

SQL

Structured Query Language



Tomasz Woliński

StormIT.pl
fb.com/stormitPL

Poznajmy się

Agenda

- Co to jest baza danych?
- Instalacja MySQL
- Obsługa konsoli MySQL
- Tabele, typy danych, relacje, klucze obce
- Zarządzanie danymi, dodawanie, modyfikacja, usuwanie
- Odczytywanie i filtrowanie danych
- Łączenie danych z kilku tabel
- Indexy
- Użytkownicy i uprawnienia

Baza danych

Co to jest?

Jakie znasz bazy danych?

Czym jest Baza danych

Zbiór danych, w szczególności cyfrowych danych, ułożonych w **ustalonej strukturze**, zapisanych zgodnie z ustalonymi regułami dla danej aplikacji/systemu.

Podział Baz danych

Wybrane rodzaje:

- ✓ Relacyjne
- ✓ Obiektowe
- ✓ Graflowe
- ✓ Nierelacyjne

Zarządzanie bazą danych

RDMS

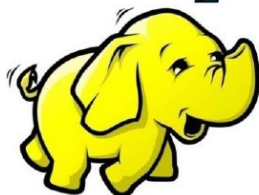
- ✓ **RDBMS** – Relational Database Management System
- ✓ System zarządzania relacyjną bazą danych
- ✓ Zbiór narzędzi służących do korzystania i zarządzania bazami danych

Implementacje Baz danych



cassandra

hadoop



PostgreSQL



mongoDB

ORACLE®
DATABASE

Diagram związków encji

Diagram ERD (ang. *Entity-Relationship Diagram*)

- ✓ Projekt relacyjnej bazy danych rozpoczynamy od koncepcji
- ✓ Rodzaj graficznego przedstawienia związków pomiędzy encjami

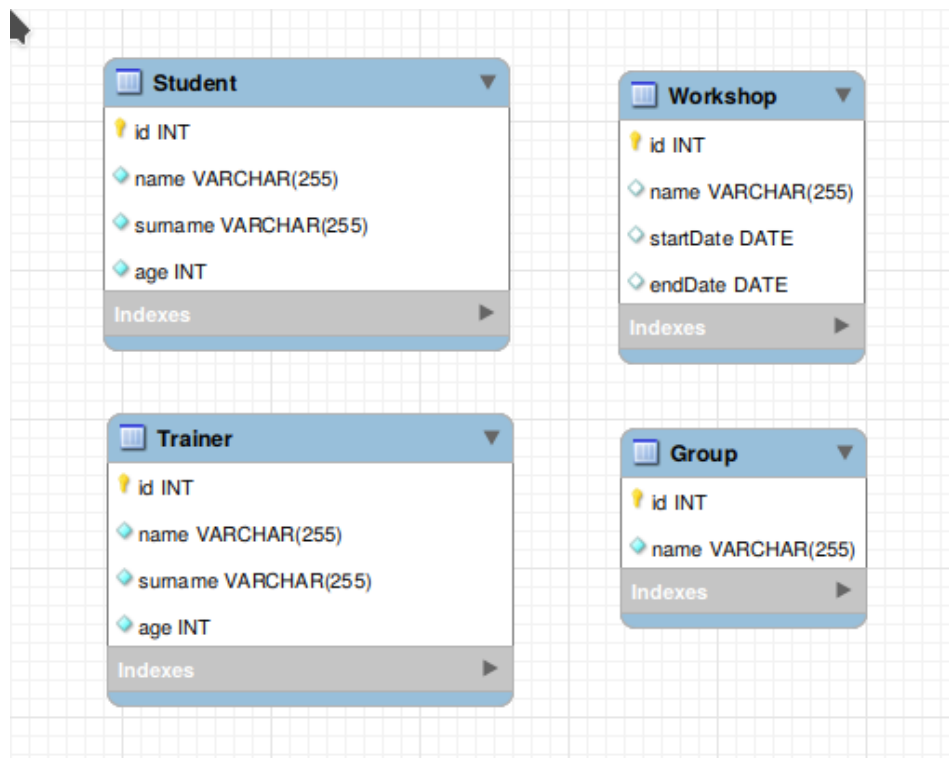
MySQL Workbench

Zarządzanie bazami danych

- ✓ Dedykowane narzędzie od Oracle do obsługi baz danych MySQL
- ✓ Instalacja
sudo apt-get install mysql-workbench

