

**Praca z bazą danych MySQL
wersja 2.0**

Nr ćwiczenia:	4
Temat:	Polecenie SELECT, część 1
Cel ćwiczenia:	Celem ćwiczenia jest poznanie działania polecenia SELECT. Student zapozna się z następującymi elementami języka: klauzule ORDER BY oraz WHERE, operatory, aliasy, wyrażenia, wartości puste NULL.
Wymagane przygotowanie teoretyczne:	Informacje podane na wykładzie oraz samodzielne przygotowanie się do zajęć.
Sposób zaliczenia:	Pozytywna ocena ćwiczenia przez prowadzącego pod koniec zajęć.
Prowadzący zajęcia:	dr inż. Artur Gramacki, dr inż. Jarosław Gramacki

1. Uwagi wstępne

Instrukcja ta ma nieco inny charakter niż poprzednie, w których omawiano (siłą rzeczy bardzo skrótowo i pobieżnie) ćwiczony materiał. Tutaj ograniczamy się jedynie do podania poleceń do wykonania oraz zaprezentowania wyniku jaki powinieneś uzyskać. Uczyniono tak, ponieważ język SQL był (będzie) dosyć dokładnie omawiany na wykładzie i nie powodu, aby w instrukcjach laboratoryjnych powtarzać ten materiał.

Jeżeli będziesz uzyskiwał na ekranie inne wyniki niż te prezentowane w instrukcji najprawdopodobniej będzie to oznaczało, że twoje zapytanie SQL jest błędne (chodzi o błędy logiczne, bo błędy składniowe zostaną natychmiast wykryte przez serwer MySQL).

Warunkiem uzyskania w ćwiczeniu takich jak poniżej wyników jest praca na demonstracyjnym schemacie relacyjnym, o którym mowa jest w jednym z poprzednich ćwiczeń. Dlatego też należy upewnić się, że odpowiedni skrypt wykonał się bezbłędnie (poprawnie utworzyły się wszystkie tabele oraz załadowały się wszystkie dane). Dla pewności warto przed rozpoczęciem ćwiczenia po prostu utworzyć schemat od nowa, nawet jeżeli wydaje ci się, że ten, który masz aktualnie zainstalowany jest poprawny.

Poniżej zamieszczone zestawienie ilości rekordów w każdej z tabel¹. Ten prosty test poprawności danych należy jednak potraktować bardzo ostrożnie, gdyż pokazuje on tylko ile jest rekordów w poszczególnych tabelach a nie czy są one zgodne z oryginałem.

```
mysql> SELECT table_name, table_rows
-> FROM information_schema.tables
-> WHERE table_schema='blab';
```

```
+-----+-----+
| table_name | table_rows |
+-----+-----+
| customer  |          15 |
| dept      |          12 |
| emp       |          25 |
```

¹Korzystamy tutaj ze specjalnej „systemowej” bazy danych o nazwie *information_schema*. W bazie tej przechowywane są różne informacje na temat innych baz. Jest to więc swego rodzaju *baza metadanych*

```

| inventory |          114 |
| item      |           62 |
| ord       |           16 |
| product   |           33 |
| region    |            5 |
| title     |            8 |
| warehouse |            5 |
+-----+-----+
10 rows in set (0.02 sec)

```

Gdy korzystasz z wersji MySQL wcześniejszej niż 5 ilość rekordów można łatwo sprawdzić wydając dla każdej tabeli polecenie:

```

mysql> SELECT COUNT(*) FROM emp;
+-----+
| COUNT(*) |
+-----+
|          25 |
+-----+
1 row in set (0.03 sec)

```

2. Operatory

W poleceniu `SELECT` będziesz wielokrotnie używał różnych *operatorów*. Na początku będą to głównie operatory porównania i operatory logiczne, choć w bardziej złożonych (nietypowych) przypadkach mogą być potrzebne też inne ich rodzaje.

