Julia Ruszer 247775

Dominik Gałkowski 247659

Jan Śladowski 247806

Wiktor Żelechowski 247833

Microsoft SQL Server

Zadanie 1.

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Zadeklaruj zmienną o nazwie *salary*.
* Przypisz jej wartość pensji pracownika o identyfikatorze równym 100.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Obniż o 1000 zł wartość zmiennej *salary*.
* Ustaw pensję pracownika o identyfikatorze równym 100 na wartość zmiennej *salary*.
* Wyświetl wartość zmiennej *salary*.
* Zatwierdź transakcję.

begin transaction;

declare @salary numeric(8, 2);

set @salary = (select salary from employees where employee\_id = 100);

WAITFOR DELAY '00:00:15';

set @salary = @salary - 1000;

update employees set salary = @salary where employee\_id = 100;

select @salary;

commit transaction;

go

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Zadeklaruj zmienną o nazwie *salary*.
* Przypisz jej wartość pensji pracownika o identyfikatorze równym 100.
* Obniż o 2000 zł wartość zmiennej *salary*.
* Ustaw pensję pracownika o identyfikatorze równym 100 na wartość zmiennej *salary*.
* Wyświetl wartość zmiennej *salary*.
* Zatwierdź transakcję.

begin transaction;

declare @salary numeric(8, 2);

set @salary = (select salary from employees where employee\_id = 100);

set @salary = @salary - 2000;

update employees set salary = @salary where employee\_id = 100;

select @salary;

commit transaction;

go

**3. Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2 w ramach dwóch różnych sesji.**

sesja dla skryptu 1.



Sesja dla skryptu 2.

  
**4. Wyświetl pensję pracownika o identyfikatorze równym 100.**

select salary from employees where employee\_id = 100;

  
**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Utracone modyfikacje (ang. lost updates)

**6. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

Repeatable Read

Zadanie 2.

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Wyświetl pensję pracownika o identyfikatorze równym 100.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Wyświetl pensję pracownika o identyfikatorze równym 100.
* Zatwierdź transakcję.

begin transaction;

select salary from employees where employee\_id = 100;

WAITFOR DELAY '00:00:15';

select salary from employees where employee\_id = 100;

commit transaction;

go

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Podwyższ o 1000 zł pensję pracownika o identyfikatorze równym 100.
* Zatwierdź transakcję.

begin transaction;

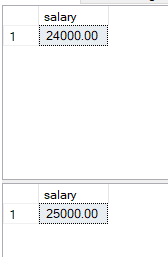
update employees set salary = salary + 1000 where employee\_id = 100;

commit transaction;

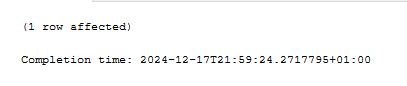
go

**3. Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2 w ramach dwóch różnych sesji.**

sesja dla skryptu 1.



Sesja dla skryptu 2.



**4. Wyświetl pensję pracownika o identyfikatorze równym 100.**

select salary from employees where employee\_id = 100;



**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Niepowtarzalne odczyty (ang. Nonrepeatable (fuzzy) reads/inconsistent analysis)

**6. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

Repeatable Read

Zadanie 3.

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Wyświetl wszystkich pracowników, którzy zostali zatrudnieni w grudniu.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Wyświetl wszystkich pracowników, którzy zostali zatrudnieni w grudniu.
* Zatwierdź transakcję.

begin transaction;

select \* from employees where MONTH(hire\_date) = 12;

WAITFOR DELAY '00:00:15';

select \* from employees where MONTH(hire\_date) = 12;

commit transaction;

go

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Dodaj nowego pracownika, który został zatrudniony 1 grudnia 2022 roku.
* Zatwierdź transakcję.

begin transaction;