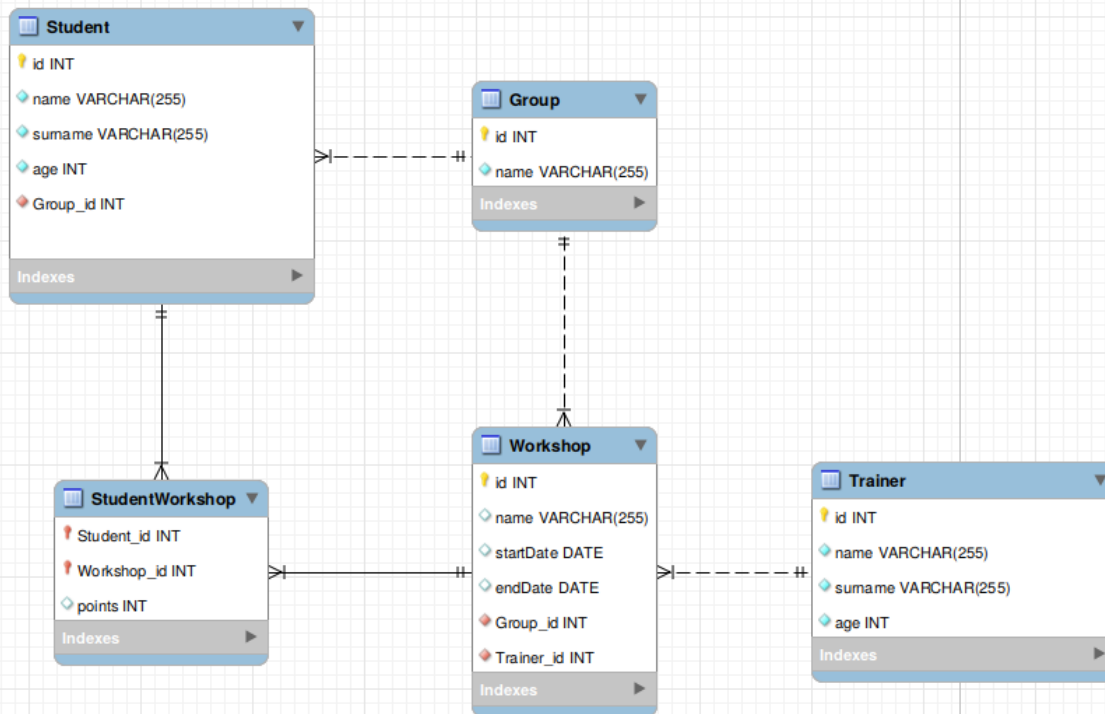
MySQL Workbench

Identyfikacja encji



MySQL Workbench

Relacje





MySQL

Własny serwer

MySQL Server

Docker

✓ https://hub.docker.com/_/mysql/

✓ `docker run -p 3306:3306 --name mysql
-e MYSQL_ROOT_PASSWORD=root
-v /home/<user>/mysql/data:/var/lib/mysql
-d mysql:latest`

MySQL Client

- ✓ Konsolowy klient mysql
- ✓ Zazwyczaj wykorzystywany na zdalnych maszynach
- ✓ `sudo apt-get install mysql-client-core-5.7`
- ✓ `mysql -h<host> -u <user> -p`
- ✓ `docker inspect mysql | grep IPAddress`

Podstawowe użycie klienta

Konsola MySQL

MySQL client

Pierwsze zapytania do bazy danych

✓ Zapytanie bazy o wersję serwera:

```
> select version();
```

✓ Użycie SQL jako kalkulatora

```
> select (1+pi())*3;
```

✓ Kolejowanie zapytań w jednej linii:

```
> select now(); select version();
```

MySQL

database

✓ Wyświetlenie listy baz

```
> show databases;
```

✓ Utworzenie nowej bazy danych:

```
> create database <name> default character set utf8;
```

