

Import i eksport danych (CSV)

Czas: 2 × 45 min

praca indywidualna

Cel: Importowanie danych z plików CSV do tabel (phpMyAdmin i LOAD DATA) z poprawnym kodowaniem, separatorem i formatem liczb oraz eksportowanie danych do CSV.

Po zajęciach potrafisz:

- rozpoznaje w pliku CSV kodowanie, separator, nagłówek i znak końca wiersza
- importuje plik CSV w phpMyAdmin i poleceniem LOAD DATA LOCAL INFILE
- przekształca dane podczas importu (przecinek dziesiętny, format daty, kodowanie Windows-1250)
- eksportuje dane do CSV w phpMyAdmin i poleceniem SELECT ... INTO OUTFILE, rozumie rolę secure_file_priv
- sprawdza kompletność importu zapytaniem kontrolnym (COUNT, SUM)

Sprzęt i materiały

- komputer z Windows 11 i zainstalowanym XAMPP (Apache + MariaDB/MySQL)
- pliki `r1_13_import.sql`, `r1_13_klienci_utf8.csv` i `r1_13_cennik_excel.csv` z folderu Pliki_do_zadan (skopiuj folder np. do `C:\Pliki_do_zadan`)
- baza `sklep` (plik `sklep.sql`)

Oprogramowanie

- **XAMPP** — phpMyAdmin i konsola `mysql` [apachefriends.org/download.html](https://httpd.apache.org/download.html)
- **Visual Studio Code** — podgląd CSV: kodowanie i znak końca wiersza widać na pasku stanu code.visualstudio.com
- **LibreOffice Calc lub Excel** — otwarcie wyeksportowanego pliku pl.libreoffice.org/pobierz

Przygotowanie przed zajęciami

- Skopiuj folder Pliki_do_zadan na dysk lokalny tak, by ścieżka nie zawierała spacji ani polskich liter (np. `C:\Pliki_do_zadan`).

Bezpieczeństwo

- Importuj tylko do bazy `import_danych` (tworzy ją plik z zadania). Bazy `sklep` w tym zadaniu nie zmieniasz — tylko z niej eksportujesz.
- Opcja `local_infile` pozwala serwerowi czytać pliki z komputera klienta. Jeśli włączasz ją w MySQL 8.4, rób to tylko na własnym komputerze i wyłącz po zajęciach.
- Nie eksportuj danych osobowych (np. tabeli klientów) na pendrive'y ani do chmury — w zadaniu eksportujesz tylko tabelę produktów.

Przebieg zadania

1 Przygotowanie i oglądnięcie plików

1. Uruchom XAMPP (Apache i MySQL) i zaimportuj plik `r1_13_import.sql` (phpMyAdmin → Import). Sprawdź, że powstała baza `import_danych` z pustymi tabelami `klienci_nowi` i `cennik`.
2. Otwórz oba pliki CSV w VS Code. Dla każdego zapisz w tabeli sprawozdania: separator pól, czy jest nagłówek, kodowanie i znak końca wiersza (prawy dolny róg okna: np. UTF-8, LF/CRLF).
3. Jeśli w `r1_13_cennik_excel.csv` polskie litery wyglądają źle, kliknij kodowanie na pasku stanu → Reopen with Encoding → Central European (Windows 1250). Nie zapisuj pliku.

2 Import w phpMyAdmin

Tabela `klienci_nowi` ma kolumnę `id_klienta` (AUTO_INCREMENT), której w pliku nie ma — dlatego podajesz listę kolumn.

1. Z1. Zaimportuj `r1_13_klienci_utf8.csv`: baza `import_danych` → tabela `klienci_nowi` → Import. Ustaw: Kodowanie znaków pliku: utf-8; Format: CSV; Kolumny oddzielone: `;`; Kolumny dołączyć: `"`; Nazwy kolumn: `imie,nazwisko,email,miasto,data_rejestracji`; w części „Częściowy import” pole „Pomiń tę liczbę zapytań (dla SQL), zaczynając od pierwszej” ustaw na 1 (pominie wiersz nagłówka). Kliknij Import.
2. Z2. Napisz zapytanie, które policzy zaimportowane wiersze, oraz zapytanie wyświetlające klientów z miast zaczynających się od „Ł” lub „Bi” — sprawdź polskie znaki.
3. Z3. Jedna klientka nie podała miasta. Sprawdź dwoma zapytaniami, czy w kolumnie `miasto` jest NULL, czy pusty tekst `''`. Jeśli pusty tekst — napisz UPDATE, który zamieni puste miasta na NULL (najpierw SELECT z tym samym WHERE).

3 Import poleceniem LOAD DATA

Uruchom konsolę z obsługą plików lokalnych i polskich znaków. W MySQL 8.4 opcja serwera `local_infile` jest domyślnie wyłączona — sprawdź ją i w razie potrzeby włącz.

```
chcp 65001
mysql -u root --default-character-set=utf8mb4 --local-infile=1

SHOW GLOBAL VARIABLES LIKE 'local_infile';  -- OFF? wtedy: SET GLOBAL local_infile = 1;

-- szablon polecenia (uzupełnij):
LOAD DATA LOCAL INFILE 'C:/Pliki_do_zadan/plik.csv'  -- ukośniki /, nie \
INTO TABLE tabela
CHARACTER SET ...                                -- kodowanie pliku
FIELDS TERMINATED BY '...' OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '...'                        -- '\n' albo '\r\n'
IGNORE 1 LINES
(kolumna1, kolumna2, @zmienna)
SET kolumna3 = funkcja(@zmienna);
```

