



Zapytania MySQL

nauka przez przykłady

Poznaj i przećwicz: zapytania wyszukiujące, select, where, like, order by, zmieniające, update, set, wstawianie rekordów, insert, into, zmiana struktury bazy, alter, drop, delete, create

Structured Query Language

SQL

jest akronimem pochodzącym od **Structured Query Language** (strukturalny język zapytań kierowanych do bazy danych). Jest to język opracowany przez firmę IBM w celu współpracy z relacyjnymi bazami danych. SQL stał się standardem w komunikacji z relacyjnymi bazami danych. Dziś SQL-a spotkamy najczęściej w trzech systemach zarządzania bazami: **MySQL** firmy Oracle, **PostgreSQL** opracowany na uniwersytecie w Berkeley oraz **FireBird** korporacji Borland



Zapytania wyszukiujące (1a)

Tabelę wykorzystywaną w ćwiczeniu można pobrać i dokonać importu do dowolnej bazy danych. Tabela zawiera pytania z egzaminów zawodowych

<http://bit.ly/pytania-sql>

Znajdź w bazie:

1. Pytanie nr 25
2. Pytania, w których poprawną odpowiedzią jest “a”
3. Pytania z roku 2007
4. Pytania z programowania z roku 2009
5. Tylko treści pytań i 4 odpowiedzi z roku 2010
6. Pytania ułożone alfabetycznie wg treści
7. Treści pytań o numerach 10-12
8. Pytania zaczynające się od słów “Jaki”
9. Pytania zawierające frazę “blok” (także wewnątrz innych słów)
10. Pytania z programowania oraz systemów operacyjnych z roku 2012

Zapytania wyszukiujące (1b)

Odpowiedzi do zadań:

1. `SELECT * FROM pytania WHERE id=25`
2. `SELECT * FROM pytania WHERE answer="a" lub SELECT * FROM pytania WHERE answer='a'`
3. `SELECT * FROM pytania WHERE rok=2007`
4. `SELECT * FROM pytania WHERE rok=2009 AND kategoria='programowanie'`
5. `SELECT id, tresc, odpa, odpb, odpc, odpd FROM pytania WHERE rok=2010`
6. `SELECT * FROM pytania ORDER BY tresc ASC`
7. `SELECT id, tresc FROM pytania WHERE id>=100 AND id<=120 albo SELECT id, tresc FROM pytania WHERE id BETWEEN 10 AND 12`
8. `SELECT * FROM pytania WHERE tresc LIKE 'Jaki%'`
9. `SELECT * FROM pytania WHERE tresc LIKE '%blok%' OR odpa LIKE '%blok%' OR odpb LIKE '%blok%' OR odpc LIKE '%blok%' OR odpd LIKE '%blok%'`
10. `SELECT * FROM pytania WHERE rok=2012 AND (kategoria='systemy operacyjne i sieci' OR kategoria='programowanie')`

Zapytania wyszukiujące (2a)

Trzy tabele wykorzystywane w tym ćwiczeniu można pobrać i dokonać importu do dowolnej bazy danych. Są to tabele obsługujące księgarnię

<http://bit.ly/ksiegarnia-sql>

Znajdź w bazie:

1. Książki ułożone alfabetycznie wg tytułu
2. Najdroższą książkę w bazie
3. Wszystkie wysłane zamówienia
4. Wszystkich klientów o nazwisku Rutowski
5. Książki zawierające wyrażenie "PHP" w tytule
6. Najwcześniej dokonane zamówienie
7. Z użyciem podzapytania: najdroższa książka w bazie

Zapytania związane z łączeniem tabel:

8. Wyjmij dla wszystkich zamówień: imię i nazwisko klienta zamawiającego, id zamówienia, datę zamówienia
9. Jak w punkcie 8, ale z aliasami (przezskami) dla tabel
10. Które osoby zamówiły książkę nr 2
11. Jakie książki zamówiła osoba nr 4
12. Jakie książki zamówiła osoba o nazwisku Grzywocz
13. Książki j.w. w dodatku posortowane wg alfabetu
14. Z użyciem klauzuli LEFT JOIN: Znajdź imiona i nazwiska klientów, którzy niczego nie zamówili (będą mieli wartość NULL czyli niezdefiniowaną dla idzamowienia)

Zapytania wyszukiujące (2b)

Odpowiedzi do zadań:

1. `SELECT * FROM ksiazki ORDER BY tytul ASC`
2. `SELECT * FROM ksiazki ORDER BY cena DESC LIMIT 1`
3. `SELECT * FROM zamowienia WHERE status="wyslano"`
4. `SELECT * FROM klienci WHERE nazwisko="Rutkowski"`
5. `SELECT * FROM ksiazki WHERE tytul LIKE '%PHP%'`
6. `SELECT * FROM zamowienia ORDER BY data ASC`
7. `SELECT * FROM ksiazki WHERE cena = (SELECT MAX(cena) FROM ksiazki)`
8. `SELECT klienci.imie, klienci.nazwisko, zamowienia.idzamowienia, zamowienia.data FROM klienci, zamowienia WHERE klienci.idklienta = zamowienia.idklienta`
9. `SELECT k.imie, k.nazwisko, z.idzamowienia, z.data FROM klienci AS k, zamowienia AS z WHERE k.idklienta = z.idklienta`
10. `SELECT klienci.imie, klienci.nazwisko FROM klienci, zamowienia WHERE zamowienia.idksiazki=2 and zamowienia.idklienta = klienci.idklienta`
11. `SELECT ksiazki.tytul, ksiazki.imieautora, ksiazki.nazwiskoautora FROM ksiazki, zamowienia WHERE zamowienia.idklienta=4 and zamowienia.idksiazki = ksiazki.idksiazki`
12. `SELECT ksiazki.tytul, ksiazki.imieautora, ksiazki.nazwiskoautora FROM ksiazki, zamowienia, klienci WHERE klienci.nazwisko="Grzywocz" and klienci.idklienta=zamowienia.idklienta and zamowienia.idksiazki = ksiazki.idksiazki`
13. `SELECT ksiazki.tytul, ksiazki.imieautora, ksiazki.nazwiskoautora FROM ksiazki, zamowienia, klienci WHERE klienci.nazwisko="Grzywocz" and klienci.idklienta=zamowienia.idklienta and zamowienia.idksiazki = ksiazki.idksiazki ORDER BY ksiazki.tytul DESC`
14. `SELECT klienci.idklienta, klienci.imie, klienci.nazwisko FROM klienci LEFT JOIN zamowienia USING (idklienta) WHERE zamowienia.idzamowienia IS NULL`

Zapytania zmieniające dane (3a)

Trzy tabele wykorzystywane w tym ćwiczeniu można pobrać i dokonać importu do dowolnej bazy danych. Są to tabele obsługujące księgarnię

<http://bit.ly/ksiegarnia-sql>

Zmień w bazie:

1. Zmień nazwisko osoby nr 1 na “Kowalczyk”
2. Spróbuj zmienić idklienta osoby nr 3 na wartość 1
3. Zwiększ cenę wszystkich książek w bazie o 10%
4. Zmniejsz cenę najdroższej książki o 10 zł
5. Zmień imię i nazwisko klientki Anny Kareniny na: “Joanna Dostojewska”
6. Zmień status zamówień o numerach 3-5 na “wyslano”

Zapytania zmieniające dane (3b)

Odpowiedzi do zadań:

1. UPDATE klienci SET nazwisko="Kowalczyk" WHERE idklienta=1
2. Nie można tego uczynić, bo wystąpiłyby dwa rekordy o tym samym idklienta!
3. UPDATE ksiazki SET cena=cena*1.1
4. UPDATE ksiazki SET cena=cena-10 ORDER BY cena DESC LIMIT 1

uwaga! - nie można tego zrobić tak:

```
UPDATE ksiazki SET cena=cena-10 WHERE  
cena = (SELECT MAX(cena) FROM ksiazki)
```

5. UPDATE klienci SET imie="Joanna", nazwisko="Dostojewska"
WHERE imie="Anna" AND nazwisko="Karenina"
6. UPDATE zamowienia SET status='wyslano' WHERE idzamowienia
BETWEEN 3 AND 5

Zapytania wstawiające dane (4a)

Trzy tabele wykorzystywane w tym ćwiczeniu można pobrać i dokonać importu do dowolnej bazy danych. Są to tabele obsługujące księgarnię

<http://bit.ly/ksiegarnia-sql>

Wstaw do bazy:

1. Dodaj nowego klienta: Franciszek Janowski z Chorzowa
2. Dodaj nowe zamówienie: Artur Rutkowski zamówił książkę: “HTML 5. Tworzenie gier”
3. Wstaw do bazy książkę o tytule: “Symfonia C++” autora o nazwisku Grębosz, ale nie wstawiaj jeszcze imienia autora ani ceny książki
4. Dodaj dwóch nowych klientów za pomocą jednego zapytania
5. Spróbuj wstawić do tabeli klienci nową osobę o id równym 5
6. Wstaw nową osobę używając klauzuli SET

Zapytania wstawiające dane (4b)

Odpowiedzi do zadań:

1. INSERT INTO klienci VALUES (NULL, 'Franciszek', 'Janowski', 'Chorzów')
2. INSERT INTO zamowienia VALUES (NULL, 7, 3, "2012-10-11", 'oczekiwanie')
3. INSERT INTO ksiazki
(idksiazki, nazwiskoautora, tytul) VALUES
(NULL, 'Grębosz', 'Symfonia C++')
4. INSERT INTO klienci VALUES
(NULL, 'Mario', 'Hydraulik', 'Gliwice'),
(NULL, 'Luigi', 'Hydraulik', 'Gliwice')
5. Nie można tego zrobić, bo rekord o id=5 już istnieje!
6. INSERT INTO klienci set idklienta=NULL, imie='Jan', nazwisko='Nowy',
miescowosc='Rybnik'

Zapytania zmieniające strukturę (5a)

Trzy tabele wykorzystywane w tym ćwiczeniu można pobrać i dokonać importu do dowolnej bazy danych. Są to tabele obsługujące księgarnię

<http://bit.ly/ksiegarnia-sql>

Polecenia do wykonania:

1. Dodaj do bazy tabelę o nazwie “promocje” zawierającą następujące dane:
 - idpromocji (klucz podstawowy, typ integer)
 - idksiazki (typu integer)
 - rodzajpromocji (typu text)
 - cenawpromocji (typu float)
2. Dodaj trzy nowe wiersze do tej tabeli przy pomocy jednego zapytania SQL
3. Zmień cenę trzeciej książki w promocji na 15.99 PLN i rodzaj jej promocji na “święteczna”
4. Zmień nazwę kolumny “rodzajpromocji” na “rodzaj” (a potem przywróć dawną nazwę)
5. Zmień typ danych kolumny “rodzajpromocji” na longtext
6. Usuń z tabeli “promocje” kolumnę “rodzajpromocji”
7. Usuń z tabeli “promocje” klucz podstawowy
8. Ponownie dodaj do tabeli “promocje” klucz podstawowy “idpromocji”
9. Dodaj na końcu nową kolumnę “aktualna” i ustaw typ danych na “boolean”
10. Wstaw do wszystkich wartości w kolumnie “aktualna” wartość TRUE
11. Usuń całą tabelę “promocje”
12. Usuń jeden rekord z tabeli zamowienia o id równym 1

Zapytania zmieniające strukturę (5b)

Odpowiedzi do zadań:

1. CREATE TABLE ksiegarnia.promocje (idpromocji INT NOT NULL AUTO_INCREMENT PRIMARY KEY, idksiazki INT NOT NULL, rodzajpromocji TEXT CHARACTER SET utf8 COLLATE utf8_polish_ci, cenawpromocji FLOAT NOT NULL)
2. INSERT INTO promocje VALUES (NULL, 3, 'mikołajki', 10), (NULL, 1, 'mikołajki', 34.99), (NULL, 4, 'mikołajki', 12.99)
3. UPDATE promocje SET cenawpromocji=15.99, rodzajpromocji='świąteczna' WHERE idpromocji=3
4. ALTER TABLE promocje CHANGE rodzajpromocji rodzaj TEXT CHARACTER SET utf8 COLLATE utf8_polish_ci NULL
5. ALTER TABLE promocje CHANGE rodzajpromocji rodzajpromocji LONGTEXT CHARACTER SET utf8 COLLATE utf8_polish_ci NULL
6. ALTER TABLE promocje DROP rodzajpromocji
7. ALTER TABLE promocje DROP PRIMARY KEY (ale trzeba usunąć najpierw A_I):
ALTER TABLE promocje CHANGE idpromocji idpromocji INT NOT NULL
8. ALTER TABLE promocje CHANGE idpromocji idpromocji INT NOT NULL PRIMARY KEY AUTO_INCREMENT
9. ALTER TABLE promocje ADD aktualna BOOLEAN NOT NULL
10. UPDATE promocje SET aktualna=1
11. DROP TABLE promocje
12. DELETE FROM zamowienia WHERE idzamowienia=1