

# Projekt studencki - Bazy danych

Kamil Polit

Nr. 168159

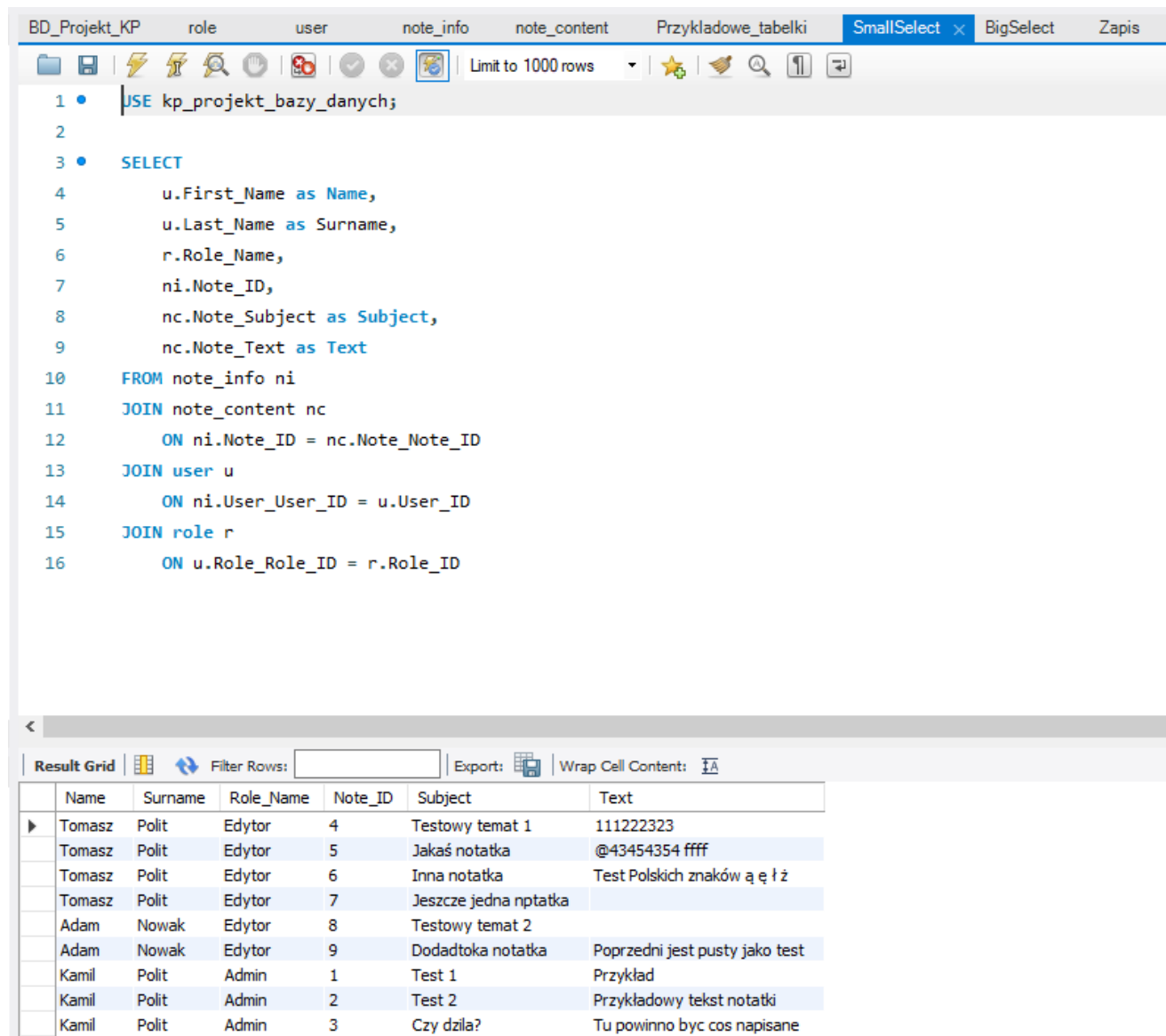
## Temat projektu: Baza danych do zapisu notatek tekstowych.

```
BD_Projekt_KP  role  user  note_info  note_content  Przykładowe_tabelki x SmallSelect  Zapis
Limit to 1000 rows

1 • INSERT role VALUES (10,'Edytor');
2 • INSERT role VALUES (30,'Admin');
3
4 • INSERT user VALUES (301, 'Kamil', 'Polit', NULL, 30);
5 • INSERT user VALUES (101, 'Tomasz', 'Polit', NULL, 10);
6 • INSERT user VALUES (102, 'Adam', 'Nowak', NULL, 10);
7 • INSERT user VALUES (103, 'Paweł', 'Kowalski', NULL, 10);
8
9 • INSERT note_info VALUES (1, 301, '2022-11-17', 0, NULL, 0);
10 • INSERT note_info VALUES (2, 301, '2022-11-17', 0, NULL, 0);
11 • INSERT note_info VALUES (3, 301, '2022-11-20', 0, NULL, 0);
12 • INSERT note_info VALUES (4, 101, '2022-11-21', 0, NULL, 0);
13 • INSERT note_info VALUES (5, 101, '2022-11-21', 0, NULL, 0);
