

Projekt roczny z baz danych — założenia i zasady

Cel: zaprojektować, zbudować i obronić działającą bazę danych dla wybranej firmy — od analizy wymagań, przez strukturę, dane i zapytania, po procedury, wyzwalacze, transakcje, uprawnienia i kopię zapasową. Czas: rok szkolny (30 lekcji projektowych po 2 × 45 min + rezerwa).

1. Organizacja

- Projekt jest **indywidualny**. Temat (jeden z 10) przydziela nauczyciel na pierwszej lekcji; ten sam temat może mieć w klasie najwyżej 3 osoby.
- Każda osoba nadaje firmie **własną nazwę**, dopisuje co najmniej **2 własne reguły biznesowe** i dobiera własne progi (np. „ponad X zł”) — dlatego dwa projekty na ten sam temat nie mogą być takie same.
- Praca jest podzielona na **6 kamieni milowych (KM)**. Każdy KM ma termin oddania (lekcja) i jest oceniany osobno — patrz punkt 9 i harmonogram.
- Rezerwa 2–4 lekcji jest przeznaczona na poprawy, nieobecności i konsultacje.

2. Środowisko i konwencje

- Serwer: **MariaDB 10.4+** (XAMPP) lub **MySQL 8.0+**; klient konsolowy `mysql` i phpMyAdmin. Skrypty muszą działać w konsoli.
- Baza: `CHARACTER SET utf8mb4 COLLATE utf8mb4_polish_ci`, silnik **InnoDB** (klucze obce i transakcje).
- Nazwy tabel i kolumn: małe litery, bez polskich znaków, słowa rozdzielone podkreśleniem (`data_zwrotu`); tabele w liczbie mnogiej (`klienci`).
- Klucz główny: `id_...` (INT AUTO_INCREMENT) lub klucz złożony w tabelach pośredniczących. Klucze obce nazwane `fk_tabela_cel`.
- Kwoty: `DECIMAL(10,2)` (nigdy FLOAT). Daty: `DATE` lub `DATETIME` — nie tekst.
- Każdy skrypt zaczyna się komentarzem z nazwą projektu, autorem i opisem; kod jest czytelnie sformatowany.

3. Wymagania minimalne bazy

Struktura

- 8–12 tabel (minimalny zakres w karcie tematu — można go rozszerzać).
- Co najmniej jeden związek N:M z tabelą pośredniczącą.
- Klucze obce we wszystkich związkach, z uzasadnionym `ON DELETE`.
- Co najmniej 3 ograniczenia `CHECK`, 2 `UNIQUE`, kolumny `NOT NULL` i `DEFAULT` tam, gdzie wynikają z reguł.
- Tabela dziennika zmian (dla wyzwalacza W3).

Dane

- Dane **fikcyjne**, ale realistyczne (polskie imiona, miasta, ceny). Bez prawdziwych danych osobowych.
- Słowniki: 5–25 wierszy; osoby (klienci, pracownicy): 40–150; tabela zdarzeń: **co najmniej 300 wierszy** (liczby w karcie tematu).
- Daty zdarzeń od **1 stycznia poprzedniego roku do dziś** — wtedy zapytania o „poprzedni miesiąc / kwartał / rok” zawsze zwracają wyniki.
- Dane wolno generować zapytaniami (rekurencyjne CTE) lub importować z CSV.

4. Zapytania (Z1-Z12)

- Karta tematu zawiera 12 zapytań w 4 grupach. Każde zapytanie: treść, kod SQL, wynik (zrzut lub liczba wierszy i 3 pierwsze wiersze).
- Każde zapytanie zapisane w pliku `03_zapytania.sql` i jako widok `v_z01 ... v_z12`.
- Zakresy dat liczone względem `CURDATE()` — **bez wpisanych dat**. „Poprzedni miesiąc kalendarzowy” to pełny miesiąc przed bieżącym, a nie „ostatnie 30 dni”.
- Wartości „X zł” i nazwiska „o podanym nazwisku” dobierz tak, aby wynik miał 3–10 wierszy. Zapytanie z parametrem (Z5) dodatkowo jako procedura P2.

5. Logika w bazie

Element	Wymaganie
Funkcje F1-F2	z karty tematu; zwracają wartość, używane w co najmniej jednym zapytaniu
Procedury P1-P3	P1 — modyfikacja z parametrami IN i walidacją (<code>SIGNAL</code>); P2 — zapytanie Z5 z parametrem; P3 — parametry OUT
Wyzwalacze W1-W3	W1 — zmiana statusu po przekroczeniu progu; W2 — kontrola reguły biznesowej (<code>BEFORE</code> , <code>SIGNAL</code>); W3 — dziennik zmian lub blokada
Transakcje	procedura operacji biznesowej z <code>ROLLBACK</code> przy błędzie; obserwacja blokad i zakleszczenia w dwóch sesjach
Indeksy	co najmniej 2 indeksy z porównaniem <code>EXPLAIN</code> przed i po
Użytkownicy	3 konta wg ról z karty tematu; uprawnienia na tabele, widoki i procedury; test odmowy dostępu
Kopia	<code>mysqldump --routines --triggers</code> i odtworzenie na czystym serwerze

