

# Bazy danych

mgr inż. Nikodem Bulanda

19 kwietnia 2023

## **Streszczenie**

Instalacja, konfiguracja i obsługa systemu zarządzania bazą danych MySQL 8.0.  
Ćwiczenia podstawowe, realizowaną w oparciu o oficjalną dokumentację.

# Spis treści

<b>1</b>	<b>Uruchomienie serwera SZDB MySQL 8.0</b>	<b>3</b>
1.1	Podstawowa konfiguracja . . . . .	3
1.2	Inicjalizacja . . . . .	3
1.3	Uruchomienie serwera . . . . .	4
1.4	Zatrzymanie serwera . . . . .	4
1.5	Diagnostyka serwera MySQL z wykorzystaniem narzędzia mysqladmin.exe i klienta mysqlshow.exe . . . . .	5
1.6	Uruchomienie klienta mysql . . . . .	6
1.7	Polecenie SHOW (Database Administration Statements) . . . . .	6
<b>2</b>	<b>DQL - Grupowanie i widoki</b>	<b>8</b>
2.1	Grupowanie . . . . .	8
2.1.1	Funkcje agregujące . . . . .	8
2.1.2	Grupowanie . . . . .	9
2.1.3	Grupowanie z klauzulą HAVING . . . . .	10
2.1.4	Grupowanie z klauzulą WITH ROLLUP . . . . .	10
2.2	Widoki . . . . .	11

# 1 Uruchomienie serwera SZDB MySQL 8.0

Niniejszy rozdział stanowić będzie skompensowany przewodnik po instalacji i uruchomieniu SZDB MySQL 8.0. Przybliżone zostaną w nim sposoby uruchomienia, zarządzania i konserwacji systemu bazodanowego. Pełny opis działań i dokumentacje znajdziecie w poniższym linku.

**Dokumentacja !** Przeczytaj i zapoznaj się z dokumentacją  
<https://dev.mysql.com/doc/refman/8.0/en/windows-install-archive.html>

## 1.1 Podstawowa konfiguracja

1. Pobierz archiwum ZIP SZBD mysql w wersji 8.0 - *"no debug"*  
<https://dev.mysql.com/downloads/mysql/>
2. Zawartość archiwum wypakuj do katalogu *C:\ISNS\mysql*.  
**[UWAGA!]** Pamiętaj aby przenieść tam zawartość archiwum a nie folder powstały po jego wypakowaniu
3. Stwórz plik konfiguracyjny i zapisz go w katalogu *C:\ISNS\mysql*

```
1 [mysqld]
2 # set basedir to your installation path
3 basedir=C:\ISNS\mysql
4 # set datadir to the location of your data directory
5 datadir=C:\ISNS\mysql\data
```

Listing 1: Plik konfiguracyjny my.ini - lokalizacja basedir

## 1.2 Inicjalizacja

Po zainstalowaniu<sup>1</sup> MySQL, konieczne będzie przeprowadzanie procesu inicjalizacji. W skład czynności inicjalizacyjnych wchodzić będzie między innymi:

- utworzenie katalogu *'data'*,
- utworzenie bazy konfiguracyjnej "mysql"<sup>2</sup>,
- utworzenie domyślnych kont użytkowników,
- deklaracja schematu domyślnego itp.

Inicjalizacja może odbywać się w dwóch trybach, zabezpieczonym i niezabezpieczonym. Tryby te odróżnia poziom domyślnie ustalanych zabezpieczeń. W trybie *-initialize-insecure*, użytkownikowi root zostaje nadane domyślnie **puste** hasło.

---

<sup>1</sup>w naszym przypadku osadzeniu plików

<sup>2</sup>ze względu na to iż SZDB MySQL jest systemem przechowywania informacji, nie dziwić powinien fakt iż większość ustawień przechowywanych jest właśnie w bazie

**[UWAGA!]** Proces inicjalizacji wykonaj tylko raz !!!  
**[UWAGA!]** Polecenie wykonaj z poziomu katalogu mysql "basedir".

```
1 bin\mysqld --initialize-insecure --user=mysql
2
3 lub
4
5 bin\mysqld --initialize-insecure --user=mysql
6 --defaults-file = C:\ISNS\mysql\my.ini
```

Listing 2: Inicjalizacja SZDB MySQL 8.X

### 1.3 Uruchomienie serwera

Uruchomienie najlepiej przeprowadzać z wiersza poleceń w oknie konsoli (lub „oknie DOS/CMD”). W ten sposób możesz wyświetlać komunikaty o stanie serwera w oknie wiersza poleceń, w którym był on uruchamiany. Komunikaty te ułatwiają identyfikację i naprawę problemów w razie ich wystąpienia. Aby uruchomić serwer, wpisz to polecenie:

```
1 bin\mysqld --defaults-file = C:\ISNS\mysql\my.ini --console
2
3 lub
4
5 bin\mysqld
```

Listing 3: Uruchomienie serwera bazy danych

W przypadku pominięcia opcji *--console* serwer zapisuje<sup>3</sup> dane diagnostyczne w dzienniku błędów wskazanym w konfiguracji lub w deklaracji *'in-line'* podczas uruchamiania *mysqld*.

