

Bazy Danych Projekt

Tomasz Polit

168160

Przykładowe zapytanie SELECT:

```
1 • USE royal_family_tree;
2 • SELECT
3     p.First_Name,
4     p.Regnal_Number,
5     t.Title_Name,
6     r.Relation_Type
7
8 FROM persons p
9 JOIN title_person tp
10     ON p.Person_ID = tp.Person_ID
11 JOIN titles t
12     ON tp.Title_ID = t.Title_ID
13 JOIN primary_person_in_relation ppir
14     ON ppir.PPIR_Person_ID = p.Person_ID
15 JOIN secondary_person_in_relation spir
16     ON spir.SPIR_Person_ID = p.Person_ID
17 JOIN relations r
18     ON ppir.PPIR_Relation_ID = r.Relation_ID
19     OR spir.SPIR_Relation_ID = r.Relation_ID
20 WHERE (tp.Title_ID = 1 OR tp.Title_ID = 2) AND (r.Secondary_Person_ID = 31
21 AND (r.Relation_Type LIKE '%fathe_' OR r.Relation_Type LIKE '%Fathe_'))
22
23 GROUP BY p.First_Name, p.Regnal_Number
24 ORDER BY p.First_Name, p.Regnal_Number
```

<

Result Grid | | Filter Rows: | Export: | Wrap Cell Content:

	First_Name	Regnal_Number	Title_Name	Relation_Type
▶	Charles	III	King of the United Kingdom and other Commonwealth realms	Father
	George	V	King of the United Kingdom and other Commonwealth realms	Great-Great-Grandfather
	George	VI	King of the United Kingdom and other Commonwealth realms	Great-Grandfather

Wykorzystuję w nim wszystkie tablice. Poszukujemy imienia, numeru porządkowego, nazwy tytułu, typu relacji wszystkich osób, które są dla Wiliama Windsora ojcem, dziadkiem, pradziadkiem itd., które posiadały tytuł króla Anglii.

Przykładowy UPDATE:

Zmienię relację pomiędzy królową Elżbietą II oraz księciem Walii Wiliamem w taki sposób, że Elżbieta zostanie nazwana dziadkiem a nie babcią Wiliama.

```
1 • USE royal_family_tree;
2
3 • UPDATE relations SET Relation_Type = 'Grandfather'
4   WHERE Relation_ID = 851 AND Primary_Person_ID = 12 AND Secondary_Person_ID = 31
```

Urzuwając widoczny powyżej zapytania UPDATE zmienię wynik jaki pokaże poprzednio urzyty SELECT.

```
1 • USE royal_family_tree;
2 • SELECT
3     p.First_Name,
4     p.Regnal_Number,
5     t.Title_Name,
6     r.Relation_Type
7
8   FROM persons p
9   JOIN title_person tp
10    ON p.Person_ID = tp.Person_ID
11  JOIN titles t
12    ON tp.Title_ID = t.Title_ID
13  JOIN primary_person_in_relation ppir
14    ON ppir.PPIR_Person_ID = p.Person_ID
15  JOIN secondary_person_in_relation spir
16    ON spir.SPIR_Person_ID = p.Person_ID
17  JOIN relations r
18    ON ppir.PPIR_Relation_ID = r.Relation_ID
19    OR spir.SPIR_Relation_ID = r.Relation_ID
20  WHERE (tp.Title_ID = 1 OR tp.Title_ID = 2) AND (r.Secondary_Person_ID = 31
21         AND (r.Relation_Type LIKE '%fathe_' OR r.Relation_Type LIKE '%Fathe_'))
22
23  GROUP BY p.First_Name, p.Regnal_Number
24  ORDER BY p.First_Name, p.Regnal_Number
```

Result Grid				
Filter Rows: <input type="text"/>				
Export: Wrap Cell Content:				
	First_Name	Regnal_Number	Title_Name	Relation_Type
▶	Charles	III	King of the United Kingdom and other Commonwealth realms	Father
	Elizabeth	II	Queen of the United Kingdom and other Commonwealth re...	Grandfather
	George	V	King of the United Kingdom and other Commonwealth realms	Great-Great-Grandfather
	George	VI	King of the United Kingdom and other Commonwealth realms	Great-Grandfather

Jak widać na zdjęciu powyżej Elżbieta pojawiła się w wyniku jako dziadek Wiliama.