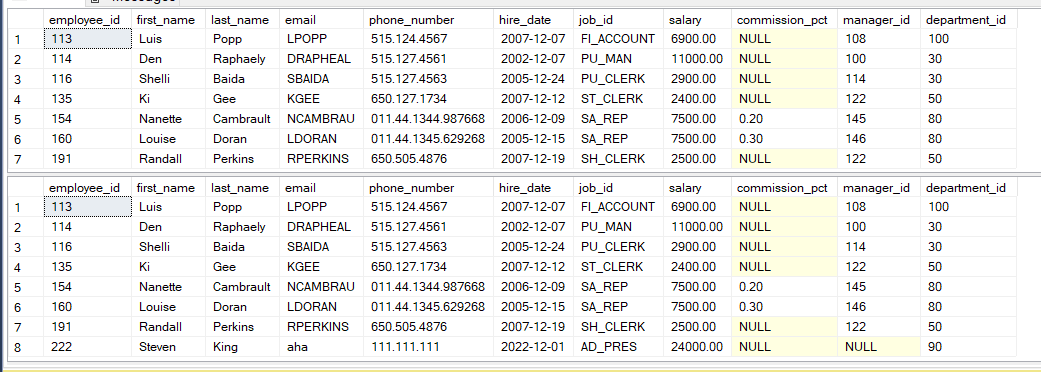
INSERT INTO employees VALUES ( 222 , 'Steven' , 'King' , 'aha' , '111.111.111' , CONVERT(DATE, '01-12-2022', 105) , 'AD\_PRES' , 24000 , NULL , NULL , 90 );

commit transaction;

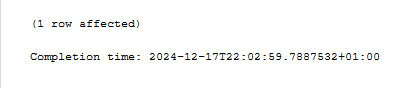
go

**3. Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2.**

sesja dla skryptu 1.



sesja dla skryptu 2.

  
**4. Wyświetl wszystkich pracowników, którzy zostali zatrudnieni w grudniu.**

select \* from employees where MONTH(hire\_date) = 12;

**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Odczyt widmo (ang. Phantom reads/phantoms)

**6. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

Serializable

Zadanie 4.

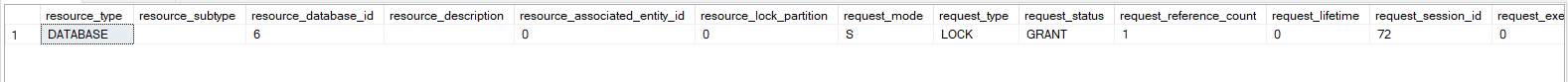
1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**

begin transaction;

select \*

from sys.dm\_tran\_locks

where request\_session\_id = @@spid

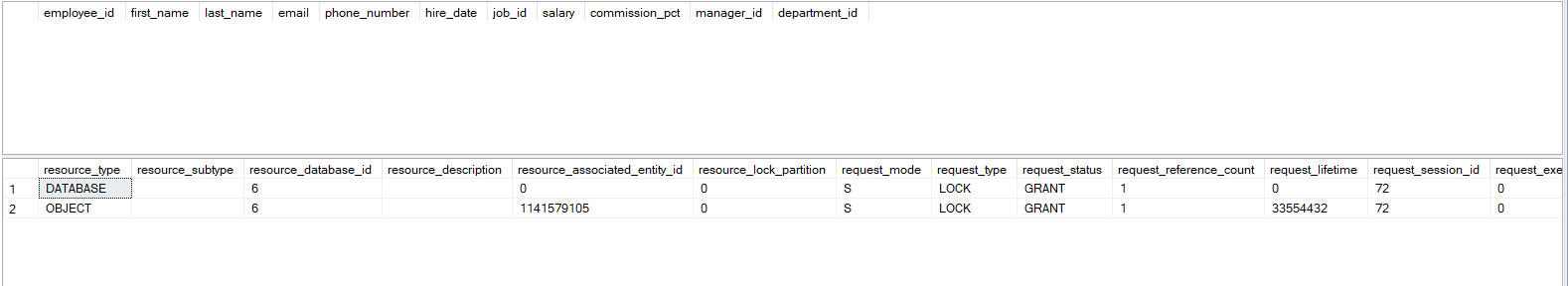
  
**3. W ramach sesji nr 1 na cały czas trwania transakcji załóż blokadę dzieloną na tabeli pracowników.**  
**4. W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**

select\* from employees WITH (HOLDLOCK, tablock) where employee\_id = null;