### 1.4 Zatrzymanie serwera

Narzędzie administracyjne MySQL **mysqladmin.exe**, pozwala połączyć się z serwerem i nakazać mu zamknięcie. Narzędzie to łączy się z serwerem jako użytkownik **root** MySQL, (który jest domyślnym kontem administracyjnym w systemie MySQL). Jeśli konto użytkownika **root** MySQL posiada hasło, musisz wywołać **mysqladmin** z dodatkowym przełącznikiem *'-p'* i podać hasło, gdy zostaniesz o to poproszony.

```
1 bin\mysqladmin -u root shutdown
2
3 lub
4
5 bin\mysqladmin -u root -p shutdown
```

Listing 4: Zatrzymanie serwera

---

<sup>3</sup>jeśli zostało to skonfigurowane zgodnie z <https://dev.mysql.com/doc/refman/8.0/en/server-logs.html>

**[UWAGA!]** *Użytkownicy w systemie MySQL są całkowicie niezależni od użytkowników systemu operacyjnego Microsoft Windows czy UNIX/LINUX, chyba że zostało to skonfigurowane inaczej <sup>4</sup>.*

## 1.5 Diagnostyka serwera MySQL z wykorzystaniem narzędzia mysqladmin.exe i klienta mysqlshow.exe

Mysqladmin jest narzędziem przeznaczonym do wykonywania operacji administracyjnych w obrębie SZDB. Jego głównym przeznaczeniem jest kontrola konfiguracji i aktualnego stanu serwera, tworzenia i usuwania baz danych itp.

**Dokumentacja !** Przeczytaj i zapoznaj się z pełną dokumentacją <https://dev.mysql.com/doc/refman/8.0/en/mysqladmin.html>

**[Uwaga!].** Polecenie wykonaj z poziomu katalogu mysql "basedir"

```
1 bin\mysqladmin -u root -p variables
2
3 bin\mysqladmin -u root -p version
4
5 bin\mysqladmin -u root -p processlist
6
7 bin\mysqladmin -u root -p kill id,id....
8
9 bin\mysqladmin -u root -p create db_name
10
```

Listing 5: Diagnostyka serwera

Klienta<sup>5</sup> **mysqlshow.exe** służy do szybkiej kontroli i przeglądu bazy danych, ich tabele, kolumny czy indeksów tabel. Mysqlshow zapewnia interfejs wiersza poleceń dla SQL-owej instrukcji SHOW.

**[Uwaga!].** Polecenie wykonaj z poziomu katalogu mysql "basedir"

```
1 bin\mysqlshow -u root -p
2
3 bin\mysqlshow -u root -p mysql
4
5 bin\mysqlshow -u root -p mysql user
6
```

Listing 6: Wyświetlenie informacji o bazie danych

---

<sup>5</sup>narzędzie

## 1.6 Uruchomienie klienta mysql

Należy pamiętać, że serwer bazodanowy MySQL oparty został o architekturę klient-serwer<sup>6</sup>. Mimo tego, iż fakt ten jest ogólnie znany, intuicyjny i niezaskakujący, to nader często szczególnie w przypadku młodych użytkowników tej technologii, dochodzi do próby łączenia się z usługą która nie została wcześniej uruchomiona (czytaj *mysqld.exe*) jak i mylnego utożsamiania klienta serwera MySQL (*mysql.exe*) z samym serwerem bazodanowy (*mysqld.exe*).

**Mysql.exe** to prosta powłoka SQL z możliwością edycji i wprowadzania poleceń w języku SQL. W przypadku użycia trybu interaktywnego wyniki zapytania są prezentowane w formacie tabeli ASCII. W trybie nieinteraktywnym (na przykład jako filtr), wynik jest prezentowany w formacie oddzielonym tabulatorami. Format wyjściowy można zmienić za pomocą opcji poleceń.

```
1 bin\mysql -u root -p -h localhost -P 3306
```

Listing 7: Uruchomienie klienta mysql

Mysql wysyła każdą wydaną instrukcję SQL do serwera w celu wykonania. Istnieje również zestaw poleceń, które sam mysql<sup>7</sup> interpretuje. Aby uzyskać listę tych poleceń, wpisz `help` lub `\h` po znaku zachęty serwera (*mysql>*).