1. Z4. Wyczyść tabelę `klienci_nowi` (TRUNCATE) i zaimportuj ten sam plik poleceniem LOAD DATA LOCAL INFILE. Sprawdź liczbę wierszy i komunikat `Records/Skipped/Warnings`.
2. Z5. Zaimportuj `r1_13_cennik_excel.csv` do tabeli `cennik` „naiwnie”: bez CHARACTER SET, kolumny wprost (kod, nazwa, cena, data_od). Wykonaj `SHOW WARNINGS`; i obejrzyj dane: ceny, daty i polskie litery. Zapisz, co jest nie tak.
3. Z6. Wyczyść tabelę `cennik` i zaimportuj plik poprawnie: CHARACTER SET cp1250, cena i data przez zmienne `@cena`, `@data`, a w SET zamiana przecinka na kropkę (`REPLACE`) i zamiana daty funkcją `STR_TO_DATE(@data, '%d.%m.%Y')`.
4. Z7. Zapytanie kontrolne: liczba wierszy, suma cen, najwcześniejsza i najpóźniejsza data w tabeli `cennik`. Porównaj z plikiem (liczba wierszy bez nagłówka, przykładowe ceny).

4 Eksport danych

SELECT ... INTO OUTFILE zapisuje plik na komputerze serwera (u Ciebie — ten sam komputer). Zmienna `secure_file_priv` określa, w jakim folderze wolno zapisywać: pusta — bez ograniczeń, NULL — eksport wyłączony, ścieżka — tylko ten folder.

1. Z8. W phpMyAdmin: baza `sklep` → tabela `produkty` → Eksport → Metoda eksportu: „Dostosuj – wyświetl wszystkie możliwe opcje” → Format: „CSV dla MS Excel” → zaznacz „Umieść nazwy kolumn w pierwszym wierszu” → Eksport. Otwórz plik w arkuszu i sprawdź kolumny oraz polskie znaki.
2. Z9. Sprawdź wartość `secure_file_priv` (SHOW VARIABLES LIKE ...). Ustal folder, do którego możesz eksportować (w XAMPP np. `C:/xampp/tmp/`, w MySQL 8.4 folder podany w zmiennej).
3. Z10. Wyeksportuj wszystkie kolumny tabeli `sklep.produkty` poleceniem `SELECT * INTO OUTFILE '...' ... FROM sklep.produkty` (opcje pliku przed FROM) do pliku `produkty_eksport.csv`: CHARACTER SET utf8mb4, separator `;`, pola tekstowe w cudzysłowie, wiersze zakończone `\r\n`. Obejrzyj plik w VS Code: jak zapisano NULL w kolumnie kategorii huba USB-C i cudzysłów w nazwach monitorów?
4. Z11. Wykonaj Z10 jeszcze raz bez zmiany nazwy pliku i zanotuj numer błędu.
5. Z12. Wyeksportuj do pliku `produkty_naglowek.csv` te same dane z wierszem nagłówka. Wskazówka: `SELECT 'id_produktu', 'nazwa', ... UNION ALL SELECT ... FROM sklep.produkty INTO OUTFILE ...`.
6. Z13. Test „w obie strony”: w bazie `import_danych` utwórz tabelę `produkty_kopia` o strukturze jak `sklep.produkty` (`CREATE TABLE ... LIKE ...`) i wczytaj do niej plik z Z10 poleceniem LOAD DATA INFILE (bez LOCAL — plik leży na serwerze) z tymi samymi opcjami.
7. Z14. Zapytanie kontrolne: porównaj w obu tabelach liczbę wierszy, sumę wartości magazynu (`cena × stan`) i liczbę produktów bez kategorii. Zrób zrzut ekranu. Jeśli w Z4 włączyłeś `local_infile` w MySQL 8.4, wyłącz ją (`SET GLOBAL local_infile = 0;`).

W sprawozdaniu umieść

- ☐ tabela parametrów obu plików CSV (separator, nagłówki, kodowanie, koniec wiersza)
- ☐ tabela Z1–Z14 z poleceniami SQL (dla Z1 i Z8 — opis ustawień) i wynikami
- ☐ zrzut ustawień importu w phpMyAdmin (Z1) i wyniku Z3
- ☐ opis problemów z importu naiwnego (Z5) — ostrzeżenia i błędne dane — oraz poprawione polecenie (Z6)
- ☐ zrzut wyników zapytań kontrolnych Z7 i Z14
- ☐ odpowiedzi na pytania kontrolne

Pytania kontrolne

1. Dlaczego plik CSV zapisany w polskim Excelu ma zwykle średnik zamiast przecinka jako separator?
2. Co się stało z ceną 49,90 przy imporcie naiwnym i dlaczego serwer nie przerwał importu błędem? (Podpowiedź: słowo LOCAL.)
3. Czym różni się LOAD DATA LOCAL INFILE od LOAD DATA INFILE? Który plik czyta serwer w każdym z przypadków?
4. Do czego służy zmienna `secure_file_priv` i dlaczego dobrze, że w MySQL 8.4 domyślnie ogranicza eksport do jednego folderu?
5. Jak w pliku CSV zapisać wartość NULL, żeby LOAD DATA wczytało ją jako NULL, a nie jako pusty tekst?

Kryteria oceny

Kryterium	Punkty
Oględziny plików i import w phpMyAdmin (Z1–Z3)	4
LOAD DATA: poprawny import obu plików z przekształceniami (Z4–Z7)	6
Eksport phpMyAdmin i INTO OUTFILE, błąd 1086, nagłówki (Z8–Z12)	4
Test w obie strony i zapytania kontrolne (Z13–Z14)	3
Odpowiedzi na pytania kontrolne	3
Razem	20

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).