W czasie używania operatorów warto pamiętać o kolejności ich wykonywania. Domyślna kolejność operatorów może być zmieniona za pomocą nawiasów. Poniżej przedstawiona jest leksykalna kolejność (w malejącym porządku) najczęściej używanych operatorów w MySQL. Więcej szczegółów znajdziesz w dokumentacji.

```

BINARY   (porównanie ciągów znaków z uwzględnieniem wielkości liter)
NOT !    (odwraca wyrażenie logiczne)
-        (minus jednoskładnikowy)
* / %    (mnożenie, dzielenie, reszta z dzielenia)
+ -      (dodawanie, odejmowanie)
<< >>   (przesunięcie w prawo i w lewo)
&        (bitowy iloczyn logiczny)
|        (bitowa suma logiczna)
< <= = <=> <> >= > IN LIKE (operatory porównania, sprawdzenie istnienia
                             wartości w liście, porównanie ciągów ze
                             znakami specjalnymi)

BETWEEN  (test zakresu)
AND &&   (iloczyn logiczny)
OR ||    (suma logiczna)

```

3. Polecenia do wykonania

Ćwiczenie 1

Wyświetlić dane o pracownikach (tabela emp). Zwrócić uwagę na to w jaki sposób wyświetlana jest kolumna start_date. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+-----+
| first_name | last_name   | start_date       | salary |
+-----+-----+-----+-----+
| Carmen     | Velasquez  | 1990-03-03 00:00:00 | 2500.00 |
| LaDoris    | Ngao       | 1990-03-08 00:00:00 | 1450.00 |
| Midori     | Nagayama   | 1991-06-17 00:00:00 | 1400.00 |
| Mark       | Quick-To-See | 1990-04-07 00:00:00 | 1450.00 |
| Audry      | Ropeburn   | 1990-03-04 00:00:00 | 1550.00 |
| Molly      | Urguhart   | 1991-01-18 00:00:00 | 1200.00 |
| Roberta    | Menchu     | 1990-05-14 00:00:00 | 1250.00 |
| Ben        | Biri       | 1990-04-07 00:00:00 | 1100.00 |
| Antoinette | Catchpole  | 1992-02-09 00:00:00 | 1300.00 |
| Marta      | Havel      | 1991-02-27 00:00:00 | 1307.00 |
| Colin      | Magee      | 1990-05-14 00:00:00 | 1400.00 |
| Henry      | Giljum     | 1992-01-18 00:00:00 | 1490.00 |
| Yasmin     | Sedeghi    | 1991-02-08 00:00:00 | 1515.00 |
| Mai        | Nguyen     | 1992-01-22 00:00:00 | 1525.00 |
| Andre      | Dumas      | 1991-10-09 00:00:00 | 1450.00 |
| Elena      | Maduro     | 1992-02-07 00:00:00 | 1400.00 |
| George     | Smith      | 1990-03-08 00:00:00 | 940.00  |
| Akira      | Nozaki     | 1991-02-09 00:00:00 | 1200.00 |
| Vikram     | Patel      | 1991-08-06 00:00:00 | 795.00  |
| Chad       | Newman     | 1991-07-21 00:00:00 | 750.00  |
| Alexander  | Markarian  | 1991-05-26 00:00:00 | 850.00  |
| Eddie      | Chang      | 1990-11-30 00:00:00 | 800.00  |
| Radha      | Patel      | 1990-10-17 00:00:00 | 795.00  |
| Bela       | Dancs     | 1991-03-17 00:00:00 | 860.00  |
| Sylvie     | Schwartz   | 1991-05-09 00:00:00 | 1100.00 |
+-----+-----+-----+-----+
25 rows in set (0.13 sec)

```

Ćwiczenie 2

Wyświetlić dane o pracownikach (tabela emp). Zwrócić uwagę na trochę być może sztuczny i „na siłę” układ i zawartość kolumn. Użyć funkcji scalającej napisy CONCAT. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+
| last_name   | last_name   | first_name & last_name |
+-----+-----+-----+
| Velasquez   | Velasquez   | Carmen->Velasquez    |
| Ngao        | Ngao        | LaDoris->Ngao         |
| Nagayama    | Nagayama    | Midori->Nagayama      |
| Quick-To-See | Quick-To-See | Mark->Quick-To-See    |
| Ropeburn    | Ropeburn    | Audry->Ropeburn       |
| Urguhart    | Urguhart    | Molly->Urguhart       |
| Menchu      | Menchu      | Roberta->Menchu       |
| Biri        | Biri        | Ben->Biri              |
| Catchpole   | Catchpole   | Antoinette->Catchpole |
| Havel       | Havel       | Marta->Havel           |
| Magee       | Magee       | Colin->Magee          |
| Giljum      | Giljum      | Henry->Giljum          |
| Sedeghi     | Sedeghi     | Yasmin->Sedeghi      |
| Nguyen      | Nguyen      | Mai->Nguyen            |
| Dumas       | Dumas       | Andre->Dumas           |

