

Laboratorium z Podstaw Baz Danych

Data wykonania ćwiczenia: 16.05.2024

Przygotowali: Michał Czyż, Dawid Głąb

Grupa 6, Sekcja 13

I. Instalacja

1. - 3. Wprowadzenie

Pobrano, rozpakowano i uruchomiono serwer MySQL. Skrypt `run_server.bat` uruchamia serwer bazy danych mysql z parametrami zdefiniowanymi w pliku `mysql.ini`.

4. Bazy danych znajdujących się na serwerze

```
mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| uczelnia           |
| university         |
+-----+
4 rows in set (0.00 sec)
```

Na serwerze znajdują się cztery bazy danych. `uczelnia` oraz `university` to bazy z danymi zadania. Natomiast `information_schema` oraz `mysql` to bazy z konfiguracją i parametrami mysql.

5. Przeglądanie bazy danych

a. Połączenie z serwerem

```
C:\mysql\bin> .\mysql.exe -u root -p --port 6033
```

b. Uaktywnienie bazy danych uczelnia

```
mysql> use uczelnia
Database changed
```

c. analiza struktury tabeli *przydziały*

```
mysql> describe przydzialy
-> ;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| NR_PRAC    | int(11)   | NO   |     | NULL    |       |
| NR_TEM      | int(11)   | NO   |     | NULL    |       |
| KOD_FUNKCJI | char(3)   | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

d. wykonano polecenie `select * from przydzialy`