14 • INSERT note_info VALUES (6, 101, '2022-11-24', 0, NULL, 0);
15 • INSERT note_info VALUES (7, 101, '2022-11-25', 0, NULL, 0);
16 • INSERT note_info VALUES (8, 102, '2022-11-25', 0, NULL, 0);
17 • INSERT note_info VALUES (9, 102, '2022-11-25', 0, NULL, 0);
18
19 • INSERT note_content VALUES (322, 1, 'Test 1', 'Przykład');
20 • INSERT note_content VALUES (323, 2, 'Test 2', 'Przykładowy tekst notatki');
21 • INSERT note_content VALUES (324, 3, 'Czy dzila?', 'Tu powinno byc cos napisane');
22 • INSERT note_content VALUES (202, 4, 'Testowy temat 1', '111222323');
23 • INSERT note_content VALUES (203, 5, 'Jakaś notatka', '@43454354 ffff');
24 • INSERT note_content VALUES (204, 6, 'Inna notatka', 'Test Polskich znaków ą ę ł ż');
25 • INSERT note_content VALUES (205, 7, 'Jeszcze jedna nptatka', '');
26 • INSERT note_content VALUES (190, 8, 'Testowy temat 2', '');
27 • INSERT note_content VALUES (191, 9, 'Dodadtoka notatka', 'Poprzedni jest pusty jako test');
```