✓ Usunięcie bazy danych

```
> drop database <name>;
```

✓ Przełączenie się do bazy danych:

```
> use <name>;
```

Tabele

Forma do przechowywania danych

MySQL – Tabele

Lista tabel

✓ Wyświetlenie wszystkich tabel

```
> show tables;
```

✓ Usunięcie tabeli

```
> drop table <name>;
```

MySQL – Tabele

Utworzenie tabeli

```
CREATE [TEMPORARY] TABLE [IF NOT EXISTS]  
tbl_name  
    (create_definition, ...)  
    [table_options]  
    [partition_options]
```

✓ <https://dev.mysql.com/doc/refman/5.7/en/create-table.html>

MySQL – Tabele

Utworzenie tabeli

- ✓ W przypadku braku wybranej domyślnej bazy danych nazwę definiujemy wraz z nazwą bazy danych:
``<database>`.`<tablename>``
- ✓ `IF NOT EXISTS` – oznacza, że tabela zostanie stworzona tylko jeżeli jeszcze nie istnieje. Przydatne przy wielokrotnym uruchamianiu skryptów.
- ✓ `TEMPORARY tables` – tabele tymczasowe, które istnieją tylko na czas trwania sesji.

MySQL – Tabele

Typy danych

✓ Numeryczne

- ✓ Stało przecinkowe
- ✓ Zmiennie przecinkowe
- ✓ Logiczne (bool)

✓ Data/czas

✓ Znakowe

✓ BLOB (Binary Large Object)

- ✓ <https://dev.mysql.com/doc/refman/5.7/en/data-types.html>

Typy danych

Numeryczne

Type	Storage (Bytes)	Minimum Value Signed	Minimum Value Unsigned	Maximum Value Signed	Maximum Value Unsigned
TINYINT	1	-128	0	127	255
SMALLINT	2	-32768	0	32767	65535
MEDIUMINT	3	-8388608	0	8388607	16777215
INT	4	-2147483648	0	2147483647	4294967295
BIGINT	8	-2^{63}	0	$2^{63}-1$	$2^{64}-1$

<https://dev.mysql.com/doc/refman/5.7/en/integer-types.html>

Typy danych

Zmiennie przecinkowe

- ✓ FLOAT
- ✓ DOUBLE
- ✓ Wartości przybliżone
- ✓ Kłopotliwe przy dokładnym porównywaniu
- ✓ Błędy zaokrągleń

Typy danych

Stało przecinkowe

- ✓ DECIMAL (n, p)
 - ✓ n – łączna ilość cyfr
 - ✓ p – ilość cyfr po przecinku

np. DECIMAL(5, 2)
min wartość: -999.99
max wartość: 999.99

Typy danych

Daty

✓Format dat

`yyyy-mm-dd`

`np. : 2018-11-19`

✓Format czasu

`hh:mm:ss`

`np. : 16:31:12`

Zadania

1) Tworzymy podstawowe tabele, bez relacji i bez dodatkowych właściwości pól.

a) Student (id, name, surname, age)

b) Workshop (id, name, startDate, endDate)

c) Trainer (id, name, surname, age)

d) Group (id, name)

Kolumny – własności

Definiowanie dodatkowych własności kolumn

Klucz unikatowy

Unique key

Zbiór atrybutów, które jednoznacznie identyfikują każdy pojedynczy wiersz w tabeli.

- ✓ Może być wartością pustą (*null*)
- ✓ Musi być unikatowy w całym zbiorze
- ✓ Może występować więcej niż jedno oznaczenia klucza unikatowego w tabeli

Klucz główny

Primary key

Zbiór atrybutów, które jednoznacznie identyfikują każdy pojedynczy wiersz w tabeli.

- ✓ Nie może być wartością pustą (*null*)
- ✓ Musi być unikatowy w całym zbiorze
- ✓ Może być tylko jedno oznaczenia klucza głównego w tabeli

Tabele – kolumny

Przykładowe własności kolumn

- ✓ *auto_increment* – wartość jest automatycznie zwiększona z każdym kolejnym rekordem
- ✓ *not null* – wartość nie może przyjmować null
- ✓ *default <value>* – wartość domyślna
- ✓ *unique* – unikatowa nazwa w obrębie tabeli

Tabele – modyfikacja

Modyfikacja istniejącej kolumny

- ✓ Modyfikacja istniejących już kolumn
- ✓ `ALTER TABLE tbl_name ...`
- ✓ <https://dev.mysql.com/doc/refman/8.0/en/alter-table.html>