```
mysql> select * from przydzialy
-> ;
+-----+-----+-----+
| NR_PRAC | NR_TEM | KOD_FUNKCJI |
+-----+-----+-----+
| 1       | 1      | PRG          |
| 1       | 4      | ANL          |
| 1       | 5      | ADM          |
| 1       | 10     | ANL          |
| 1       | 11     | ADM          |
| 1       | 11     | PRG          |
| 1       | 11     | PRJ          |
| 1       | 13     | PRG          |
| 1       | 16     | ANL          |
| 1       | 21     | TST          |
| 1       | 25     | PRJ          |
| 1       | 33     | KRW          |
| 1       | 101    | ANL          |
| 1       | 101    | PRJ          |
| 1       | 103    | PRG          |
| 1       | 104    | ANL          |
| 1       | 104    | PRG          |
| 1       | 106    | KRW          |
| 1       | 107    | ANL          |
| 1       | 107    | PRJ          |
| 2       | 2      | PRJ          |
| 2       | 3      | PRJ          |
| 2       | 13     | PRG          |
| 2       | 15     | ANL          |
| 2       | 16     | PRG          |
| 2       | 18     | PRJ          |
| 2       | 101    | PRG          |
| 2       | 104    | PRG          |
| 2       | 104    | PRJ          |
| 2       | 105    | PRG          |
| 3       | 31     | ADM          |
| 3       | 31     | ANL          |
+-----+-----+-----+
```

	3	31	PRG
	3	31	PRJ
	4	3	TST
	4	5	PRJ
	4	6	PRG
	4	11	ADM
	4	25	ANL
	4	106	ADM
	4	106	PRJ
	5	2	PRJ
	5	3	PRJ
	5	13	PRG
	5	18	ADM
	5	18	PRG
	5	103	PRJ
	5	106	PRG
	6	5	PRG
	6	8	ANL
	6	8	TST
	6	10	PRJ
	6	17	PRJ
	6	21	PRG
	6	24	TST
	7	39	PRG
	7	39	TST
	8	6	PRG
	8	8	ADM
	8	13	PRG
	8	14	ANL
	8	16	PRJ
	8	22	PRJ
	8	28	PRG
	9	13	PRJ
	9	14	ANL
	9	18	PRG
	9	18	TST
	9	28	PRG
	9	104	ADM
	9	104	PRG
	9	107	PRG
	10	24	KRW
	10	101	KRW
	11	2	PRJ
	11	15	KRW
	11	18	ANL
	11	21	KRW
	11	102	ANL
	11	104	PRG
	11	105	PRG
	11	106	ADM
	12	3	ADM
	12	3	PRG
	12	4	PRG
	12	20	ANL

	12	20	PRG
	12	31	PRG
	12	40	PRG
	12	105	PRJ
	13	2	PRJ
	13	10	PRJ
	13	17	PRG
	13	21	PRG
	13	22	ANL
	13	28	ANL
	13	104	TST
	14	40	ANL
	15	6	PRG
	15	6	PRJ
	15	15	ANL
	15	20	ANL
	15	22	ANL
	15	31	ANL
	15	104	ANL
	15	106	PRG
	16	5	PRG
	16	10	PRJ
	16	13	PRG
	16	15	TST
	16	17	PRJ
	16	22	TST
	16	28	PRJ
	18	6	ADM
	18	8	PRG
	18	14	ADM
	18	17	PRG
	18	21	ANL
	18	32	KRW
	18	103	ADM
	18	105	PRG
	19	8	PRG
	19	18	PRG
	19	106	TST
	19	107	ANL
	20	3	ANL
	20	4	TST
	20	5	PRG
	20	6	PRG
	20	20	PRJ
	20	30	PRJ
	20	39	ANL
	20	42	PRG
	20	104	ANL
	20	106	PRJ
	21	8	KRW
	21	30	KRW
	21	42	ANL
	21	44	KRW
	22	3	ANL

22	4	PRG
22	11	PRG
22	22	PRG
22	25	PRG
22	103	PRG
23	14	ANL
23	20	PRG
23	28	ANL
23	28	PRJ
23	42	KRW
23	43	KRW
23	101	PRG
23	104	ANL
23	104	PRJ
23	106	PRG
25	1	PRG
25	1	PRJ
25	11	PRG
25	16	KRW
25	20	ANL
25	22	PRG
25	24	ADM
25	39	PRJ
25	102	KRW
26	2	PRJ
26	6	PRG
26	10	KRW
26	17	PRG
26	17	TST
26	24	ADM
26	103	PRG
26	106	PRG
27	4	ANL
27	15	PRG
27	15	TST
27	18	ADM
27	22	PRG
27	24	ANL
27	25	KRW
29	6	KRW
29	8	PRG
29	11	KRW
29	16	PRG
29	24	PRJ
29	25	PRJ
29	28	KRW
29	102	PRG
29	104	PRG
30	4	TST
30	10	PRG
30	22	PRG
30	24	ADM
30	24	PRG
30	40	ANL

30	40	PRG
30	104	TST
30	105	PRG
30	105	TST
31	2	PRG
31	13	KRW
31	17	PRJ
31	22	KRW
31	31	KRW
31	100	KRW
32	1	KRW
32	2	KRW
32	5	KRW
32	10	PRG
32	18	KRW
32	22	PRG
32	39	KRW
32	45	KRW
32	107	KRW
33	1	PRJ
33	8	PRG
33	18	PRG
33	36	KRW
33	103	KRW
33	105	KRW
34	3	KRW
34	14	KRW
34	20	KRW
34	22	PRJ
34	30	TST
35	4	KRW
35	10	PRG
35	17	KRW
35	104	KRW
36	5	PRG
36	10	ANL
36	34	KRW
36	37	KRW
36	40	PRG
36	40	PRJ
37	10	ANL
37	35	KRW
37	38	KRW
37	40	ANL
37	101	PRG
38	40	KRW
38	41	KRW
40	31	PRG
41	31	ADM

+-----+-----+-----+
243 rows in set (0.00 sec)

6. Tworzenie bazy danych i analiza stanu serwera z linii poleceń

a. Zmieniono hasło użytkownika *root* na *root_sql*

```
C:\mysql\bin>mysqladmin -u root -pala password root_sql --port 6033
```

b. Uruchomiono program

```
C:\mysql\bin>mysql -u root -proot_sql --port 6033
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7 to server version: 5.1.34-community

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

c. Wykonano następujące polecenia

```
mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| uczelnia           |
| university         |
+-----+
4 rows in set (0.00 sec)

mysql> show processlist;
+----+-----+-----+-----+-----+-----+-----+-----+
---+
| Id | User | Host                | db      | Command | Time | State | Info
|
+----+-----+-----+-----+-----+-----+-----+-----+
---+
| 1  | root | localhost:50026     | uczelnia | Sleep   | 1029 |      | NULL
|
| 7  | root | localhost:62861     | NULL    | Query   | 0    | NULL | show
processlist |
+----+-----+-----+-----+-----+-----+-----+-----+
---+
2 rows in set (0.00 sec)

mysql> use labbd
ERROR 1049 (42000): Unknown database 'labbd'
```

```
mysql> show create table pracownicy;  
ERROR 1046 (3D000): No database selected
```

d. Komentarz

Polecenie `show databases;` pokazuje listę wszystkich baz danych dostępnych na serwerze mysql.