Powyższy skrypt służy do wypełnienia tabel w bazie danych.

## Prosta komenda SELECT :



The screenshot shows a database management interface with a SQL editor and a results grid. The SQL editor contains a query that selects user information and note details. The results grid displays the output of this query, showing columns for Name, Surname, Role\_Name, Note\_ID, Subject, and Text. The data is organized into rows, with each row representing a specific user and their associated notes.

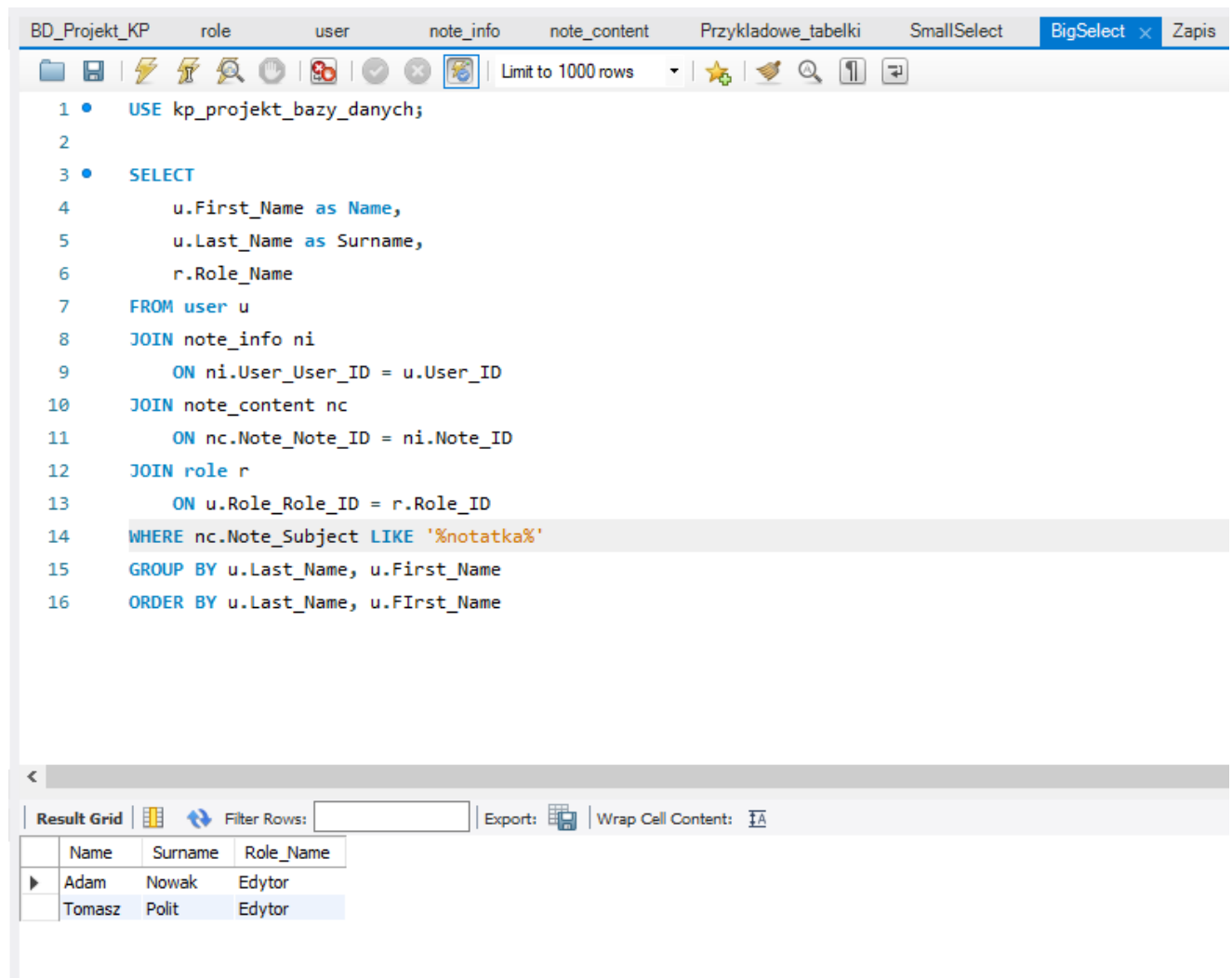
```
USE kp_projekt_bazy_danych;

SELECT
    u.First_Name as Name,
    u.Last_Name as Surname,
    r.Role_Name,
    ni.Note_ID,
    nc.Note_Subject as Subject,
    nc.Note_Text as Text
FROM note_info ni
JOIN note_content nc
    ON ni.Note_ID = nc.Note_Note_ID
JOIN user u
    ON ni.User_User_ID = u.User_ID
JOIN role r
    ON u.Role_Role_ID = r.Role_ID
```

	Name	Surname	Role_Name	Note_ID	Subject	Text
▶	Tomasz	Polit	Edytor	4	Testowy temat 1	111222323
	Tomasz	Polit	Edytor	5	Jakaś notatka	@43454354 ffff
	Tomasz	Polit	Edytor	6	Inna notatka	Test Polskich znaków ą ę ł ż
	Tomasz	Polit	Edytor	7	Jeszcze jedna nptatka	
	Adam	Nowak	Edytor	8	Testowy temat 2	
	Adam	Nowak	Edytor	9	Dodadtoka notatka	Poprzedni jest pusty jako test
	Kamil	Polit	Admin	1	Test 1	Przykład
	Kamil	Polit	Admin	2	Test 2	Przykładowy tekst notatki
	Kamil	Polit	Admin	3	Czy dzila?	Tu powinno byc cos napisane

Powyższy skrypt pozwala na wyświetlenie wszystkich użytkowników (ich imię, nazwisko i rolę) oraz notatek (jej ID, temat oraz zawartość).