Zadania

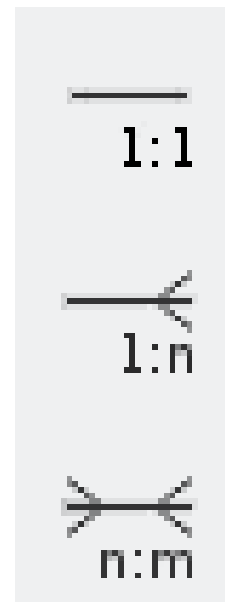
- 1) Zmodyfikuj istniejącą strukturę ustawiając wszystkie pola *id* jako klucze główne z automatycznie generowanymi wartościami
- 2) Dodaj ograniczenie *unique na* imię + nazwisko dla tabel *Student* i *Trainer*
- 3) Zablokuj możliwość ustawienia wartości *null* dla wszystkich pól *name*
- 4) Ustaw wartość domyślną *wiek=18* dla *Student* i *Trainer*
- 5) Przygotuj nową wersję skryptów *create table* ze wszystkimi powyższymi warunkami

Relacje i klucze obce

Budowanie relacji

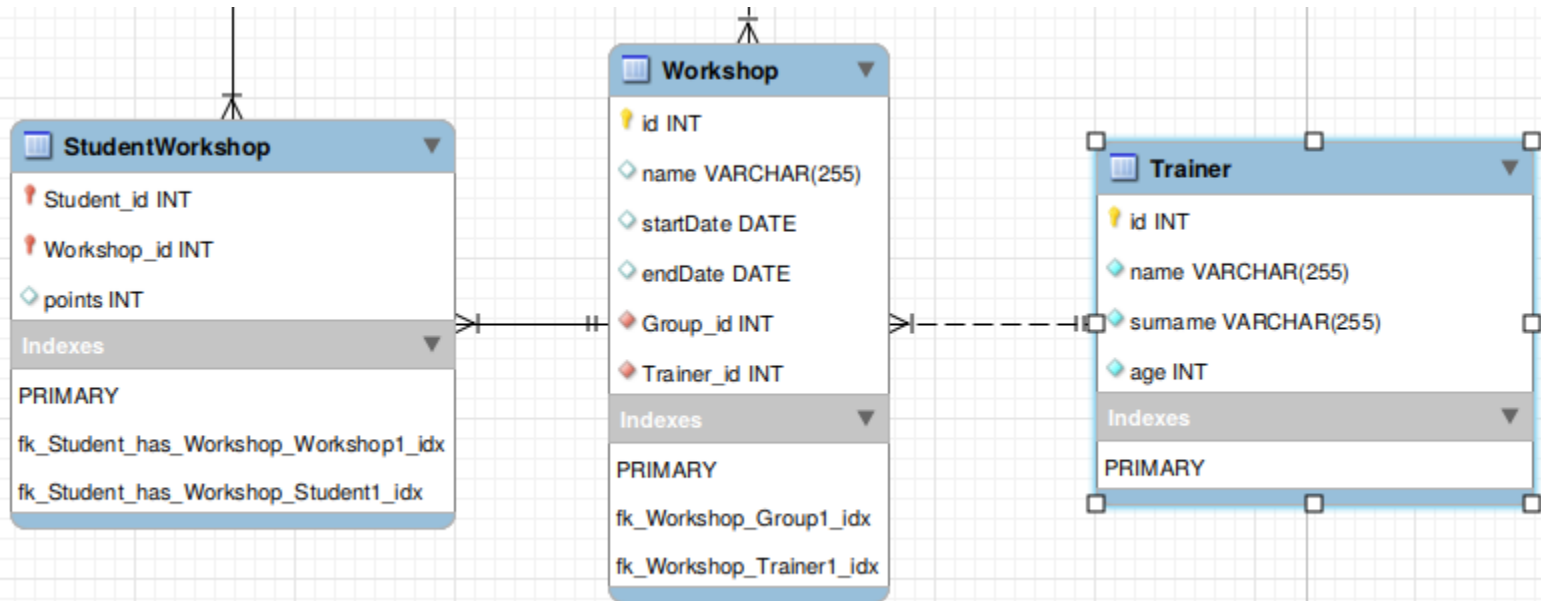
Relacje

- ✓ one-to-one
- ✓ one-to-many
- ✓ many-to-many



Klucze obce

Foreign key



- ✓ Zapewnienie spójności danych
- ✓ Lepsza wydajność przy odczycie

Klucze obce

Foreign key

- ✓

```
alter table Student
    add foreign key (Group_id)
    references `Group`(id);
```
- ✓

```
create table Student (
    id int auto_increment,
    Group_id int,
    constraint Student_Group_id
        foreign key (Group_id)
        references `Group`(id)
);
```

Zadania

- 1) Dodaj relacje w MySQL Workbench
 - a) Group – Student 1:n
 - b) Trainer – Workshop 1:n
 - c) Group – Workshop 1:n
 - d) Student – Workshop n:m
- 2) Dodaj brakujące relacje w istniejącej strukturze
- 3) Dodaj klucze obce we wszystkich relacjach
- 4) Zaktualizuj skrypty create table

Zarządzanie danymi

Przechowujemy dane w tabelach

Dodawanie danych

Insert into

- ✓ Dodanie nowych wpisów do tabeli
- ✓ `insert into <table-name> (columns, ...)
values (values, ...);`
- ✓ <https://dev.mysql.com/doc/refman/5.7/en/insert.html>

Modyfikacja danych

update

- ✓ Modyfikacja istniejących wpisów w tabeli
- ✓

```
update <table-name>  
    set assignment_list, ...  
    where conditions;
```
- ✓ UWAGA: brak lub błędny warunek where może spowodować utratę danych