6. Struktura plików i oddawanie

Nazwisko_Imie_temat/	
00_informacje.sql	polecenia informacyjne (lekcja 1)
01_struktura.sql	CREATE DATABASE, CREATE TABLE, więzy
02_dane.sql	INSERT, generowanie danych
03_zapytania.sql	Z1-Z12 i widoki v_z01-v_z12
04_funkcje_procedury.sql	F1-F2, P1-P3
05_wyzwalacze.sql	W1-W3
06_transakcje.sql	procedura transakcyjna, opis testów współbieżności
07_indeksy.sql	indeksy i EXPLAIN
08_uzytkownicy.sql	konta i uprawnienia
09_testy.sql	polecenia, które MUSZĄ zakończyć się błędem, i testy poprawnych wywołań
install.sql	SOURCE 01 ... 08 – instalacja całej bazy jednym poleceniem
kopia.sql	kopia z mysqldump
dokumentacja.pdf	dokumentacja (wzór: Wzor_dokumentacji_projektu.docx)
diagram.png	diagram ERD

- Na każdy KM oddajesz **aktualny folder** (spakowany ZIP lub repozytorium Git — wg ustaleń z nauczycielem).
- Warunek przyjęcia KM2-KM6: `mysql -u root -p < install.sql` na czystym serwerze tworzy bazę bez błędów.

7. Dokumentacja projektu

1. **Opis firmy i wymagań** — czym zajmuje się firma, kto korzysta z bazy, lista wymagań i reguł biznesowych.
2. **Projekt konceptualny** — obiekty (encje) z opisem i związki.
3. **Projekt logiczny** — diagram ERD, normalizacja (uzasadnienie 3NF).
4. **Słownik danych** — tabela dla każdej tabeli bazy.
5. **Zapytania** — treść, SQL, wynik, krótki komentarz.
6. **Funkcje, procedury, wyzwalacze** — cel, kod, test (wywołanie i wynik, także błąd).
7. **Transakcje i współbieżność** — opis operacji, test COMMIT i ROLLBACK, zrzuty dwóch sesji z opisem różnic.
8. **Indeksy, użytkownicy, kopia** — EXPLAIN przed i po, tabela uprawnień, procedura odtworzenia.
9. **Podsumowanie** — co działa, czego nie udało się zrobić, co zrobiłbyś inaczej.

8. Polecenia informacyjne (lekcja 1, sprawdzane przy obronie)

```
SELECT VERSION(), CURDATE(), NOW(), USER(), CURRENT_USER();
SHOW DATABASES;          USE nazwa_bazy;          SHOW TABLES;
SHOW CREATE TABLE nazwa_tabeli;  DESCRIBE nazwa_tabeli;
SHOW TABLE STATUS LIKE 'nazwa_tabeli';  SHOW TRIGGERS;
SHOW PROCEDURE STATUS WHERE Db = DATABASE();  SHOW GRANTS FOR CURRENT_USER();
```

9. Ocenianie

KM	Zakres	Termin	Punkty	Punkty	Ocena roczna z projektu
KM1	Projekt bazy	lekcja 5	15 pkt	0–29	niedostateczny
KM2	Struktura i dane	lekcja 12	20 pkt	30–49	dopuszczający
				50–69	dostateczny
KM3	Zapytania i widoki	lekcja 18	25 pkt	70–84	dobry
				85–94	bardzo dobry
KM4	Funkcje, procedury, wyzwalacze	lekcja 23	20 pkt	95–100 + rozszerzenie	celujący
KM5	Transakcje i administracja	lekcja 28	10 pkt	Progi do dostosowania do statutu (WSO) szkoły. Ocena celująca wymaga co najmniej jednego rozszerzenia (punkt 11).	
KM6	Prezentacja i obrona	lekcja 30	10 pkt		
Razem			100 pkt		

- KM oddany do tygodnia po terminie — najwyżej 80% punktów; później — tylko w rezerwie, najwyżej 50%.
- Punkty za element przyznaje się, gdy **działa na serwerze nauczyciela** (z `install.sql`) i jest opisany w dokumentacji.
- Elementy można poprawiać do kolejnego KM (jedna poprawa na KM).

10. Samodzielność i narzędzia AI

Narzędzi AI (czaty, asystenci kodu) **wolno używać** jako pomocy: do wyjaśnienia błędu, podpowiedzi składni, generowania danych testowych. Warunki: w dokumentacji (rozdz. 9) wpisujesz, do czego użyłeś AI; każdy fragment kodu musisz umieć **wyjaśnić i zmodyfikować na żywo**. Element, którego nie potrafisz wyjaśnić podczas obrony, otrzymuje 0 punktów. Kopiowanie projektu innej osoby — 0 punktów za cały KM.

11. Rozszerzenia na ocenę celującą (co najmniej jedno)

- Prosta strona PHP (PDO) wyświetlająca wybrane widoki i wywołująca procedurę z formularza (rok 2, lekcje 20–22).
- Zdarzenie (EVENT) wykonywane cyklicznie, np. codzienne zamykanie przeterminowanych rezerwacji.
- Kalendarz dni z rekurencyjnego CTE, aby zapytanie Z1 pokazywało także dni z wynikiem 0.
- Kolumna JSON z dodatkowymi parametrami (np. wyposażenie, preferencje) i zapytania JSON_EXTRACT.
- Indeks pełnotekstowy i wyszukiwanie MATCH ... AGAINST w opisach.
- Role (CREATE ROLE) zamiast uprawnień nadawanych bezpośrednio użytkownikom.