```

```

| Maduro      | Maduro      | Elena<->Maduro      |
| Smith       | Smith       | George<->Smith      |
| Nozaki      | Nozaki      | Akira<->Nozaki      |
| Patel       | Patel       | Vikram<->Patel      |
| Newman      | Newman      | Chad<->Newman       |
| Markarian   | Markarian   | Alexander<->Markarian |
| Chang       | Chang       | Eddie<->Chang       |
| Patel       | Patel       | Radha<->Patel       |
| Dancs       | Dancs       | Bela<->Dancs        |
| Schwartz    | Schwartz    | Sylvie<->Schwartz   |
+-----+-----+-----+
25 rows in set (0.01 sec)
4 rows in set (0.05 sec)

```

Ćwiczenie 3

Wyświetlić dane o pracownikach (tabela emp). Tym razem wybieramy tylko pracowników zarabiających powyżej 1500 i wynik sortujemy wg. nazwisk. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+-----+
| first_name | last_name | start_date          | salary |
+-----+-----+-----+-----+
| Mai        | Nguyen    | 1992-01-22 00:00:00 | 1525.00 |
| Audry      | Ropeburn  | 1990-03-04 00:00:00 | 1550.00 |
| Yasmin     | Sedeghi   | 1991-02-08 00:00:00 | 1515.00 |
| Carmen     | Velasquez | 1990-03-03 00:00:00 | 2500.00 |
+-----+-----+-----+-----+
4 rows in set (0.05 sec)

```

Ćwiczenie 4

Wyświetlić dane o pracownikach (tabela emp). Tym razem wybieramy tylko pracowników, których zarobki mieszczą się w przedziale od 1500 do 2500 (włącznie) i wynik sortujemy wg. tych zarobków. Na ile różnych sposobów można wykonać to polecenie? Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+-----+
| first_name | last_name | start_date          | salary |
+-----+-----+-----+-----+
| Yasmin     | Sedeghi   | 1991-02-08 00:00:00 | 1515.00 |
| Mai        | Nguyen    | 1992-01-22 00:00:00 | 1525.00 |
| Audry      | Ropeburn  | 1990-03-04 00:00:00 | 1550.00 |
| Carmen     | Velasquez | 1990-03-03 00:00:00 | 2500.00 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

Ćwiczenie 5

Wyświetlamy dane jak w poprzednim punkcie, ale zmieniamy etykiety. Zwrócić uwagę na wielkość liter w nazwach etykiet. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+-----+
| IMIE  | nazwisko | data zatrudnienia  | ZaRoBkI |
+-----+-----+-----+-----+
| Yasmin | Sedeghi  | 1991-02-08 00:00:00 | 1515.00 |
| Mai    | Nguyen   | 1992-01-22 00:00:00 | 1525.00 |
| Audry  | Ropeburn | 1990-03-04 00:00:00 | 1550.00 |

```

```

| Carmen | Velasquez | 1990-03-03 00:00:00 | 2500.00 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

Ćwiczenie 6

Wyświetlić dane o pracownikach (tabela emp). Tym razem wybieramy tylko pracowników, którzy pracują w dziale o numerze 41 (kolumna dept_id). Dodatkowo wyświetlamy kwotę zarobków pomnożoną przez 12. Jak można zmienić nazwę etykiety "salary*12" ? Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+-----+
| first_name | last_name | salary | salary*12 |
+-----+-----+-----+-----+
| LaDoris    | Ngao      | 1450.00 | 17400.00 |
| Molly      | Urguhart  | 1200.00 | 14400.00 |
| Elena      | Maduro    | 1400.00 | 16800.00 |
| George     | Smith     | 940.00  | 11280.00 |
+-----+-----+-----+-----+
4 rows in set (0.06 sec)

