

Julia Ruszer 247775

Dominik Gałkowski 247659

Jan Śladowski 247806

Wiktor Żelechowski 247833

# Microsoft SQL Server

## Zadanie 1.

### 1. Utwórz nowe loginy o nazwach login1, login2 i login3 z domyślną bazą danych HR.

```
create login login1 with password = 'login1', DEFAULT_DATABASE = HR;  
create login login2 with password = 'login2', DEFAULT_DATABASE = HR;  
create login login3 with password = 'login3', DEFAULT_DATABASE = HR;
```

4	login1	2025-01-09 14:47:00.720	HR
5	login2	2025-01-09 14:47:00.723	HR
6	login3	2025-01-09 14:47:00.723	HR

### 2. Utwórz nowych użytkowników w bazie danych HR o nazwach user1, user2 i user3 przypisanych do loginów login1, login2 i login3.

```
create user user1 for login login1;  
create user user2 for login login2;  
create user user3 for login login3;
```

6	user1	SQL_USER	2025-01-09 14:47:24.680
7	user2	SQL_USER	2025-01-09 14:47:24.683
8	user3	SQL_USER	2025-01-09 14:47:24.683

### 3. Nadaj użytkownikom user1, user2 i user3 uprawnienia do wyświetlania departamentów.

```
grant select on departments to user1, user2, user3;
```

	UserName	ObjectName	PermissionName	PermissionState
1	user1	departments	SELECT	GRANT
2	user2	departments	SELECT	GRANT
3	user3	departments	SELECT	GRANT

### 4. Utwórz nową rolę o nazwie role1 i uczynź użytkownika user1 jej właścicielem.

```
create role role1 authorization user1;
```

	RoleName	RoleType	CreateDate	OwnerName
1	role1	DATABASE_ROLE	2025-01-09 14:56:19.273	user1

5. Nadaj użytkownikowi user1 uprawnienia do dodawania departamentów z możliwością dalszego przekazywania uprawnień.

```
grant insert on departments to user1 with grant option;
```

	GranteeName	ObjectName	PermissionName	PermissionState
1	user1	departments	INSERT	GRANT_WITH_GRANT_OPTION

6. Nadaj roli role1 uprawnienia do usuwania departamentów.

```
grant delete on departments to role1;
```

	GranteeName	ObjectName	PermissionName	PermissionState
1	role1	departments	DELETE	GRANT

7. Jako user1 nadaj użytkownikowi user2 uprawnienia do dodawania departamentów.

```
execute as user = 'user1';  
grant insert on departments to user2;
```

	GranteeName	ObjectName	PermissionName	PermissionState
1	user2	departments	INSERT	GRANT

8. Jako user1 nadaj użytkownikowi user2 rolę role1.

```
alter role role1 add member user2;
```

	RoleName	MemberName
1	role1	user2

9. Pozbaw użytkownika user1 uprawnień do dodawania departamentów.

```
revert;  
revoke insert on departments to user1 cascade;
```

	GranteeName	ObjectName	PermissionName	PermissionState
--	-------------	------------	----------------	-----------------

10. Pozbaw użytkownika user1 roli role1.

```
exec sp_droprolemember 'role1', 'user1';
```

	RoleName	MemberName
1	role1	user2

11. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 uprawnienia do dodawania departamentów?

```
execute as user = 'user2';
grant insert on departments to user3;
--nie mozna
```

## 12. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 rolę role1?

```
alter role role1 add member user3;
--nie mozna
```

```
Msg 15151, Level 16, State 1, Line 60
Cannot find the object 'departments', because it does not exist or you do not have permission.

Completion time: 2025-01-09T15:06:11.5852709+01:00
```

## 13. Czy jako użytkownik user1 możesz nadać użytkownikowi user3 rolę role1?

```
revert;
execute as user = 'user1';
alter role role1 add member user3;
--mozna
```

```
Msg 15151, Level 16, State 1, Line 64
Cannot alter the role 'role1', because it does not exist or you do not have permission.

Completion time: 2025-01-09T15:06:20.9130724+01:00
```