**[Uwaga!]. Polecenie wykonaj z poziomu klienta mysql, po wcześniejszym zalogowaniu. Zapoznaj się z możliwościami klienta mysql.**

```
1 mysql> help
```

Listing 8: Uruchomienie klienta mysql

## 1.7 Polecenie SHOW (Database Administration Statements)

Podstawowym zadaniem klauzuli **SHOW**<sup>8</sup> jest dostarczenie informacji o bazach danych, tabelach, kolumnach lub informacji o statusie serwera bazodanowego.

**Dokumentacja !** Poniżej zaprezentowano jedynie kilka podstawowych poleceń, całą listę dostępnych kombinacji znajdziesz w dokumentacji.  
<https://dev.mysql.com/doc/refman/8.0/en/show.html>

---

<sup>6</sup>Podobnie jak ORACLE, POSTGRES, MSSQL i wiele innych

<sup>7</sup>klient, mysql.exe

<sup>8</sup>wywołanej jako polecenie języka SQL

Poniżej wykonaj i sprawdź działanie kilku zapytań z zakresu administrowania bazą danych.

```
1 mysql> SHOW DATABASES;
```

Listing 9: Listowanie dostępnych baz

```
1 mysql> SHOW TABLES;
```

Listing 10: Listowanie tabel domyślnej bazy / konieczne jest wybranie bazy domyślnej

```
1 mysql> USE db_name;
```

Listing 11: Wybieranie domyślnej bazy

```
1 mysql> SHOW TABLES;
```

Listing 12: Wyświetlenie listy tabel

```
1 mysql> \h SHOW
```

Listing 13: Lista możliwości polecenia show

Instrukcja **DESCRIBE** jest synonimem, używanym do uzyskiwania informacji o strukturze tabel. W przypadku tabeli zadziała dokładnie jak 'SHOW COLUMNS FROM'

```
1 mysql> DESC user;  
2  
3 lub  
4  
5 mysql> DESCRIBE user;  
6  
7 lub  
8  
9 mysql> SHOW COLUMNS FROM user;
```

Listing 14: Wyświetlenie struktury tabeli user

## 2 DQL - Grupowanie i widoki

### 2.1 Grupowanie

Grupowanie to działanie sql mające na celu wyodrębnienie podzbiorów przetwarzanej kolekcji, odznaczających się tymi samymi właściwościami. Realizacja grupowania w większości SZDB, realizowana jest poprzez klauzulę GROUP BY jak i klauzule modyfikatorów, takich jak HAVING czy WITH ROLLUP. Użycie funkcji agregującej w instrukcji nie zawierającej klauzuli GROUP BY jest równoznaczne z grupowaniem po wszystkich wierszach. Dodatkowo o ile nie określono inaczej, funkcje agregujące ignorują wartości NULL.

Filtrowanie grup realizowane jest poprzez klauzulę HAVING która działa na całych grupach wierszy. Klauzula ta określana jest również mianem selekcji poziomej (grup wierszy).<sup>9</sup>

**Dokumentacja !** Przeczytaj i zapoznaj się z dokumentacją  
<https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions-and-modifiers.html>

#### 2.1.1 Funkcje agregujące

- **COUNT()** – zwraca liczbę wierszy otrzymanych w wyniku zapytania,
- **COUNT(DISTINCT)** - zwraca liczbę unikalnych wartości (wierszy),
- **MIN()** – określa wartość minimalną dla wybranej kolumny w wyniku realizacji zapytania,
- **MAX()** – określa wartość maksymalną dla kolumny w w wyniku realizacji zapytania,
- **SUM()** – sumuje zawartość kolumny (lub wyrażenia) dla każdego wiersza będącego wynikiem zapytania,
- **AVG()** – wylicza średnią arytmetyczną zawartości kolumny (lub wyrażenia) dla każdego wiersza będącego wynikiem zapytania,
- **GROUP\_CONCAT()** - funkcja zwraca wynik w postaci ciągu z połączonymi wartościami grupy innymi niż NULL. Umożliwiając oddzielenie ich poprzez określony znacznik np " , ",
- **MEDIAN()** – wylicza medianę
- **STDDEV()** - oblicza odchylenie standardowe
- **VARIANCE()** - oblicza wariancję