select \*

from sys.dm\_tran\_locks

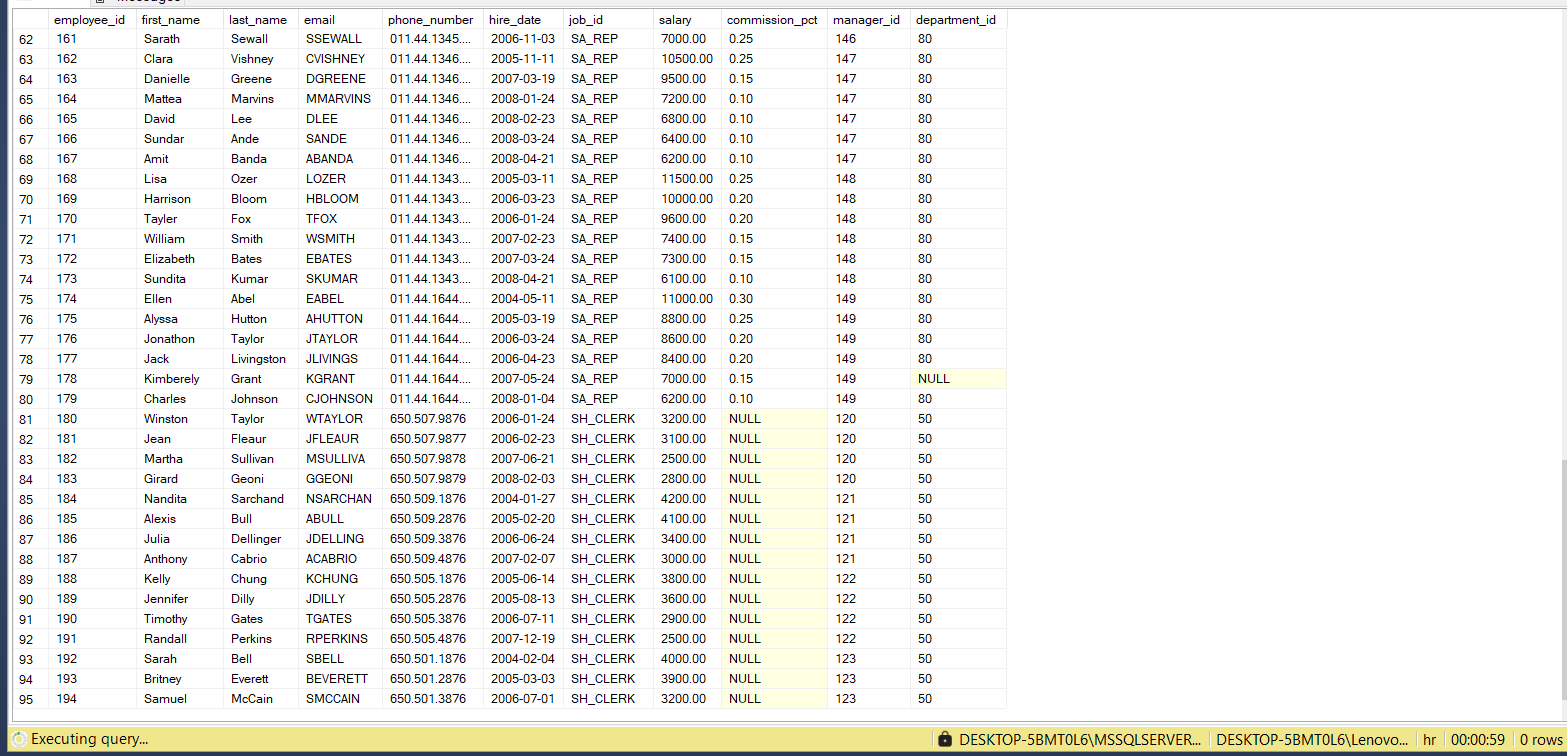
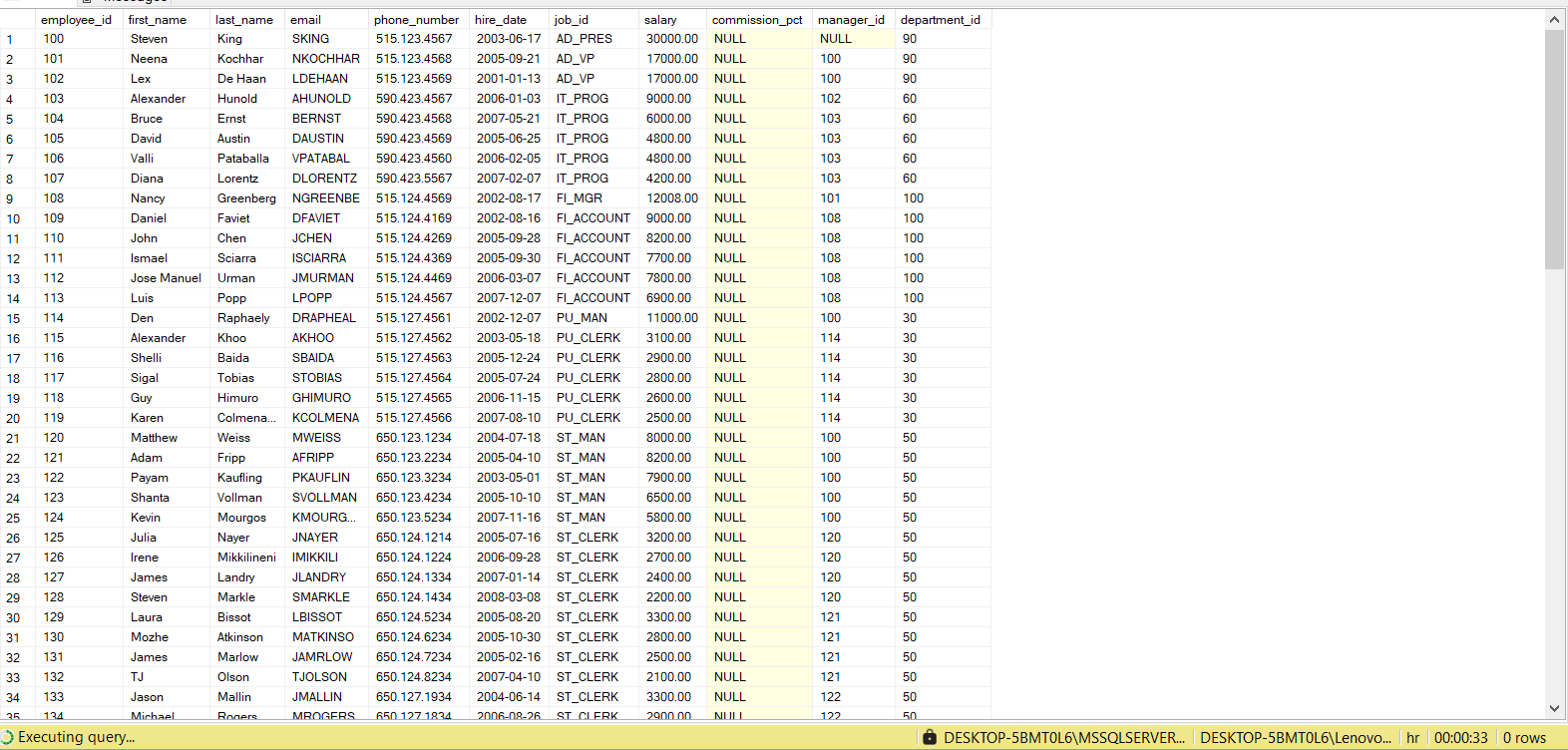
where request\_session\_id = @@spid