## Zadanie 2.

### 1. Utwórz tabelę o nazwie admin\_logs, która zawierać będzie następujące pola:

- log\_id typu numerycznego będące kluczem głównym,
- log\_desc typu znakowego,
- log\_data typu datowego (z uwzględnieniem czasu).

```
CREATE TABLE admin_logs ( log_id INT IDENTITY(1,1) PRIMARY KEY,
log_desc NVARCHAR(255), log_data DATETIME );
```

### 2. Utwórz nowe zadanie o nazwie login\_report, które:

- przy każdym uruchomieniu doda do tabeli admin\_logs wpis zawierający liczbę wszystkich loginów,
- poczynawszy od 1 grudnia 2024 roku będzie wykonywane codziennie o godzinie 2:00.

```
CREATE PROCEDURE insert_login_report
AS
BEGIN
    INSERT INTO admin_logs (log_desc, log_data)
    SELECT CONCAT('Liczba loginów: ', COUNT(*)), GETDATE()
    FROM sys.syslogins;
END;
```

Name: login\_report

Owner: DELL\Dominik ...

Category: [Uncategorized (Local)] ▼ ...

Description: No description available.

☒ Enabled

Job Step Properties - inserting

Select a page


General

Advanced


Connection


Server:  
DELL

Connection:  
DELL\Dominik

 [View connection properties](#)

Progress

 Ready

Script  Help

Step name:  
inserting

Type:  
Transact-SQL script (T-SQL)

Run as:

Database:  
HR

Command:  
exec insert\_login\_report

Open...

Select All

Copy

Paste

Parse

Previous

Next

OK

Cancel

**Job Schedule Properties - daily**

Name:  Jobs in Schedule

Schedule type:  ☒ Enabled

---

One-time occurrence

Date:  Time:

---

Frequency

Occurs:

Recurs every:  day(s)

---

Daily frequency

☒ Occurs once at:

☐ Occurs every:  hour(s)

Starting at:  Ending at:

---

Duration

Start date:  ☐ End date:  ☒ No end date:

---

Summary

Description:

OK Cancel Help

	log_id	log_desc	log_data
1	13	Liczba loginów: 16	2025-01-08 02:00:00.650
2	14	Liczba loginów: 16	2025-01-09 02:00:00.573

# PostgreSQL

## Zadanie 1.

1. Utwórz nowych użytkowników o nazwach user1, user2 i user3.

```
create user user1;
create user user2;
create user user3;
```

	username name
1	postgres
2	user1
3	user2
4	user3

**2 Nadaj użytkownikom user1, user2 i user3 uprawnienia do wyświetlania departamentów.**

```
grant select on table departments to user1, user2, user3;
```

	username name
1	postgres
2	user1
3	user2
4	user3

**3. Utwórz nową rolę o nazwie role1 i uczynź użytkownika user1 jej właścicielem.**

```
create role role1 with admin user1;
```

	role_name name	admin_role name
1	role1	user1

**4. Nadaj użytkownikowi user1 uprawnienia do dodawania departamentów z możliwością dalszego przekazywania uprawnień.**

```
grant insert on table departments to user1 with grant option;
--grant option umożliwia nadanie przywileju obiektowego innym
uzytkownikom (z grant lub bez)
--lub rolom
```

	grantee name	privilege_type character varying	is_grantable character varying (3)
7	postgres	TRIGGER	YES
8	user1	INSERT	YES

**5. Nadaj roli role1 uprawnienia do usuwania departamentów.**

```
grant delete on table departments to role1;
```

	grantee name	privilege_type character varying
1	role1	DELETE

6. Jako user1 nadaj użytkownikowi user2 uprawnienia do dodawania departamentów.

```
set role user1;  
grant insert on table departments to user2;
```

	grantee name	privilege_type character varying
1	user2	INSERT

7. Jako user1 nadaj użytkownikowi user2 rolę role1.

```
set role user1;  
grant role1 to user2;
```

	granted_role name	grantee name	grantor name
1	role1	user2	user1

8. Pozbaw użytkownika user1 uprawnień do dodawania departamentów.

```
reset role;  
revoke insert on table departments from user1 CASCADE;
```

	grantee name	privilege_type character varying	table_name name
1	user1	SELECT	departments

