

Databricks: Lakehouse i tabele Delta – ingestia, ewolucja schematu i wydajność

Szkolenie Databricks: Lakehouse i tabele Delta to praktyczny kurs dla osób, które budują warstwę danych w architekturze Medallion. Program koncentruje się na dwóch decyzjach, od których zależy jakość całego Lakehouse: jak wybrać sposób zasilania danych dla danego źródła oraz jak zaprojektować tabele Delta, żeby przyrosty, zmiany schematu i aktualizacje były bezpieczne.

Uczestnicy przechodzą przez cały proces od źródła do warstwy Silver: wybierają mechanizm ingestii, budują przyrostowe ładowanie do Bronze z Auto Loader, obsługują ewolucję schematu i wykonują bezpieczny upsert do Silver poleceniem MERGE. Dodatkowo poznają mechanizmy Delta Lake związane z historią zmian, Change Data Feed i wydajnością tabel. Ćwiczenia są realizowane na scenariuszu RetailOps (fikcyjna firma handlowa, na której danych realizujemy wszystkie ćwiczenia).



Odbiorcy szkolenia

Szkolenie jest przeznaczone dla osób technicznych, które projektują lub budują przepływy danych w Databricks i chcą robić to zgodnie z dobrymi praktykami Lakehouse.

W szczególności kurs sprawdzi się dla:

- inżynierów danych budujących warstwy Bronze i Silver,
- programistów ETL przechodzących z klasycznych hurtowni danych na Lakehouse,
- architektów danych, którzy dobierają wzorce ingestii dla różnych źródeł,

- absolwentów szkolenia DBX-FND, którzy chcą przejść od pierwszego przepływu do rozwiązania gotowego na przyrosty danych.

Szkolenie będzie szczególnie wartościowe dla zespołów, które mają już pierwsze tabele w Databricks i chcą uporządkować sposób ich zasilania oraz aktualizacji.



Korzyści

- Architektura Medallion w praktyce – dowiesz się, gdzie przebiega granica między warstwami Bronze, Silver i Gold oraz jakie decyzje podejmujemy na każdej z nich.
- Wybór mechanizmu ingestii – nauczysz się dobierać COPY INTO, Auto Loader, streaming tables, Lakeflow Connect lub JDBC do rodzaju źródła.
- Przyrostowe ładowanie danych – zbudujesz ładowanie z Auto Loader z kontrolą schematu i obsługą nieoczekiwanych kolumn.
- Bezpieczne aktualizacje w Delta Lake – poznasz, jak wykonać aktualizację istniejących i dopisanie nowych rekordów (upsert) poleceniem MERGE i odczytać zmiany przez Change Data Feed.
- Wydajność i koszt tabel – dowiesz się, kiedy stosować liquid clustering, OPTIMIZE i VACUUM oraz jak uzasadnić decyzję optymalizacyjną.
- Przygotowanie do produkcji – zobaczysz, jak lineage i historia tabel pomagają utrzymać przepływ danych.



Program szkolenia

Moduł – Architektura Medallion i decyzje projektowe

Cel modułu: uczestnik wie, jak podzielić dane na warstwy i jakie decyzje podjąć przed napisaniem pierwszego procesu.

- Warstwy Bronze, Silver i Gold – surowe dane, dane oczyszczone i dane gotowe dla biznesu; co trafia do której warstwy
- Klucz biznesowy i ziarnistość – po czym rozpoznajemy, że dwa rekordy to ten sam obiekt (np. zamówienie)
- Idempotentność – proces uruchomiony dwa razy nie może zdublować danych
- Metadane ładowania – zapis, z jakiego pliku i kiedy przyszedł każdy rekord, żeby dało się odtworzyć historię
- Układ warstw w Unity Catalog – osobne katalogi lub schematy dla warstw i domen
- Lab: uczestnik projektuje warstwy, klucze i metadane dla danych RetailOps

2. Moduł – Wybór mechanizmu ingestii

Cel modułu: uczestnik potrafi dobrać sposób ładowania danych do rodzaju źródła.