  
**5. W ramach sesji nr 2 rozpocznij transakcję.**  
**6. W ramach sesji nr 2 wyświetl wszystkich pracowników.**  
**7. W ramach sesji nr 2 podwyższ o 100 zł pensję wszystkich pracowników.**

begin transaction;

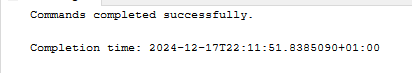
select \* from employees;

update employees set salary = salary + 100;

  
**8. W ramach sesji nr 1 zatwierdź transakcję.**

commit transaction;

sesja dla skryptu 1.



sesja dla skryptu 2.

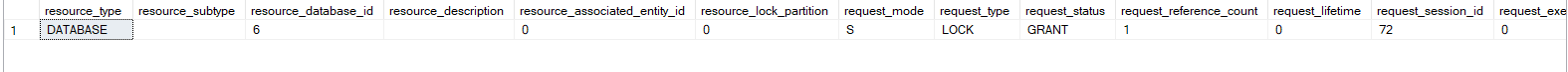


**9. W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**

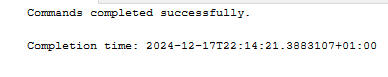
select \*

from sys.dm\_tran\_locks

where request\_session\_id = @@spid

  
**10. W ramach sesji nr 2 wycofaj transakcję.**

rollback transaction;



**11. W którym momencie udało Ci się podwyższyć pensję wszystkich pracowników w ramach sesji nr 2?**

Po zatwierdzeniu transakcji w kroku 8.

