Potrzeba dodatkowych narzędzi (np. dbt, Airflow):** Często trzeba zintegrować wiele rozwiązań.
17. **Warstwa Gold nie zawsze znaczy „analityczna" dla biznesu:** Bywa nieintuicyjna dla końcowego odbiorcy.
18. **Brak spójności wdrożeń:** Każdy zespół może rozumieć podział warstw inaczej.
19. **Łańcuch zależności:** Awaria na jednym etapie stopuje cały pipeline.
20. **Przewymiarowanie dla małych firm:** Startupy i małe organizacje często nie mają zasobów, by utrzymać taki model.