Polecenie `show processlist;` pokazuje listę aktualnie wykonujących się poleceń na serwerze

Polecenie `use labbd;` powinno wybrać i uaktywnić podaną bazę danych, ale taka baza nie istnieje na serwerze.

Polecenie `show create table pracownicy` powinno wyświetlić wynik tworzenia nowej tabeli, lecz baza nie została wybrana (poprzednie polecenie nie powiodło się, ponieważ nie znaleziono takiej bazy).

e. Utworzono bazę danych `labbd`, uaktywniono ją oraz utworzono w niej obiekty

```
mysql> create database labbd;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> use labbd;  
Database changed  
  
mysql> \. ../labbd.sql  
Query OK, 0 rows affected (0.01 sec)  
...  
Query OK, 0 rows affected (0.01 sec)
```

f. Wyświetlono zawartość bazy danych

Polecenie `show tables;` wyświetla wszystkie tabele znajdujące się w tej bazie danych, natomiast `show columns from funkcje;` pokazuje wszystkie kolumny z jakich składa się dana tabela (tutaj `funkcje`) oraz ich parametry (typ, czy jest kluczem, właściwości klucza, domyślna wartość, itd).

```
mysql> show tables  
-> ;  
+-----+  
| Tables_in_labbd |  
+-----+  
| funkcje          |  
| instytucje       |  
| kierunki         |  
| oceny            |  
| pracownicy       |  
| przedmioty       |  
| przydzialy       |  
| rozklady         |  
| sale             |  
| studenci         |
```



```
| tematy          |
| typy_przedmiotow |
| wypłaty        |
| zespoły        |
+-----+
14 rows in set (0.01 sec)

mysql> show columns from funkcje
-> ;
+-----+-----+-----+-----+-----+-----+
| Field          | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| KOD_FUNKCJI    | char(3)   | NO   |     | NULL    |       |
| NAZWA_FUNKCJI  | char(30)  | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

II. Operacje na bazie danych

1. Ładowanie danych

a. Wykorzystując program *mysql* utworzono bazę danych *Lab_sql*

```
mysql> create database Lab_sql;
Query OK, 1 row affected (0.00 sec)
```

Następnie aktywowano utworzoną bazę danych.

```
mysql> use Lab_sql;
Database changed
```

b. Uruchomiono skrypt tworzący tablicę *casdat*

```
mysql> \. ../crcasdat.sql
Query OK, 0 rows affected (0.01 sec)
```

Tablice została utworzona. Czas wykonania zajął niecałe 0.01 sekundy.

c. Uruchomiono skrypt ładujący dane

```
mysql> \. ../casdat_dump.sql
Query OK, 1 row affected (0.00 sec)
...
Query OK, 1 row affected (0.00 sec)
```

d. Operacja ładowania danych zajęła 01:37.87

e. Zapisano dane z tabeli do pliku `casdat.dump`

```
mysql> select * from casdat into outfile 'casdat.dump'
-> ;
Query OK, 76211 rows affected (0.16 sec)
```

f. usunięto wszystkie rekordy z tabeli

```
mysql> truncate casdat;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> select * from casdat
-> ;
Empty set (0.00 sec)
```

g. ponownie załadowano dane z utworzonego pliku z danymi

```
mysql> load data infile 'casdat.dump' into table casdat;
Query OK, 76211 rows affected (0.29 sec)
Records: 76211 Deleted: 0 Skipped: 0 Warnings: 0
```

h. Komentarz

Wykonane kroki pokazały że można wyeksportować wynik zapytania do pliku i w przypadku utraty albo wyczyszczenia tabeli można przywrócić jej zawartość z kopii zapasowej.

2. Typy wyliczeniowe

a. uaktywniono bazę danych Lab_sql i utworzona tabelę `dyżury` z użyciem typu `ENUM`

```
mysql> use Lab_sql;
Database changed

mysql> create table dyzury(
-> ido int,
-> dzien enum
('poniedzialek','wtorek','sroda','czwartek','piatek','sobota','niedziela'));
Query OK, 0 rows affected (0.01 sec)
```

b. do tabeli wprowadzono dwa rekordy z dowolnie wybranymi dniami oraz jeden z poza zakresu

```
mysql> insert into dzury values (1, 'sobota');
Query OK, 1 row affected (0.00 sec)

mysql> insert into dzury values (2, 'piatek');
Query OK, 1 row affected (0.00 sec)

mysql> insert into dzury values (3, 'aaaa');
Query OK, 1 row affected, 1 warning (0.00 sec)
```

Wyświetla się warning, że coś mogło pójść nie tak

c. wprowadzono rekord z dwoma dniami