- ✓ <https://dev.mysql.com/doc/refman/5.7/en/update.html>

Usuwanie danych

delete

- ✓ Usunięcie istniejących wpisów w tabeli
- ✓ `delete from <table-name>
where conditions;`
- ✓ <https://dev.mysql.com/doc/refman/5.7/en/delete.html>

Usuwanie danych

truncate

- ✓ Zapytanie służy do czyszczenia całej tabeli z danych
- ✓ `truncate <table-name>;`
- ✓ <https://dev.mysql.com/doc/refman/5.7/en/truncate-table.html>

Zadania

- 1) Dodaj przynajmniej 5 wpisów do każdej z tabel
- 2) Przetestuj jak działa:
 - a) dodawanie pojedynczego rekordu do tabeli
 - b) jednoczesne dodawanie więcej niż jednego rekordu
 - c) usuwanie wybranych rekordów vs czyszczenie tablicy
(zobacz jak zachowa się sekwencja dla ID)

Import / export

✓ export

```
mysqldump -h <host> -u <username>  
-p <db-name> > <file>.sql
```

✓ import

```
mysql -h <host> -u <username>  
-p <db-name> < <file>.sql
```

Odczytywanie danych

Warunkowe pobieranie danych

Odczytywanie danych

select

SELECT

[ALL | DISTINCT | DISTINCTROW]

select_expr [, select_expr ...]

[FROM table_references

[WHERE where_condition]

[GROUP BY {col_name | expr | position}

[ASC | DESC], ... [WITH ROLLUP]]

[HAVING where_condition]

[ORDER BY {col_name | expr | position}

[ASC | DESC], ...]

[LIMIT {[offset,] row_count | row_count OFFSET offset}]

✓ <https://dev.mysql.com/doc/refman/5.7/en/select.html>

Zadania

- 1) Wyświetl wszystkie grupy, wszystkich studentów itp
- 2) Wyświetl wszystkich studentów po 30 roku życia
- 3) Wyświetl wszystkich studentów między 30, a 50 rokiem życia
- 4) Wyświetl wszystkie unikatowe imiona studentów
- 5) Wyświetl informacje o studentach i przypisanej do nich grupie
- 6) W powyższym zapytaniu tak zmodyfikuj warunek select by wyświetlić tylko dwa pola:
 - 1) label o wartości imię – nazwisko, np. "Tomasz – Woliński"
 - 2) group o wartości nazwa grupy