## Rozbudowana komenda SELECT:



The screenshot shows a database management interface with a tab labeled 'BigSelect'. The SQL query is as follows:

```
1 • USE kp_projekt_bazy_danych;
2
3 • SELECT
4     u.First_Name as Name,
5     u.Last_Name as Surname,
6     r.Role_Name
7 FROM user u
8 JOIN note_info ni
9     ON ni.User_User_ID = u.User_ID
10 JOIN note_content nc
11     ON nc.Note_Note_ID = ni.Note_ID
12 JOIN role r
13     ON u.Role_Role_ID = r.Role_ID
14 WHERE nc.Note_Subject LIKE '%notatka%'
15 GROUP BY u.Last_Name, u.First_Name
16 ORDER BY u.Last_Name, u.First_Name
```

Below the query, the 'Result Grid' shows the following data:

	Name	Surname	Role_Name
▶	Adam	Nowak	Edytor
	Tomasz	Polit	Edytor

Powyższy skrypt wyświetla wszystkie osoby, które posiadają notatkę w której temacie znajduje się fragment tekstu „notatka” z dowolnym ciągiem znaków po obu stronach. Zgrupowane oraz poukładane są one względem imienia i nazwiska.

## Funkcja update:

```
1 • USE kp_projekt_bazy_danych;
2
3 • UPDATE note_content SET Note_Subject = 'Nowy test 2'
4 WHERE Note_Note_ID = 2 AND Note_Content_PK = 323
```

Powyższe komendy służą do zmiany tematu notatki o ID 2 oraz kluczu 323.

Taką tabelę zwraca skrypt „SmallSelect” przed użyciem skryptu:

	Name	Surname	Role_Name	Note_ID	Subject	Text
▶	Tomasz	Polit	Edytor	4	Testowy temat 1	111222323
	Tomasz	Polit	Edytor	5	Jakaś notatka	@43454354 ffff
	Tomasz	Polit	Edytor	6	Inna notatka	Test Polskich znaków ą ę ł ż
	Tomasz	Polit	Edytor	7	Jeszcze jedna nptatka	
	Adam	Nowak	Edytor	8	Testowy temat 2	
	Adam	Nowak	Edytor	9	Dodadtoka notatka	Poprzedni jest pusty jako test
	Kamil	Polit	Admin	1	Test 1	Przykład
	Kamil	Polit	Admin	2	Test 2	Przykładowy tekst notatki
	Kamil	Polit	Admin	3	Czy dzila?	Tu powinno byc cos napisane

Tak wygląda tabela po użyciu podanego skryptu Update:

	Name	Surname	Role_Name	Note_ID	Subject	Text
▶	Tomasz	Polit	Edytor	4	Testowy temat 1	111222323
	Tomasz	Polit	Edytor	5	Jakaś notatka	@43454354 ffff
	Tomasz	Polit	Edytor	6	Inna notatka	Test Polskich znaków ą ę ł ż
	Tomasz	Polit	Edytor	7	Jeszcze jedna nptatka	
	Adam	Nowak	Edytor	8	Testowy temat 2	
	Adam	Nowak	Edytor	9	Dodadtoka notatka	Poprzedni jest pusty jako test
	Kamil	Polit	Admin	1	Test 1	Przykład
	Kamil	Polit	Admin	2	Nowy test 2	Przykładowy tekst notatki
	Kamil	Polit	Admin	3	Czy dzila?	Tu powinno byc cos napisane