```
mysql> insert into dzury values (4, 'wtorek,sroda');
Query OK, 1 row affected, 1 warning (0.00 sec)
```

Podobnie jak wcześniej pojawił się warning

d. wprowadzono parę wartości

```
mysql> insert into dzury values (5,5);
Query OK, 1 row affected (0.00 sec)
```

e. efekt wykonanych wstawień

```
mysql> select * from dzury;
+-----+-----+
| ido | dzien |
+-----+-----+
| 1 | sobota |
| 2 | piatek |
| 3 |      |
| 4 |      |
| 5 | piatek |
+-----+-----+
5 rows in set (0.00 sec)
```

f. Komentarz

Dokumentacja na stronie dev.mysql.com wskazuje, że wartości wstawione do typu `ENUM` mogą być puste "" albo `NULL`, jeżeli wstawiona wartość jest z poza zakresu lub nieprawidłowa.

g. Utworzono drugą tabelę

```
mysql> create table dzury_set(  
  -> ido int,  
  -> dzien set  
  ('poniedzialek','wtorek','sroda','czwartek','piatek','sobota','niedziela'));  
Query OK, 0 rows affected (0.02 sec)
```

h. Powtórzono operacje dla nowej tabeli

```
mysql> insert into dzury_set values (1, 'wtorek');  
Query OK, 1 row affected (0.00 sec)  
  
mysql> insert into dzury_set values (2, 'sroda');  
Query OK, 1 row affected (0.00 sec)  
  
mysql> insert into dzury_set values (3, 'aaa');  
Query OK, 1 row affected, 1 warning (0.00 sec)  
  
mysql> insert into dzury_set values (4, 'wtorek,sroda');  
Query OK, 1 row affected (0.00 sec)  
  
mysql> insert into dzury_set values (5,5);  
Query OK, 1 row affected (0.00 sec)
```

i. wprowadzono dodatkowo rekord:

```
mysql> insert into dzury_set values (13, 'poniedzialek,aaa');  
Query OK, 1 row affected, 1 warning (0.00 sec)
```

j. wyświetlono zawartość tabeli

```
mysql> select * from dzury_set;  
+-----+-----+  
| ido | dzien                |  
+-----+-----+  
| 1   | wtorek               |  
| 2   | sroda                |  
| 3   |                      |  
| 4   | wtorek,sroda        |  
| 5   | poniedzialek,sroda  |  
| 13  | poniedzialek        |  
+-----+-----+  
6 rows in set (0.00 sec)
```

k. wykonano polecenie

```
mysql> update dyzury_set set dzien='sobota' where find_in_set('wtorek',dzien);
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
mysql> select * from dyzury_set
-> ;
+-----+-----+
| ido | dzien |
+-----+-----+
| 1 | sobota |
| 2 | sroda |
| 3 |  |
| 4 | sobota |
| 5 | poniedzialek,sroda |
| 13 | poniedzialek |
+-----+-----+
6 rows in set (0.00 sec)
```

I. Komentarz

Ostatnim poleceniem (update) zmieniono wszystkie wtorki na soboty. Jeśli chodzi o poprzednie polecenia, to set akceptuje wartości wymienione po przecinku. Wpisywanie liczb jak np. 5,5 wciąż zostanie odrzucone. Zamiana wtorków na soboty zamienia także wszystko na sobotę tam, gdzie jest więcej niż jeden dzień. Wpisanie trzech liter aaa jest traktowane jako liczba i wstawiany jest dzień tygodnia.

III. Silniki (storage engine)

- 1. Sprawdzenie z jakiego silnika korzystają tablice w bazie labbd

```
mysql> show table status where name='studenci';
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+
| Name      | Engine | Version | Row_format | Rows | Avg_row_length | Data_length |
Max_data_length | Index_length | Data_free | Auto_increment | Create_time
| Update_time      | Check_time | Collation      | Checksum | Create_options
| Comment |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+
| studenci | MyISAM | 10 | Fixed | 66 | 34 | 2244 |
9570149208162303 | 1024 | 0 | NULL | 2024-05-16 12:07:12
| 2024-05-16 12:07:13 | NULL | latin1_swedish_ci | NULL |
|  |
```