Zadania

- 7) Wyświetl wszystkich studentów o imieniu Tomek
- 8) Wyświetl wszystkich studentów o imieniu zaczynającym się na 'T'
- 9) Wyświetl wszystkich studentów bez przypisanej grupy
- 10) Posortuj studentów po nazwisku
- 11) Posortuj studentów po nazwisku, a następnie po imieniu
- 12) Wyświetl tylko jeden wynik z powyższego zapytania

Zadania

- 13) Wyświetl id grupy oraz ilość studentów do niej przypisanych
- 14) Wyświetl nazwę grupy oraz ilość studentów do niej przypisanych
- 15) Wyświetl średni wiek studentów
- 16) Wyświetl najstarszego studenta
- 17) Wyświetl łączny wiek trenerów
- 18) Wyświetl zajęcia, które zaczęły się po 1 kwietnia

Stronicowanie rekordów

- ✓ `select * from Trainer order by surname limit 0, 2;`
- ✓ `select * from Trainer order by surname limit 2, 2;`

Złączenia

Łączenie danych z tabel

Łączenie tabel

join

- ✓ W celu pobrania danych jednocześnie z kilku tabel wykorzystujemy złączenia

```
select * from Workshop w  
        join Trainer t on w.trainer_id = t.id
```

- ✓ Alternatywne rozwiązanie pozwala dodać więcej niż jedną tabelę do klauzuli from

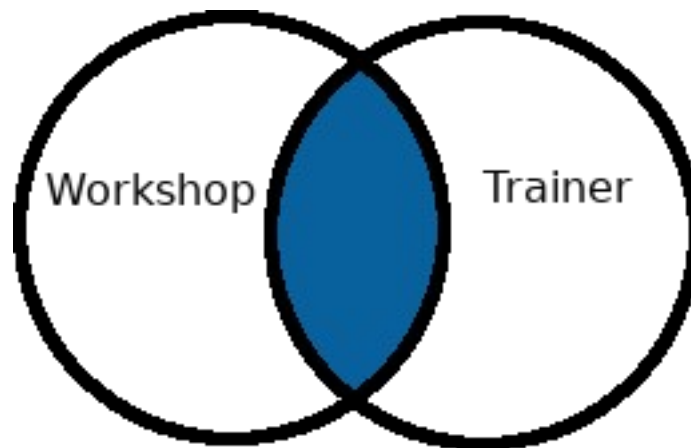
```
select * from Workshop w, Trainer t  
        where w.trainer_id = t.id
```

Łączenie tabel

inner join (lub samo join)

✓ Podstawowy typ złączenia, który zwróci tylko dane pasujące do dwóch tabel – czyli wszystkie zajęcia, które mają trenera

✓ `select * from Workshop w
inner join Trainer t
on w.trainer_id = t.id`

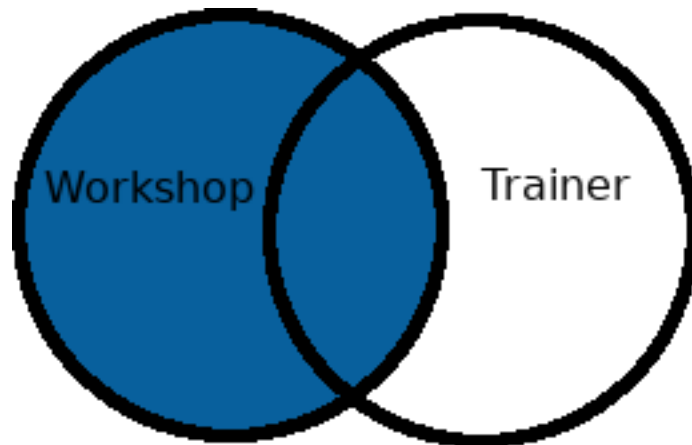


Łączenie tabel

left join

✓ Zapytanie zwróci wszystkie zajęcia, nawet jeżeli nie będzie przypisanego trenera

```
✓ select * from Workshop w  
  left join Trainer t  
  on w.trainer_id = t.id
```