**Skrypt zliczający ilość notatek** przypisanych do konkretnych użytkowników i wyświetlający ich: Imię, Nazwisko oraz ilość notatek.

```
1 • USE kp_projekt_bazy_danych;
2
3 • SELECT
4     u.First_Name as Name,
5     u.Last_Name as Surname,
6     COUNT(Note_ID) AS NumberOfNotes
7 FROM note_info ni
8 RIGHT JOIN user u
9     ON ni.User_User_ID = u.User_ID
10 GROUP BY First_Name, Last_Name
11 ORDER BY First_Name
```

	Name	Surname	NumberOfNotes
▶	Adam	Nowak	2
	Kamil	Polit	3
	Paweł	Kowalski	0
	Tomasz	Polit	4

## Funkcja tworzenia użytkowników w bazie danych:

Skrypt tworzący użytkowników bazy danych:

```

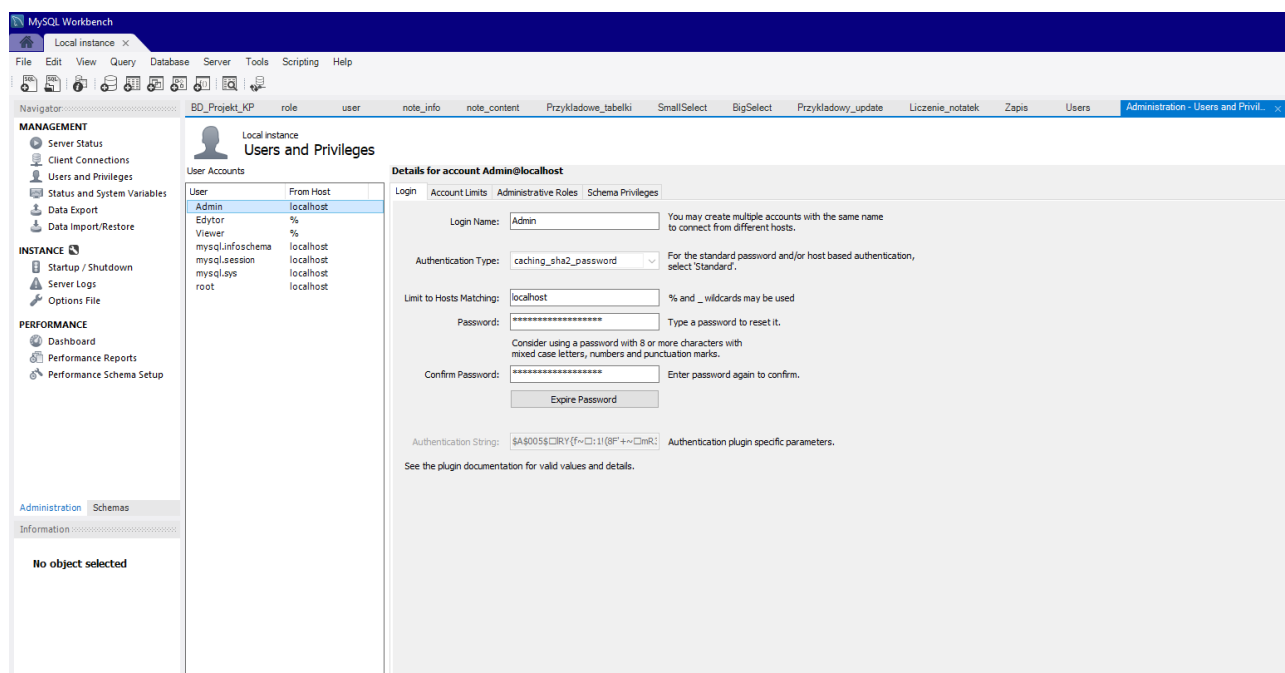
1      -- Tworzenie Admina --
2 •    CREATE USER 'Admin'@'localhost'
3          IDENTIFIED BY 'password';
4 •    GRANT ALL
5          ON *.*
6          TO 'Admin'@'localhost';
7
8      -- Tworzenie edytora --
9 •    CREATE USER 'Edytor'@'%'
10         IDENTIFIED BY 'password';
11 •    GRANT SELECT, INSERT, UPDATE, DELETE
12         ON kp_projekt_bazy_danych.*
13         TO 'Edytor'@'%';
14
15     -- Tworzenie osobę przeglądającą --
16 •    CREATE USER 'Viewer'@'%';
17 •    GRANT SELECT
18         ON kp_projekt_bazy_danych.*
19         TO 'Viewer'@'%';