Utworzenie użytkownika zapytaniem SQL:

```
1 • CREATE USER 'Administrator'@'localhost'
2     IDENTIFIED BY 'password';
3 • GRANT ALL
4     ON *.*
5     TO 'Administrator'@'localhost';
6
7 • CREATE USER 'Edytor'@'%'
8     IDENTIFIED BY 'password';
9 • GRANT SELECT,INSERT,UPDATE,DELETE
10    ON royal_family_tree.*
11    TO 'Edytor'@'%';
12
13 • CREATE USER 'Odwiedzający'@'%';
14 • GRANT SELECT
15    ON royal_family_tree.*
16    TO 'Odwiedzający'@'%';
```

W ramach projektu w bazie danych utworzeni zostali użytkownicy o odpowiednich rolach.

The screenshot shows the MySQL 'Users and Privileges' window for a local instance. On the left, a table lists existing users. The 'Odwiedzający' user is selected. The main panel shows the configuration for this user, including login name, authentication type, password, and confirmation password. The 'No password is set for this account.' message is visible at the bottom.

User	From Host
Administrator	localhost
Edytor	%
Odwiedzający	%
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost

Details for account Odwiedzający@%

Login: You may create multiple accounts with the same name to connect from different hosts.

Authentication Type: For the standard password and/or host based authentication, select 'Standard'.

Limit to Hosts Matching: % and _ wildcards may be used

Password: Type a password to reset it.

Confirm Password: Enter password again to confirm.

Authentication String: Authentication plugin specific parameters.

See the plugin documentation for valid values and details.

No password is set for this account.

Wynikiem zapytania są utworzeni użytkownicy, co widzimy na zdjęciu powyżej.

Baza danych posiada trzech użytkowników:

Administrator

Jego rola polega na zarządzaniu działaniem baz danych. Nie jest on ograniczony do żadnej bazy danych czy konkretnego zapytania. Rola ta jest bardzo ważna i jest przez to chroniona hasłem.

Edytora

Rola ta polega na wprowadzaniu niezbędnych zmian i poprawek do istniejącej bazy danych. Jest on jednak ograniczony tylko do niezbędnych do edycji zapytań wyłącznie wewnątrz bazy `royal_family_tree`. Rola ta jest chroniona hasłem.

Odwiedzający

Jest to rola, która jest przyznawana każdemu nowemu użytkownikowi. Nie jest ona chroniona hasłem, lecz jest ograniczona ściśle do zapytania `SELECT` tylko bazy danych `royal_family_tree`.

Funkcja duplikująca:

```
1 • CREATE SCHEMA `new_royal_family_tree`;  
2 • USE new_royal_family_tree;  
3  
4 • CREATE TABLE New_Persons LIKE royal_family_tree.persons;  
5 • INSERT New_Persons SELECT * FROM royal_family_tree.persons;  
6  
7 • CREATE TABLE New_Relations LIKE royal_family_tree.relations;  
8 • INSERT New_Relations SELECT * FROM royal_family_tree.relations;  
9  
10 • CREATE TABLE New_Primary_Person_in_Relation LIKE royal_family_tree.primary_person_in_relation;  
11 • INSERT New_Primary_Person_in_Relation SELECT * FROM royal_family_tree.primary_person_in_relation;  
12  
13 • CREATE TABLE New_Secondary_Person_in_Relation LIKE royal_family_tree.secondary_person_in_relation;  
14 • INSERT New_Secondary_Person_in_Relation SELECT * FROM royal_family_tree.secondary_person_in_relation;  
15  
16 • CREATE TABLE New_Title_Person LIKE royal_family_tree.title_person;  
17 • INSERT New_Title_Person SELECT * FROM royal_family_tree.title_person;  
18  
19 • CREATE TABLE New_Titles LIKE royal_family_tree.titles;  
20 • INSERT New_Titles SELECT * FROM royal_family_tree.titles;
```

Powyższe komendy tworzą duplikat bazy danych, który może posłużyć, jako wariant testowy.