```
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+
--++-----+-----+-----+-----+-----+-----+-----+
---++-----+
1 row in set (0.00 sec) [MyIsam]
```

Używany silnik to MyIsam

2. Pliki stworzone dla każdej tablicy mają rozszerzenia:

.frm .MYD .MYI

3. Zmiana silnika tablicy na InnoDB

```
mysql> alter table studenci ENGINE = InnoDB;
Query OK, 66 rows affected (0.03 sec)
Records: 66 Duplicates: 0 Warnings: 0
```

4. Pliki w katalogu uległy zmianie i pozostały tylko pliki z rozszerzeniem:

.frm

5. Wykonanie ciągu instrukcji

```
mysql> begin;
Query OK, 0 rows affected (0.00 sec)

mysql> update pracownicy set plec='K' where nr_prac=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update studenci set plec='K' where nr_stud=1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select plec from pracownicy where nr_prac = 1;
+-----+
| plec |
+-----+
| K    |
+-----+
1 row in set (0.00 sec)

mysql> select plec from studenci where nr_stud = 1;
+-----+
| plec |
+-----+
| K    |
+-----+
1 row in set (0.00 sec)
```

```
mysql> rollback;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> select plec from pracownicy where nr_prac = 1;
+-----+
| plec |
+-----+
| K    |
+-----+
1 row in set (0.00 sec)

mysql> select plec from studenci where nr_stud = 1;
+-----+
| plec |
+-----+
| M    |
+-----+
1 row in set (0.00 sec)
```

Rollback transakcji nie działa dla silnika MyISAM, natomiast działa dla InnoDB

10. silnik Memory

```
mysql> alter table studenci ENGINE=Memory;
Query OK, 66 rows affected (0.02 sec)
Records: 66 Duplicates: 0 Warnings: 0

mysql> select plec from studenci where nr_stud=5;
+-----+
| plec |
+-----+
| K    |
+-----+
1 row in set (0.00 sec)

mysql> begin;
Query OK, 0 rows affected (0.00 sec)

mysql> update studenci set plec='M' where nr_stud=5;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select plec from studenci where nr_stud=5;
+-----+
| plec |
+-----+
| M    |
+-----+
1 row in set (0.00 sec)

mysql> rollback;
```

```
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> select plec from studenci where nr_stud=5;
+-----+
| plec |
+-----+
| M    |
+-----+
1 row in set (0.00 sec)
```

Wykonano modyfikację danych w transakcji i następnie wykonano polecenie rollback. Dane nie zostały przywrócone do wartości z przed transakcji, zatem silnik Memory nie obsługuje transakcji.

```
C:\mysql\bin>mysql -u root -paaa --port 6033
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 5.1.34-community

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use labbd;
Database changed
mysql> select * from studenci;
Empty set (0.00 sec)
```

Zawartość tabeli studenci jest pusta ponieważ silnik Memory przechowuje zawartość tabeli tylko w czasie działania serwera baz danych.

IV. Optymalizacja zapytań

Zadanie 1. Numery CAS odczynników które mają więcej właściwości SENT niż RENT

1. Zapytanie 1.