Funkcje grupujące mogą zostać wykorzystane bez klauzuli GROUP BY, w takiej sytuacji cały zbiór traktowany jest jako pojedyncza grupa. Ponadto należy pamiętać, że wszystkie funkcje grupujące, z wyjątkiem funkcji COUNT(\*), ignorują

---

<sup>9</sup>Analogicznie warunki określone w klauzurze WHERE, traktowane są jako selekcję poziomą pojedynczych rekordów.



wartości Null.

```
1 mysql> SELECT SUM(s.salary) suma, COUNT(*) ile
2 FROM employees e INNER JOIN salaries s ON e.emp_no = s.emp_no
3 WHERE e.emp_no BETWEEN 10000 AND 11000
```

Listing 15: Wykorzystanie funkcji agregujących

```
1 mysql> SELECT e.first_name, SUM(s.salary) suma, COUNT(*) ile
2 FROM employees e INNER JOIN salaries s ON e.emp_no = s.emp_no
3 WHERE e.emp_no BETWEEN 10000 AND 11000;
```

Listing 16: Wykorzystanie funkcji agregujących - zapytanie błędne

```
1 mysql> SELECT e.first_name, e.last_name,
2 SUM(s.salary) suma, COUNT(*) ile
3 FROM employees e INNER JOIN salaries s ON e.emp_no = s.emp_no
4 WHERE e.first_name = 'Sumant' AND e.last_name = 'Peac';
```

Listing 17: Wykorzystanie funkcji agregujących 1

**Zadanie 1** Dokonaj analizy powyższych listingów i przedstaw wnioski.

**Zadanie 2** Sprawdź ustawienia swojego serwera za pomocą "SELECT @@session.sql\_mode; ". Opisz za co odpowiada zmienna globalna ONLY\_FULL\_GROUP\_BY ?

### 2.1.2 Grupowanie

Dokonaj przedstawienia listy wszystkich pracowników, podaj ich imię i nazwisko wraz z podsumowaniem zarobków za cały okres pracy. Wyniki posortuj po nazwisku i imieniu pracownika.

```
1 mysql> SELECT e.first_name, e.last_name, SUM(s.salary) suma
2 FROM employees e INNER JOIN salaries s ON e.emp_no = s.emp_no
3 WHERE e.emp_no BETWEEN 10000 AND 11000
4 GROUP BY e.first_name, e.last_name
5 ORDER BY e.last_name, e.first_name;
```

Listing 18: Grupowanie

**Zadanie 3** Zmodyfikuj powyższe zapytanie zwracając informacje o ilości wypłaconych uposażeń, maksymalnej i minimalnej wartości uposażenia.

**Zadanie 4** Na podstawie informacji z zadania powyżej wylicz średnią i porównaj ją z wynikiem funkcji agregującej AVG().

**Zadanie 5** Zbadaj działanie funkcji GROUP\_CONCAT() i dokonaj zapisania w komunie "lista", listy wartości wszystkich uposażeń obdzielając wartości przecinkiem. Pamiętaj ilość elementów w liście musi pokrywać się z wartością zwracaną przez kolumnę "ilosc\_wyplat" przeliczoną na podstawie funkcji COUNT(\*). W projekcji wyświetl ponadto imię i nazwisko oraz sumę uposażeń otrzymanych przez danego pracownika.

### 2.1.3 Grupowanie z klauzulą HAVING

Dokonaj przedstawienia listy wszystkich pracowników, podaj ich imię i nazwisko wraz z podsumowaniem zarobków za cały okres pracy i ilością wypłat. Wyniki posortuj po nazwisku i imieniu pracownika przedstaw jedynie pracowników których suma wypłat przekroczyła 200 tys a ilość była mniejsza od 5.

```
1  mysql> SELECT e.first_name, e.last_name,
2  SUM(s.salary) suma, COUNT(s.salary) ilosc_wyplat
3  FROM employees e INNER JOIN salaries s ON e.emp_no = s.emp_no
4  WHERE e.emp_no BETWEEN 10000 AND 11000
5  GROUP BY e.first_name, e.last_name
6  HAVING SUM(s.salary) > 200000 AND COUNT(s.salary) < 5
7  ORDER BY e.last_name, e.first_name;
```

Listing 19: Grupowanie z klauzulą HAVING

**Zadanie 6** Na podstawie powyższego listingu i wykorzystując funkcję HAVING, dokonaj odfiltrowania gdy średnia jednego uposażenia nie przekroczyła 45000.