- COPY INTO – proste ładowanie plików z folderu; pliki już załadowane są automatycznie pomijane;

użytkownikom SQL Databricks zaleca dziś streaming tables jako rozwiązanie bardziej skalowalne i odporne na błędy

- Auto Loader – ciągle lub cykliczne wykrywanie nowych plików, także przy dużej ich liczbie
- Streaming tables w SQL – ładowanie przyrostowe bez pisania kodu w Python (Databricks automatycznie tworzy dla nich pipeline serverless)
- Lakeflow Connect – gotowe konektory do aplikacji SaaS, baz danych (w tym odczyt zmian – CDC) i plików w SharePoint lub Google Drive; wymagania sieciowe i uprawnienia
- Bezpośrednie połączenie z bazą (JDBC) – kiedy ma sens, a kiedy nadmiernie obciąża system źródłowy
- Lab: uczestnik dobiera mechanizm ładowania do pięciu przykładowych źródeł i uzasadnia wybór

3. Moduł – Auto Loader: przyrosty i zmiany struktury plików

Cel modułu: uczestnik buduje ładowanie, które przetwarza tylko nowe pliki i nie psuje się, gdy w danych pojawi się nowa kolumna.

- Jak Auto Loader zapamiętuje, które pliki już przetworzył (checkpoint) i dlaczego każdy plik trafia do tabeli dokładnie raz
- Konfiguracja: format plików, miejsce zapisu schematu, podpowiedzi typów kolumn; sposób wykrywania nowych plików – przeglądanie folderu lub powiadomienia z chmury (Databricks zaleca powiadomienia oparte na file events)
- Zmiana struktury plików – nowa kolumna może zostać dodana do schematu (proces zatrzymuje się i po ponownym uruchomieniu działa z nową kolumną – addNewColumns), odłożona do _rescued_data (rescue) albo zatrzymać proces do decyzji zespołu (failOnNewColumns)
- Kolumna _rescued_data – miejsce na wartości, które nie pasują do schematu, zamiast ich utraty
- Uruchamianie cykliczne – przetworzenie wszystkich nowych plików i zatrzymanie (availableNow)
- Lab: uczestnik łączy trzy kolejne partie plików zamówień, w tym partię z nową kolumną

4. Moduł – Delta Lake: jak działają MERGE, historia i Change Data Feed

Cel modułu: uczestnik rozumie, jak Delta Lake wykonuje aktualizacje i zapisuje, co się w tabeli zmieniło.

- Historia tabeli – każda zmiana jest wersją; podgląd kto, kiedy i co zmienił
- Zmiana struktury tabeli – kiedy Delta zablokuje zapis, a kiedy pozwoli dodać nowe kolumny
- MERGE INTO – jak działa polecenie: dopasowanie rekordów po kluczu, potem dodanie nowych, aktualizacja istniejących i usunięcie nieaktualnych
- Duplikaty w danych wejściowych – dlaczego MERGE kończy się błędem, gdy kilka rekordów pasuje do jednego wiersza tabeli
- Powrót do wcześniejszej wersji tabeli (time travel, RESTORE) – powtórka z DBX-FND
- Change Data Feed – odczyt dodanych, zmienionych i usuniętych rekordów dla wybranego zakresu wersji; w nowej, automatycznej wersji zmiany są wyliczane w momencie odczytu i nie trzeba włączać tej funkcji na każdej tabeli osobno (wymaga row tracking i Databricks Runtime 19+); w ćwiczeniach używamy wersji klasycznej, włączanej na tabeli, która działa na Runtime LTS
- Lab: uczestnik aktualizuje tabelę Silver poleceniem MERGE, odczytuje listę zmian i cofa błędną aktualizację

5. Moduł – Wydajność i koszt tabel Delta

Cel modułu: uczestnik wie, co spowalnia zapytania do tabel i jak to naprawić.