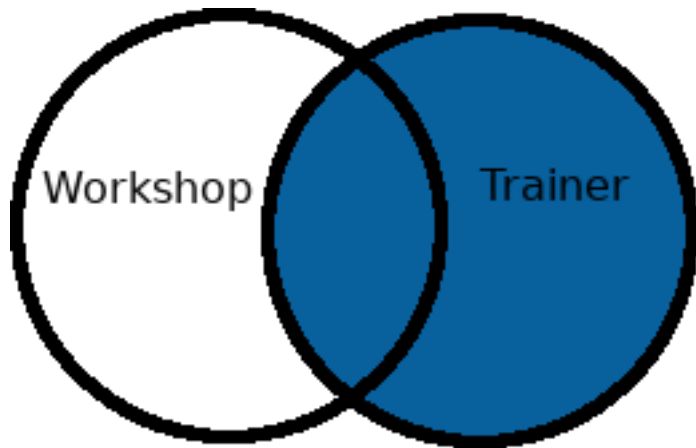


Łączenie tabel

right join

- ✓ Zapytanie zwróci wszystkich trenerów, nawet jeżeli nie mają jeszcze przypisanych zajęć
- ✓

```
select * from Workshop w  
  right join Trainer t  
    on w.trainer_id = t.id
```

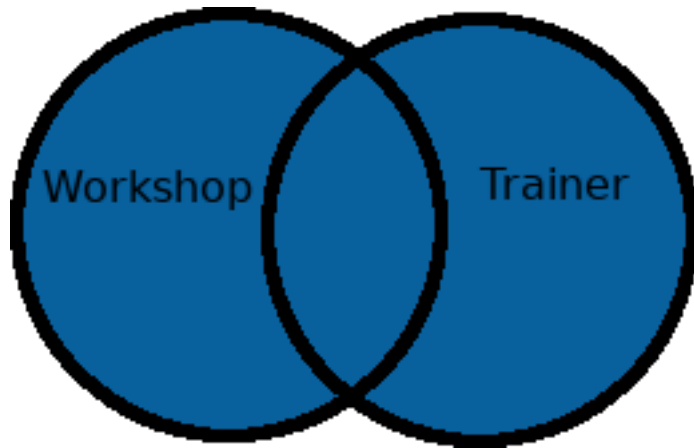


Łączenie tabel

outer join (full outer join)

- ✓ Zapytanie zwróci dane z obu tabel, również te niedopasowane
- ✓

```
select * from Workshop w  
  outer join Trainer t  
  on w.trainer_id = t.id
```



Zadania

- 1) Wyświetl wszystkich studentów oraz nazwę ich grupy. W przypadku braku grupy wyświetl napis 'brak grupy-'
- 2) Wyświetl nazwę grupy oraz ilość Studentów do niej przypisanych (uwzględnij również brak grupy, jako osobną grupę).
- 3) Powyższe zapytanie zmodyfikuj tak by wyświetlić tylko grupy z więcej niż 2 osobami
- 4) Wyświetl informacje o najliczniejszej grupie
- 5) Wyświetl studentów oraz ilość zajęć na których byli
- 6) Wyświetl studentów oraz ich najlepszy i łączny wynik z zajęć. Wyniki posortuj malejąco po łącznym wyniku.
- 7) Wyświetl studentów, którzy byli na wszystkich zajęciach
- 8) Posortuj zajęcia po czasie ich trwania

Indexy

Przyspieszamy bazę danych

Tworzenie/kasowanie indexów

- ✓ `create index index_name
on table_name (column1, column2, ...);`
- ✓ `drop index index_name on table_name;`
- ✓ `explain <query>`

Użytkownicy i uprawnienia

Zarządzanie dostępem do bazy danych

Zarządzanie kontami

uprawnienia

- ✓ `create user newuser`
 `identified by 'password';`
- ✓ `show grants for newuser;`
- ✓ `grant select on * to newuser;`
- ✓ `revoke select on * from newuser;`
- ✓ `flush privileges;`
- ✓ `drop user newuser;`

Zadania

- 1) Dodaj nowego użytkownika, który będzie miał uprawnienia tylko do odczytu tabeli *Student*;

Dodatkowe zadania

- 1) Problem magic numbers w skryptach
- 2) Zabezpieczenie skryptów przed wielokrotnym uruchamianiem (Idempotentność)
- 3) Problem zaokrągleń float/decimal
- 4) Widoki bazodanowe



Thanks!!

Pytania?

Znajdziesz mnie na:
StormIT.pl

