

Databricks: DataOps – Bundles, Terraform i CI/CD

Szkolenie Databricks: DataOps – Bundles, Terraform i CI/CD to praktyczny warsztat automatyzacji wdrożeń w Databricks, **realizowane wyłącznie w formule dedykowanej dla organizacji (dla grup zamkniętych)**. Program koncentruje się na tym, jak przenosić joby i pipeline'y między środowiskami w sposób powtarzalny i kontrolowany, z wykorzystaniem Declarative Automation Bundles (wcześniej Databricks Asset Bundles), Git i potoków CI/CD.

Uczestnicy przechodzą przez cały proces: od podziału odpowiedzialności między Terraform (zasoby platformy) a Bundles (artefakty danych), przez strukturę bundle, testy i parametry, po wdrożenie do odizolowanego środowiska dev i zaprojektowanie promocji do test i prod. Warsztat jest realizowany na repozytorium i środowisku uzgodnionym z klientem przed szkoleniem, dzięki czemu uczestnicy pracują na układzie zbliżonym do własnego.



Odbiorcy szkolenia

Szkolenie jest przeznaczone dla zespołów, które wdrażają rozwiązania danych w Databricks i chcą zautomatyzować ten proces.

W szczególności kurs sprawdzi się dla:

- inżynierów danych przygotowujących joby i pipeline'y do wdrożenia,
- inżynierów DevOps i platformy odpowiedzialnych za środowiska Databricks,
- liderów technicznych ustalających zasady promocji zmian między środowiskami,
- zespołów, które dziś wdrażają zmiany ręcznie i chcą przejść na CI/CD.

Szkolenie będzie szczególnie wartościowe dla organizacji, które mają już działające procesy w Databricks i potrzebują bezpiecznego, powtarzalnego sposobu ich wdrażania.



Korzyści

- Podział odpowiedzialności – dowiesz się, co należy do Terraform, a co do Declarative Automation Bundles oraz jak dzielą się zadaniami zespoły platformy i danych.
- Declarative Automation Bundles – nauczysz się budować strukturę bundle z targetami, zmiennymi i nadpisaniami dla środowisk.
- Wdrożenie do dev – wykonasz bundle validate, bundle plan i bundle deploy na odizolowanym środowisku dev.
- Potok CI/CD – zaprojektujesz pipeline uruchamiany przy pull requestcie, z walidacją, testami i wdrożeniem.
- Promocja między środowiskami – poznasz kryteria i zatwierdzenia przy przejściu z dev do test i prod.
- Bezpieczeństwo wdrożeń – nauczysz się korzystać z service principal, run_as i sekretów oraz przeglądać zmiany w logach audytowych.



Program szkolenia

1. Moduł – Terraform i Bundles: kto za co odpowiada

Cel modułu: uczestnik wie, które elementy platformy wdraża zespół platformy, a które zespół danych, i jakimi narzędziami.

- Terraform – zakładanie i konfiguracja platformy: workspace'y, katalogi, polityki compute, uprawnienia
- Declarative Automation Bundles (wcześniej Databricks Asset Bundles) – wdrażanie tego, co buduje zespół danych: joby, pipeline'y, dashboardy, Genie Agents, modele
- Tożsamości – konta użytkowników, konta techniczne (service principal) i grupy; kto może wdrażać na które środowisko
- Logowanie narzędzi wdrożeniowych bez haseł w kodzie (OAuth, federacja tożsamości dla CI)
- Rozdzielenie środowisk dev / test / prod – osobne workspace'y lub osobne katalogi
- Lab: uczestnicy rozpisują podział odpowiedzialności dla swojej organizacji (dyskusja, ok. 10 min)

2. Moduł – Declarative Automation Bundles

Cel modułu: uczestnik opisuje job lub pipeline jako kod i wdraża go poleceniem.

- Czym jest bundle – plik konfiguracyjny databricks.yml i kod w repozytorium, które razem opisują rozwiązanie
- Szablony startowe projektu (databricks bundle init)
- Środowiska docelowe (targets) – te same zasoby wdrażane do dev i prod z różnymi ustawieniami
- Zmienne i nadpisanie – np. inny katalog, rozmiar compute lub harmonogram na każdym środowisku
- Cykl pracy: sprawdzenie konfiguracji (validate), podgląd zmian (plan), wdrożenie (deploy), uruchomienie (run)
- Wdrażanie bundle bezpośrednio przez API (silnik direct deployment) – od CLI 1.20.0 jedyny dostępny silnik; starsze wdrożenia oparte na Terraform są przenoszone automatycznie przy deploy

- Lab: uczestnik opisuje job i pipeline z repozytorium jako bundle i sprawdza plan wdrożenia na dev

3. Moduł - Testy, parametry i zasady promocji

Cel modułu: uczestnik wie, jak sprawdzić zmianę, zanim trafi na kolejne środowisko.