```
mysql> select c1.cas, count(distinct c1.data) from casdat c1
-> where c1.type='sent'
-> group by c1.cas
-> having count(distinct c1.data)>
-> (select count(distinct c2.data) from casdat c2 where c2.type='rent' and
-> c2.cas=c1.cas);
+-----+-----+
| cas          | count(distinct c1.data) |
+-----+-----+
| 300-57-2     | 3 |
| 300-85-6     | 5 |
| 3001-72-7    | 5 |
| 302-79-4     | 3 |
```


303-07-1	2	
3030-47-5	5	
3040-44-6	2	
3060-50-2	1	
3085-30-1	4	
312-94-7	5	
3130-87-8	3	
3132-64-7	6	
3140-93-0	2	
3173-53-3	6	
3179-63-3	5	
3209-22-1	2	
321-14-2	2	
32315-10-9	6	
3234-28-4	2	
3264-82-2	2	
32740-79-7	2	
3277-26-7	3	
3279-26-3	5	
3282-30-2	6	
329-15-7	5	
329-98-6	5	
32916-07-7	5	
331-39-5	2	
333-27-7	5	
33462-81-6	5	
3347-22-6	8	
335-64-8	5	
335-67-1	5	
336-59-4	5	
33678-01-2	5	
33725-74-5	5	
3375-31-3	3	
3378-72-1	5	
33966-50-6	9	
3399-73-3	5	
341-02-6	5	
343-94-2	3	
3433-80-5	5	
3438-46-8	3	
34451-19-9	2	
345-35-7	5	
3453-83-6	5	
34560-16-2	1	
34822-90-7	4	
34837-55-3	4	
350-46-9	2	
350-70-9	5	
3518-65-8	5	
352-32-9	3	
352-67-0	2	
352-70-5	3	
352-93-2	4	
353-42-4	6	

3536-49-0	3	
354-58-5	4	
356-27-4	2	
357-57-3	3	
3571-74-2	4	
358-23-6	5	
3585-33-9	8	
3587-60-8	6	
35963-20-3	6	
36177-92-1	5	
36239-09-5	5	
363-72-4	2	
364-76-1	2	
366-18-7	3	
367-11-3	3	
367-21-5	3	
367-29-3	4	
36768-62-4	5	
368-39-8	5	
3680-02-2	4	
36805-97-7	2	
3688-92-4	4	
3710-30-3	2	
372-18-9	2	
3721-95-7	5	
3724-43-4	5	
37267-86-0	5	
3731-52-0	5	
37360-75-1	2	
375-16-6	5	
375-22-4	5	
379-52-2	4	
38053-91-7	4	
38078-09-0	6	
381-73-7	6	
3811-04-9	4	
383-63-1	7	
38444-13-2	2	
3878-44-2	4	
3886-69-9	6	
38870-89-2	5	
38966-21-1	2	
3906-55-6	2	
39098-97-0	5	
39178-35-3	5	
3926-62-3	3	
39262-22-1	5	
393-52-2	5	
3931-89-3	6	
394-47-8	2	
39409-82-0	3	
39416-48-3	5	
39433-54-0	1	
39450-01-6	5	

```

| 3952-78-1 | | 3 |
| 3958-60-9 | | 5 |
| 39637-74-6 | | 5 |
+-----+
115 rows in set (3.32 sec)

```

2. Zapytanie 2.

```

mysql> select c1.cas from
-> casdat c1 join casdat c2 on c1.cas=c2.cas
-> where c1.type='sent' and c2.type='rent'
-> group by c1.cas
-> having count(distinct c1.data)>count(distinct c2.data);

```

```

+-----+
| cas      |
+-----+
| 300-57-2 |
| 300-85-6 |
| 3001-72-7 |
| 302-79-4 |
| 3030-47-5 |
| 3040-44-6 |
| 3085-30-1 |
| 312-94-7 |
| 3132-64-7 |
| 3140-93-0 |
| 3173-53-3 |
| 3179-63-3 |
| 3209-22-1 |
| 321-14-2 |
| 32315-10-9 |
| 3264-82-2 |
| 32740-79-7 |
| 3277-26-7 |
| 3279-26-3 |
| 3282-30-2 |
| 329-15-7 |
| 329-98-6 |
| 32916-07-7 |
| 331-39-5 |
| 333-27-7 |
| 33462-81-6 |
| 3347-22-6 |
| 335-64-8 |
| 335-67-1 |
| 336-59-4 |
| 33678-01-2 |
| 33725-74-5 |
| 3375-31-3 |
| 3378-72-1 |
| 33966-50-6 |
| 3399-73-3 |

```

341-02-6	
3433-80-5	
3438-46-8	
345-35-7	
3453-83-6	
34822-90-7	
34837-55-3	
350-70-9	
3518-65-8	
352-32-9	
352-67-0	
352-70-5	
352-93-2	
353-42-4	
3536-49-0	
354-58-5	
356-27-4	
357-57-3	
3571-74-2	
358-23-6	
3585-33-9	
3587-60-8	
35963-20-3	
36177-92-1	
36239-09-5	
363-72-4	
364-76-1	
366-18-7	
367-11-3	
367-21-5	
367-29-3	
36768-62-4	
368-39-8	
3680-02-2	
36805-97-7	
3688-92-4	
3710-30-3	
372-18-9	
3721-95-7	
3724-43-4	
37267-86-0	
3731-52-0	
375-16-6	
375-22-4	
379-52-2	
38053-91-7	
38078-09-0	
381-73-7	
3811-04-9	
383-63-1	
38444-13-2	
3878-44-2	
3886-69-9	
38870-89-2	