### 2.1.4 Grupowanie z klauzulą WITH ROLLUP

Klauzula GROUP BY pozwala na użycie modyfikatora WITH ROLLUP, rozszerzającego wynik podsumowania o dodatkowe wiersze reprezentujące operacje podsumowujące wyższego poziomu (czyli super agregujące). ROLLUP w ten sposób umożliwia odpowiadanie na pytania na wielu poziomach analizy, wykorzystując do tego jedno zapytanie.

```
1  mysql> SELECT e.first_name, e.last_name, SUM(s.salary) suma,
2  COUNT(s.salary) ilosc_wyplat, t.title
3  FROM employees e
4  INNER JOIN salaries s ON e.emp_no = s.emp_no
5  INNER JOIN titles t ON e.emp_no = t.emp_no
6  WHERE e.emp_no BETWEEN 10000 AND 10100
7  GROUP BY e.first_name, e.last_name, t.title ;
```

Listing 20: Grupowanie bez ROLLUP

```

1  mysql> SELECT e.first_name, e.last_name, SUM(s.salary) suma,
2  COUNT(s.salary) ilosc_wyplat, t.title
3  FROM employees e
4  INNER JOIN salaries s ON e.emp_no = s.emp_no
5  INNER JOIN titles t ON e.emp_no = t.emp_no
6  WHERE e.emp_no BETWEEN 10000 AND 10100
   GROUP BY e.first_name, e.last_name, t.title WITH ROLLUP;

```

Listing 21: Grupowanie z modyfikatorem ROLLUP

**Zadanie 7** Dokonaj podsumowania uposażeń dla każdego pracownika, grupując dodatkowo zbiór po roku urodzenia i przedstawiając sumę uposażeń.

## 2.2 Widoki

"Widok jest predefiniowanym i zapisanym po stronie serwera zapytaniem, którego wynik może być wielokrotnie odczytywany. W standardzie ANSI SQL widoki określane są mianem tabel widokowych lub wirtualnych (ang. Viewed, virtual tables), natomiast tabele przechowujące dane noszą miano tabel bazowych (ang. Base tables)....<sup>10</sup> Inną funkcjonującą w literaturze nazwą widoku, jest perspektywa.

**Dokumentacja !** Przeczytaj i zapoznaj się z dokumentacją  
<https://dev.mysql.com/doc/refman/8.0/en/create-view.html>  
<https://dev.mysql.com/doc/refman/8.0/en/drop-view.html>

```

1  CREATE
2  [OR REPLACE]
3  [ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}]
4  [DEFINER = user]
5  [SQL SECURITY { DEFINER | INVOKER }]
6  VIEW view_name [(column_list)]
7  AS select_statement
8  [WITH [CASCADED | LOCAL] CHECK OPTION]

```

Listing 22: Struktura instrukcji

```

1  DROP VIEW [IF EXISTS]
2  view_name [, view_name] ...
3  [RESTRICT | CASCADE]

```

Listing 23: Usówanie widoków

**Zadanie 8** Zbuduj widok w którym przedstawisz wszystkie kolumny z tabeli "employees" widok nazwij pracownicy. Sprawdź listę dostępnych tabel, zbadaj strukturę obiektu pracownicy poprzez polecenie DESCRIBE i SHOW CREATE VIEW analizując szczególnie polecenie SHOW.

<sup>10</sup>Poradnik webmastera - wydawnictwo Hellion - stan na 19.04.2023

**Zadanie 9** Wybierz wszystkie atrybuty wcześniej stworzonego widoku dla użytkownika 10000 i 10001.

**Zadanie 10** Wypełnij danymi rekord dla użytkownika o emp\_no 10000 jako imię użyj "Jan" a nazwisko "Kowalski", resztę danych wypełnij według uznania. Ponów zapytanie z zadania 8.

**Zadanie 11** Zmodyfikuj widok "pracownik", ograniczając atrybuty jedynie do first\_name, last\_name, emp\_no, title.

**Zadanie 12** Stwórz widok "pracownik2" w którym na etapie tworzenia zadeklarujesz nazwy kolumn widoku (poza poleceniem "SELECT". Nazwy kolumn widoku i ich dopasowanie powinno wyglądać następująco: imię -> first\_name, nazwisko -> last\_name, numer\_pracownika -> empl\_no.

**Zadanie 13** Do wyżej stworzonego widoku spróbuj wprowadzić dane ( nadaj emp\_no wartość 9999) jak i usunąć jeden z rekordów ( gdzie emp\_no = 10003). CO zauważyłeś ?