```

Po wykonaniu skryptu zwracany jest ten komunikat:

Output			
Action Output			
#	Time	Action	
✓ 51	21:55:03	CREATE USER 'Admin'@'localhost' IDENTIFIED BY 'password'	
✓ 52	21:55:03	GRANT ALL ON *.* TO 'Admin'@'localhost'	
✓ 53	21:55:03	CREATE USER 'Edytor'@'%' IDENTIFIED BY 'password'	
✓ 54	21:55:03	GRANT SELECT, INSERT, UPDATE, DELETE ON kp_projekt_bazy_danych.* TO 'Edytor'@'%'	
✓ 55	21:55:03	CREATE USER 'Viewer'@'%'	
✓ 56	21:55:03	GRANT SELECT ON kp_projekt_bazy_danych.* TO 'Viewer'@'%'	

## Potwierdzenie utworzenia nowych użytkowników:



## Jakie możliwości mają poszczególne role w bazie danych:

- **Gość** – ta rola umożliwia jedynie używanie komendy SELECT. Nie jest zamknięta za żadnym hasłem i może jej używać każdy.
- **Edytor** – może używać komend: SELECT, INSERT, UPDATE, DELETE oraz jest zamknięta za hasłem. Oznacza to, że użytkownik może jedynie zmieniać zawartość tabel bazy danych i je przeglądać ale nie może w żaden sposób wpływać na konstrukcję bazy.
- **Administrator** – rola zezwala użytkownikowi na wykorzystywanie wszystkich komend wewnątrz bazy. Oznacza to że Administrator ma możliwość na zmianę struktury bazy. Jest ona również chroniona hasłem.

## Tworzenie wariantu testowego bazy danych:

Skrypt –

```
1 • CREATE SCHEMA `NEW_kp_projekt_bazy_danych`;  
2 • USE NEW_kp_projekt_bazy_danych;  
3  
4 • CREATE TABLE New_role LIKE kp_projekt_bazy_danych.role;  
5 • INSERT New_role SELECT * FROM kp_projekt_bazy_danych.role;  
6  
7 • CREATE TABLE New_user LIKE kp_projekt_bazy_danych.user;  
8 • INSERT New_user SELECT * FROM kp_projekt_bazy_danych.user;  
9  
10 • CREATE TABLE New_note_info LIKE kp_projekt_bazy_danych.note_info;  
11 • INSERT New_note_info SELECT * FROM kp_projekt_bazy_danych.note_info;  
12  
13 • CREATE TABLE New_note_content LIKE kp_projekt_bazy_danych.note_content;  
14 • INSERT New_note_content SELECT * FROM kp_projekt_bazy_danych.note_content;
```

Output –

Output			
Action Output			
	#	Time	Action
✓	61	22:14:16	CREATE TABLE New_user LIKE kp_projekt_bazy_danych.user
✓	62	22:14:16	INSERT New_user SELECT * FROM kp_projekt_bazy_danych.user
✓	63	22:14:16	CREATE TABLE New_note_info LIKE kp_projekt_bazy_danych.note_info
✓	64	22:14:16	INSERT New_note_info SELECT * FROM kp_projekt_bazy_danych.note_info
✓	65	22:14:16	CREATE TABLE New_note_content LIKE kp_projekt_bazy_danych.note_content
✓	66	22:14:16	INSERT New_note_content SELECT * FROM kp_projekt_bazy_danych.note_content

Potwierdzenie utworzenia nowej schem'y –