```
| 3906-55-6 |
| 39098-97-0 |
| 39178-35-3 |
| 3926-62-3 |
| 39262-22-1 |
| 393-52-2 |
| 3931-89-3 |
| 39416-48-3 |
| 39450-01-6 |
| 3958-60-9 |
| 39637-74-6 |
+-----+
101 rows in set (0.06 sec)
```

3. Zapytanie drugie wykonuje się znacznie szybciej od pierwszego zapytania. Jest to związane z tym, że zapytanie pierwsze ma dodatkowo podzapytanie co znacznie spowalnia czas.

4. Indeks

```
mysql> create index cas_idx on casdat (cas);
Query OK, 4821 rows affected (0.02 sec)
Records: 4821 Duplicates: 0 Warnings: 0
```

5. Ponowne uruchomienie zapytań

```
mysql> select c1.cas, count(distinct c1.data) from casdat c1
-> where c1.type='sent'
-> group by c1.cas
-> having count(distinct c1.data)>
-> (select count(distinct c2.data) from casdat c2 where c2.type='rent' and
-> c2.cas=c1.cas);
```

cas	count(distinct c1.data)
300-57-2	3
300-85-6	5
3001-72-7	5
302-79-4	3
303-07-1	2
3030-47-5	5
3040-44-6	2
3060-50-2	1
3085-30-1	4
312-94-7	5
3130-87-8	3
3132-64-7	6
3140-93-0	2
3173-53-3	6
3179-63-3	5

3209-22-1	2
321-14-2	2
32315-10-9	6
3234-28-4	2
3264-82-2	2
32740-79-7	2
3277-26-7	3
3279-26-3	5
3282-30-2	6
329-15-7	5
329-98-6	5
32916-07-7	5
331-39-5	2
333-27-7	5
33462-81-6	5
3347-22-6	8
335-64-8	5
335-67-1	5
336-59-4	5
33678-01-2	5
33725-74-5	5
3375-31-3	3
3378-72-1	5
33966-50-6	9
3399-73-3	5
341-02-6	5
343-94-2	3
3433-80-5	5
3438-46-8	3
34451-19-9	2
345-35-7	5
3453-83-6	5
34560-16-2	1
34822-90-7	4
34837-55-3	4
350-46-9	2
350-70-9	5
3518-65-8	5
352-32-9	3
352-67-0	2
352-70-5	3
352-93-2	4
353-42-4	6
3536-49-0	3
354-58-5	4
356-27-4	2
357-57-3	3
3571-74-2	4
358-23-6	5
3585-33-9	8
3587-60-8	6
35963-20-3	6
36177-92-1	5
36239-09-5	5

363-72-4	2
364-76-1	2
366-18-7	3
367-11-3	3
367-21-5	3
367-29-3	4
36768-62-4	5
368-39-8	5
3680-02-2	4
36805-97-7	2
3688-92-4	4
3710-30-3	2
372-18-9	2
3721-95-7	5
3724-43-4	5
37267-86-0	5
3731-52-0	5
37360-75-1	2
375-16-6	5
375-22-4	5
379-52-2	4
38053-91-7	4
38078-09-0	6
381-73-7	6
3811-04-9	4
383-63-1	7
38444-13-2	2
3878-44-2	4
3886-69-9	6
38870-89-2	5
38966-21-1	2
3906-55-6	2
39098-97-0	5
39178-35-3	5
3926-62-3	3
39262-22-1	5
393-52-2	5
3931-89-3	6
394-47-8	2
39409-82-0	3
39416-48-3	5
39433-54-0	1
39450-01-6	5
3952-78-1	3
3958-60-9	5
39637-74-6	5

+-----+

115 rows in set (0.03 sec)

```
mysql> select c1.cas from
-> casdat c1 join casdat c2 on c1.cas=c2.cas
```

```

-> where c1.type='sent' and c2.type='rent'
-> group by c1.cas
-> having count(distinct c1.data)>count(distinct c2.data);

```