9. Pozbaw użytkownika user1 roli role1.

```
revoke role1 from user1 CASCADE;
```

	role_name name	member_name name

10. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 uprawnienia do dodawania departamentów?

```
set role user2;  
grant insert on table departments to user3;  
--nie mozna
```

```
OSTRZEŻENIE: nie przyznano żadnych uprawnień do "departments"  
GRANT
```

11. Czy jako użytkownik user2 możesz nadać użytkownikowi user3 rolę role1?

```
grant role1 to user3;
```



```
--nie mozna
```

```
ERROR: Only roles with the ADMIN option on role "role1" may grant this role.permission denied to grant role "role1"
```

```
Błąd: permission denied to grant role "role1"
```

## 12. Czy jako użytkownik user1 możesz nadać użytkownikowi user3 rolę role1?

```
reset role;  
set role user1;  
grant role1 to user3;  
--nie mozna
```

```
ERROR: Only roles with the ADMIN option on role "role1" may grant this role.permission denied to grant role "role1"
```

```
Błąd: permission denied to grant role "role1"
```

## Zadanie 2.

### 1. Utwórz tabelę o nazwie admin\_logs, która zawierać będzie następujące pola:

- log\_id typu numerycznego będące kluczem głównym,
- log\_desc typu znakowego,
- log\_data typu datowego (z uwzględnieniem czasu).

```
create table admin_logs(  
    log_id serial primary key,  
    log_desc varchar,  
    log_data timestamp  
)
```

### 2. Utwórz nowe zadanie o nazwie user\_report, które:

- przy każdym uruchomieniu doda do tabeli admin\_logs wpis zawierający liczbę wszystkich użytkowników,
- począwszy od 1 grudnia 2024 roku będzie wykonywane codziennie o godzinie 2:00.

```
INSERT INTO admin_logs (log_desc, log_data)  
SELECT 'Liczba wszystkich użytkowników: ' || COUNT(*), NOW()  
FROM pg_user;
```

GeneralStepsSchedulesSQL

Name

user\_report

Enabled?

Job class

Routine Maintenance

Please select a class to categorize the job. This option will not affect the way the job runs.

Host agent

Enter the hostname of a machine running pgAgent if you wish to ensure only that machine will run this job. Leave blank if any host may run the job.

Comment

Close

Reset

Save

GeneralStepsSchedulesSQL

Name

teststep

Enabled?

Kind

SQL

Connection type

Local

On error

Fail

General

Code

1

2

3

INSERT INTO admin\_logs (log\_desc, log\_data)

SELECT 'Liczba wszystkich użytkowników: ' || COUNT(\*), NOW()

FROM pg\_user;

Close

Reset

Save

+

Name	Enabled?	Start	End
<div><div><div></div><div></div></div><div>daily</div></div>	<div><div></div><div></div></div>	<div>2024-12-01 02:00:01<div></div></div>	<div>YYYY-MM-DD HH:m<div></div></div>

GeneralRepeatExceptions

Name

daily

Enabled?

Start

2024-12-01 02:00:00 +01:00

End

YYYY-MM-DD HH:mm:ss Z

Comment

i?

X CloseResetSave

Days

Week Days

Sunday x Monday x Tuesday x Wednesday x Thursday x Friday x Saturday x

Month Days

1st x 2nd x 3rd x 4th x 5th x 6th x 7th x 8th x 9th x 10th x 11th x 12th x 13th x 14th x 15th x 16th x 17th x 18th x 19th x 20th x 21st x 22nd x 23rd x 24th x 25th x 26th x 27th x 28th x 29th x 30th x 31st x Last day x

Months

January x February x March x April x May x June x July x August x September x October x November x December x

Times

Hours

02 x

Minutes

00 x

Close

Reset

Save

	log_id [PK] integer	log_desc character varying	log_data timestamp without time zone
1	10	Liczba wszystkich użytkowników: 1	2025-01-08 02:00:03.002824
2	11	Liczba wszystkich użytkowników: 1	2025-01-09 02:00:03.105581