```

Ćwiczenie 7

Wyświetlić dane o pracownikach (tabela emp). Wybrać tylko tych pracowników, którzy zostali zatrudnieni po 31-12-1991 (kolumna start_date). Wynik posortować wg. daty zatrudnienia. Zwrócić uwagę na format podawanej daty w klauzuli WHERE. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+
| first_name | last_name | start_date          |
+-----+-----+-----+
| Henry      | Giljum    | 1992-01-18 00:00:00 |
| Mai        | Nguyen    | 1992-01-22 00:00:00 |
| Elena      | Maduro    | 1992-02-07 00:00:00 |
| Antoinette | Catchpole | 1992-02-09 00:00:00 |
+-----+-----+-----+
4 rows in set (0.00 sec)

```

Ćwiczenie 8

Wyświetlić bieżącą datę oraz bieżący czas systemowy. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+
| current_date | current_time | current_timestamp |
+-----+-----+-----+
| 2006-01-30   | 23:20:32     | 2006-01-30 23:20:32 |
+-----+-----+-----+
1 row in set (0.00 sec)

```

Ćwiczenie 9

Wyświetlić bieżącą datę oraz bieżący czas systemowy. W drugiej kolumnie specyficznie sformatować wynik (użyć funkcji DATE_FORMAT). Jako wynik powinniśmy otrzymać:

```

+-----+-----+
| current_timestamp | biezaca data i godzina |
+-----+-----+

```

```

+-----+-----+
| 2006-01-30 23:55:58 | Monday :: January :: 30 :: 2006 :: 23:55:58 |
+-----+-----+
1 row in set (0.00 sec)

```

Ćwiczenie 10

Wyświetlić dane o pracownikach. Wyświetlić datę zatrudnienia (kolumna `start_date`) oraz dodatkowo informację o tym, ile dni są już zatrudnieni. Wynik posortować wg. ilości tych dni. Użyć funkcji systemowej `DATEDIFF`. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+
| last_name | start_date | ilosc dni |
+-----+-----+-----+
| Velasquez | 1990-03-03 00:00:00 | 5812 |
| Ropeburn | 1990-03-04 00:00:00 | 5811 |
| Ngao | 1990-03-08 00:00:00 | 5807 |
| Smith | 1990-03-08 00:00:00 | 5807 |
| Quick-To-See | 1990-04-07 00:00:00 | 5777 |
| Biri | 1990-04-07 00:00:00 | 5777 |
| Magee | 1990-05-14 00:00:00 | 5740 |
| Menchu | 1990-05-14 00:00:00 | 5740 |
| Patel | 1990-10-17 00:00:00 | 5584 |
| Chang | 1990-11-30 00:00:00 | 5540 |
| Urguhart | 1991-01-18 00:00:00 | 5491 |
| Sedeghi | 1991-02-08 00:00:00 | 5470 |
| Nozaki | 1991-02-09 00:00:00 | 5469 |
| Havel | 1991-02-27 00:00:00 | 5451 |
| Dancs | 1991-03-17 00:00:00 | 5433 |
| Schwartz | 1991-05-09 00:00:00 | 5380 |
| Markarian | 1991-05-26 00:00:00 | 5363 |
| Nagayama | 1991-06-17 00:00:00 | 5341 |
| Newman | 1991-07-21 00:00:00 | 5307 |
| Patel | 1991-08-06 00:00:00 | 5291 |
| Dumas | 1991-10-09 00:00:00 | 5227 |
| Giljum | 1992-01-18 00:00:00 | 5126 |
| Nguyen | 1992-01-22 00:00:00 | 5122 |
| Maduro | 1992-02-07 00:00:00 | 5106 |
| Catchpole | 1992-02-09 00:00:00 | 5104 |
+-----+-----+-----+
25 rows in set (0.00 sec)

```

Ćwiczenie 11

Wyświetlić dane o pracownikach posortowane wg. nazwisk. Wyświetlić tylko pięć kolejnych rekordów poczynając od czwartego rekordu licząc od góry (użyć klauzuli `LIMIT`). Jako wynik powinniśmy otrzymać:

```

+-----+-----+
| first_name | last_name |
+-----+-----+
| Bela | Dancs |
| Andre | Dumas |
| Henry | Giljum |
| Marta | Havel |
| Elena | Maduro |
+-----+-----+
5 rows in set (0.00 sec)

```

Ćwiczenie 12

Wyświetlić dane z tabeli produktów (tabela `product`). Wyświetlić tylko te rekordy, gdzie nazwa produktu (kolumna `name`) rozpoczyna się na literę A, B lub C. Wynik dodatkowo posortować malejąco (od Z do A). Jako wynik powinniśmy otrzymać:

```
+-----+
| name          |
+-----+
| Chapman Helmet |
| Cabrera Bat   |
| Bunny Ski Pole |
| Bunny Boot    |
| Black Hawk Knee Pads |
| Black Hawk Elbow Pads |
| Alomar Glove  |
| Alexeyer Pro Lifting Bar |
| Ace Ski Pole  |
| Ace Ski Boot  |
+-----+
10 rows in set (0.00 sec)
```

Ćwiczenie 13

Wyświetlić dane z tabeli pracowników (tabela `emp`). Wyświetlić tylko te rekordy, gdzie wielkość prowizji (kolumna `commision_pct`) jest różna od wartości NULL. Jako wynik powinniśmy otrzymać:

```
+-----+-----+-----+
| first_name | last_name | COMMISSION_PCT |
+-----+-----+-----+
| Colin      | Magee     | 10.00 |
| Henry      | Giljum    | 12.50 |
| Yasmin     | Sedeghi   | 10.00 |
| Mai        | Nguyen    | 15.00 |
| Andre      | Dumas     | 17.50 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

Ćwiczenie 14

Wyświetlić dane z tabeli oddziałów (tabela `dept`). Wynik posortować wg. nazw oddziałów oraz wg. numerów regionów (kolumna `region_id`). Jako wynik powinniśmy otrzymać:

```
+----+-----+-----+
| id | name          | region_id |
+----+-----+-----+
| 50 | Administration | 1 |
| 10 | Finance        | 1 |
| 41 | Operations     | 1 |
| 42 | Operations     | 2 |
| 43 | Operations     | 3 |
| 44 | Operations     | 4 |
| 45 | Operations     | 5 |
| 31 | Sales          | 1 |
| 32 | Sales          | 2 |
```

```

| 33 | Sales          |          3 |
| 34 | Sales          |          4 |
| 35 | Sales          |          5 |
+-----+-----+-----+
12 rows in set (0.00 sec)

```

Ćwiczenie 15

Wyświetlić dane o klientach (tabela `customer`), których zdolność kredytowa (kolumna `credit_rating`) oceniana jest na dobrą (wartość w kolumnie: `GOOD`). Zwrócić uwagę na sposób formułowania warunku w klauzuli `WHERE`. Użyć raz operatora `=` a raz operatora `LIKE`. Jako wynik powinniśmy otrzymać:

```

+-----+-----+-----+
| name                | credit_rating |
+-----+-----+-----+
| Delhi Sports        | GOOD         |
| Sweet Rock Sports   | GOOD         |
| Muench Sports       | GOOD         |
+-----+-----+-----+
3 rows in set (0.00 sec)

```

Ćwiczenie 16

Z tabeli `dept` wyświetlić wszystkie nazwy oddziałów. Usunąć powtarzające się duplikaty (użyć klauzuli `DISTINCT`). Jako wynik powinniśmy otrzymać:

```

+-----+
| name          |
+-----+
| Administration |
| Finance       |
| Operations    |
| Sales         |
+-----+
4 rows in set (0.00 sec)

```

Literatura

- [1] Lech Banachowski (tłum.). *SQL. Język relacyjnych baz danych*. WNT Warszawa, 1995.
- [2] Paul Dubios. *MySQL. Podręcznik administratora*. Wydawnictwo HELION, 2005.
- [3] *MySQL 5.0 Reference Manual*, 2005. (jest to najbardziej aktualne opracowanie na temat bazy MySQL stworzone i na bieżąco aktualizowane przez jej twórców. Książka dostępna w wersji elektronicznej pod adresem <http://dev.mysql.com/doc/>).
- [4] Richard Stones and Neil Matthew. *Od podstaw. Bazy danych i MySQL*. Wydawnictwo HELION, 2003.
- [5] Luke Welling and Laura Thomson. *MySQL. Podstawy*. Wydawnictwo HELION, 2005.