```

+-----+
| cas      |
+-----+
| 300-57-2 |
| 300-85-6 |
| 3001-72-7 |
| 302-79-4 |
| 3030-47-5 |
| 3040-44-6 |
| 3085-30-1 |
| 312-94-7 |
| 3132-64-7 |
| 3140-93-0 |
| 3173-53-3 |
| 3179-63-3 |
| 3209-22-1 |
| 321-14-2 |
| 32315-10-9 |
| 3264-82-2 |
| 32740-79-7 |
| 3277-26-7 |
| 3279-26-3 |
| 3282-30-2 |
| 329-15-7 |
| 329-98-6 |
| 32916-07-7 |
| 331-39-5 |
| 333-27-7 |
| 33462-81-6 |
| 3347-22-6 |
| 335-64-8 |
| 335-67-1 |
| 336-59-4 |
| 33678-01-2 |
| 33725-74-5 |
| 3375-31-3 |
| 3378-72-1 |
| 33966-50-6 |
| 3399-73-3 |
| 341-02-6 |
| 3433-80-5 |
| 3438-46-8 |
| 345-35-7 |
| 3453-83-6 |
| 34822-90-7 |
| 34837-55-3 |
| 350-70-9 |
| 3518-65-8 |
| 352-32-9 |
| 352-67-0 |
| 352-70-5 |

```


	352-93-2	
	353-42-4	
	3536-49-0	
	354-58-5	
	356-27-4	
	357-57-3	
	3571-74-2	
	358-23-6	
	3585-33-9	
	3587-60-8	
	35963-20-3	
	36177-92-1	
	36239-09-5	
	363-72-4	
	364-76-1	
	366-18-7	
	367-11-3	
	367-21-5	
	367-29-3	
	36768-62-4	
	368-39-8	
	3680-02-2	
	36805-97-7	
	3688-92-4	
	3710-30-3	
	372-18-9	
	3721-95-7	
	3724-43-4	
	37267-86-0	
	3731-52-0	
	375-16-6	
	375-22-4	
	379-52-2	
	38053-91-7	
	38078-09-0	
	381-73-7	
	3811-04-9	
	383-63-1	
	38444-13-2	
	3878-44-2	
	3886-69-9	
	38870-89-2	
	3906-55-6	
	39098-97-0	
	39178-35-3	
	3926-62-3	
	39262-22-1	
	393-52-2	
	3931-89-3	
	39416-48-3	
	39450-01-6	
	3958-60-9	
	39637-74-6	

```
+-----+
101 rows in set (0.09 sec)
```

5. Czasy wykonania zmieniły się. Oba zapytania wykonują się teraz w mniej niż 0.1 sekundy.
6. Drugie zapytanie zwraca mniej odczytników niż pierwsze ponieważ jest pomijane uwzględnienie gdzie SENT istnieje ale RENT nie istnieje (czyli jest NULL)
7. Modyfikacja zapytania

```
select c1.cas
from casdat c1
left join casdat c2 on c1.cas=c2.cas and c2.type = 'rent'
where c1.type='sent'
group by c1.cas
having count(distinct c1.data) > coalesce(count(distinct c2.data),0);
```

funkcja COALESCE() zwraca pierwszą wartość, która nie jest zerem.

8. Czasy wykonania zapytania: z indeksem: 115 rows in set (0.09 sec) bez indeksu: 115 rows in set (11.44 sec)

Indeksy znacznie poprawiają czas wyszukiwania.

Zadanie 2. Znajdź, które odczytniki mają parametry wszystkich typów

```
mysql> select cas,type,data from casdat c1
-> where
-> not exists(
-> select distinct type from casdat c2 where type not in
-> (select type from casdat c3 where c3.cas=c1.cas)
-> );
Aborted
```

Zajęło ponad 5 minut i wciąż się nie skończyło

```
mysql> select cas, count(distinct type)
-> from casdat
-> group by cas
-> having count(distinct type)=(select count(distinct type) from casdat);
Empty set (0.01 sec)
```

Zapytanie drugie wykonało się prawie od razu.

Różnica czasowa ma ogromne znaczenie jeżeli chodzi o czas otrzymania odpowiedzi z zapytania.

Zadanie 3. Podaj nazwy odczynników, których własność typu "rent" zawiera liczbę 12.

```
mysql> select c0.cas,c0.data,c.data from casdat c0, casdat c where c0.type='name'
and
-> c0.cas = c.cas and c.type='rent' and c.data like '%12%' group by
-> c0.cas,c0.data,c.data order by c0.cas;
Empty set (0.07 sec)
```

Czas wykonania był poniżej 0.1 sekundy.

2. Stworzono indeks na pole type

```
mysql> create index ind_type on casdat (type asc);
Aborted
```

3. Usuń istniejące indeksy i stwórz indeks ind_tcd

```
mysql> create index ind_tcd on casdat (type asc, cas asc, data asc);
Aborted
```

4. Usuń istniejące indeksy i stwórz indeks na pola data, type, cas.

```
mysql> create index ind_dtc on casdat(data asc, type asc, cas asc);
Aborted
```

Żaden z indeksów nie utworzył się, choć czekano ponad 5 minut na wykonanie