- Problem wielu małych plików i jak Delta pomija pliki, które nie zawierają szukanych danych

- Liquid clustering – układanie danych według najczęściej filtrowanych kolumn lub automatyczny dobór kluczy (CLUSTER BY AUTO); następca partycjonowania i ZORDER
- Porządkowanie tabel: łączenie małych plików (OPTIMIZE) i usuwanie starych wersji (VACUUM)
- Predictive optimization – automatyczna konserwacja tabel przez platformę (OPTIMIZE, VACUUM, statystyki), domyślnie włączana dla tabel managed w Unity Catalog
- Profil zapytania – jak sprawdzić, ile danych faktycznie zostało odczytanych
- Lab: uczestnik porównuje w profilu zapytania liczbę odczytanych plików przed i po liquid clustering (OPTIMIZE)

6. Moduł – Warsztat zintegrowany i przygotowanie do produkcji

Cel modułu: uczestnik łączy elementy dnia w działający przepływ i wie, co sprawdzić przed przekazaniem go do produkcji.

- Warsztat: Auto Loader → tabela Bronze → aktualizacja tabeli Silver bez duplikatów dla danych zamówień RetailOps. Efektem jest przyrostowy przepływ, który można bezpiecznie uruchamiać wielokrotnie. Zadanie dodatkowe dla chętnych: odczyt zmian z tabeli Silver przez Change Data Feed i włączenie na niej liquid clustering.
- Lineage – sprawdzenie, skąd pochodzą dane w tabeli i kto z nich korzysta
- Monitorowanie – historia tabel i metryki ładowania
- Lista kontrolna przed przekazaniem przepływu do orkiestracji



Oczekiwane przygotowanie uczestnika

- Ukończone szkolenie DBX-FND (Fundamenty Databricks) lub równoważna wiedza: Unity Catalog, Volumes, tabele Delta
- Praktyczna znajomość SQL
- Zalecana podstawowa znajomość Python/PySpark – ćwiczenia z Auto Loader korzystają z gotowych szablonów kodu

Dla zwiększenia komfortu pracy oraz efektywności szkolenia zalecamy skorzystanie z dodatkowego ekranu. Brak dodatkowego ekranu nie jest przeciwwskazaniem do udziału w szkoleniu, ale w znaczący sposób wpływa na komfort pracy podczas zajęć

Informacje oraz wymagania dotyczące uczestniczenia w szkoleniach w formule zdalnej dostępne na: <https://www.altkomakademia.pl/jak-dzialamy/o-szkoleniach/?distance-learning>



Szkolenie obejmuje

* dostęp do portalu słuchacza AltKom Akademii

Szkolenie prowadzone jest w środowisku Databricks w chmurze. Każdy uczestnik otrzymuje własny workspace z dostępem do Unity Catalog, SQL Editor, Notebooków oraz katalogu z danymi testowymi.

Uwaga: W trakcie wybranych szkoleń Akademia zapewnia Uczestnikom dostęp do płatnych narzędzi i usług niezbędnych do realizacji programu szkolenia (w szczególności usług Azure, Fabric oraz innych zasobów chmurowych). Uczestnik zobowiązany jest do korzystania z udostępnionych zasobów wyłącznie w zakresie wynikającym z agendy szkolenia oraz zgodnie z poleceniami trenera. Wykorzystywanie tych zasobów w sposób wykraczający poza zakres szkoleniowy, w szczególności do realizacji projektów prywatnych, testowania dodatkowych usług, modyfikowania konfiguracji środowiska lub podejmowania działań generujących koszty niezwiązane z realizacją ćwiczeń – skutkować będzie obciążeniem Uczestnika kosztami przekraczającymi przewidziany limit, na podstawie wydatków poniesionych przez Akademię (<https://www.altkomakademia.pl/ogolne-warunki-uczestnictwa-w-szkoleniach/>)



Język

Wykład: polski

Materiały: angielski

Czas trwania

1 dni / 7 godzin

Opis egzaminu

Szkolenie nie zawiera formalnego egzaminu. Zaleca się jednak podejście do egzaminu Databricks Certified Data Engineer Associate jako kolejny krok rozwojowy.