- Testy jednostkowe kodu transformacji - uruchamiane lokalnie lub w CI (pytest; Databricks Connect instalowany przed szkoleniem)
- Testy danych po wdrożeniu - reguły jakości i zapytania kontrolne
- Parametry zależne od środowiska - katalog, schemat, compute, harmonogram
- Kryteria przejścia dev → test → prod i kto zatwierdza zmianę
- Lab: uczestnik pisze test funkcji transformującej i parametryzuje katalog dla środowisk

4. Moduł - CI/CD dla Databricks

Cel modułu: uczestnik projektuje automatyczny proces sprawdzania i wdrażania zmian.

- Praca z Git - gałęzie, pull request i przegląd kodu; Git folders w workspace
- Potok CI - przy każdym pull request: testy, walidacja bundle i podgląd zmian
- Potok CD - po zatwierdzeniu: wdrożenie na test, a po akceptacji na prod
- GitHub Actions lub Azure DevOps - konfiguracja, sekrety i logowanie do Databricks
- Lab: uczestnik projektuje potok CI/CD dla repozytorium z warsztatu

5. Moduł - Warsztat: repozytorium, środowisko dev i wdrożenie

Cel modułu: uczestnik samodzielnie przeprowadza pełne wdrożenie na środowisko dev.

- Warsztat: na uzgodnionym repozytorium uczestnicy konfigurują środowisko dev, sprawdzają konfigurację i plan, wdrażają job lub pipeline, uruchamiają go i weryfikują wynik. Efektem jest działające, powtarzalne wdrożenie do środowiska dev.

6. Moduł - Terraform dla zasobów platformy

Cel modułu: uczestnik rozumie, jak zespół platformy zarządza konfiguracją Databricks jako kodem.

- Provider Databricks dla Terraform - konfiguracja na poziomie konta i workspace
- Przykładowe zasoby - katalogi, schematy, uprawnienia, polityki compute
- Stan konfiguracji, podgląd zmian i wykrywanie ręcznych zmian wprowadzonych poza Terraform
- Kiedy zasób powinien być w Terraform, a kiedy w bundle (pokaz)

7. Moduł - Bezpieczeństwo wdrożeń i plan dalszej promocji

Cel modułu: uczestnik zabezpiecza proces wdrożeń i planuje kolejne kroki.

- Sekrety - przechowywanie haseł i kluczy poza kodem (sekrety w Unity Catalog z uprawnieniami jak dla tabel, secret scopes Databricks, w Azure także oparte na Key Vault; sekrety w GitHub lub Azure DevOps dla potoków CI)
- Uruchamianie jobów na koncie technicznym (run_as) i uprawnienia do wdrożonych zasobów
- Logi audytowe - kto, co i kiedy wdrożył
- Lista kontrolna bezpieczeństwa i plan promocji na test i prod

Poza zakresem szkolenia pozostają: provisioning sieci, tożsamości w chmurze (IAM, Entra ID), prywatne endpointy i landing zone oraz wdrożenia na środowiska test i prod podczas zajęć. Te tematy wymagają osobnego warsztatu architektonicznego.



Oczekiwane przygotowanie uczestnika

- Praktyczna znajomość Git i pracy w terminalu
- Ukończone szkolenie DBX-LKF (Lakeflow) lub działający job albo pipeline w Databricks, który ma zostać wdrożony
- Przed szkoleniem: uzgodnione repozytorium, service principal z minimalnymi uprawnieniami i odizolowane środowisko dev
- Zainstalowany Databricks CLI i Databricks Connect (instrukcję wysyłamy przed szkoleniem)

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

Metoda szkolenia:

- uzgodnienie zakresu, repozytorium i środowiska przed szkoleniem
- warsztat na repozytorium i środowisku dev uzgodnionym z klientem
- materiały do ćwiczeń w formie elektronicznej
- Środowisko i zasoby chmurowe zapewnia Klient.



Czas trwania

1 dni / 7 godzin

Język

Wykład: polski

Materiały: angielski