Zadanie 5.

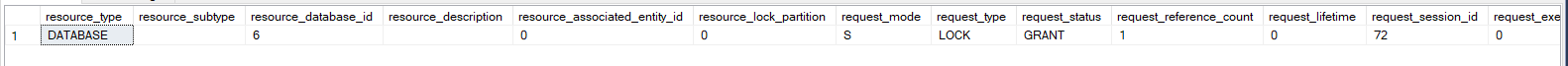
1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**

begin transaction;

select \*

from sys.dm\_tran\_locks

where request\_session\_id = @@spid

  
**3. W ramach sesji nr 1 na cały czas trwania transakcji załóż blokadę wyłączną na tabeli pracowników.**

**4. W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**  
**5. W ramach sesji nr 1 podwyższ o 100 zł pensję wszystkich pracowników.**

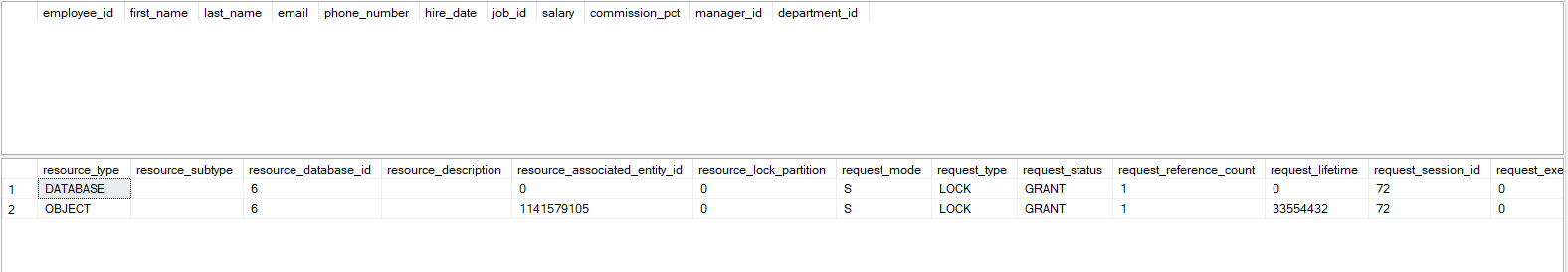
select\* from employees WITH (HOLDLOCK, tablock, xlock) where employee\_id = null;

select \*

from sys.dm\_tran\_locks

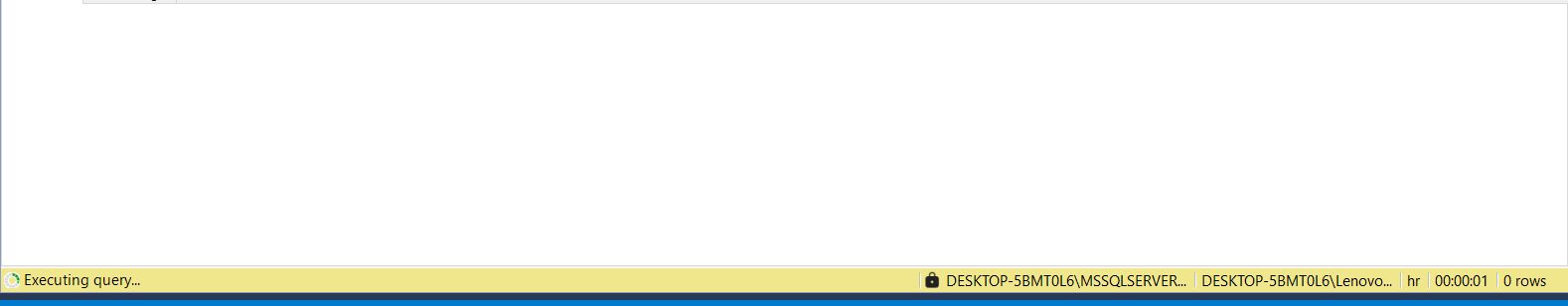
where request\_session\_id = @@spid

update employees set salary = salary + 100;

  
**6. W ramach sesji nr 2 rozpocznij transakcję.**  
**7. W ramach sesji nr 2 wyświetl wszystkich pracowników.**

begin transaction;

select \* from employees;

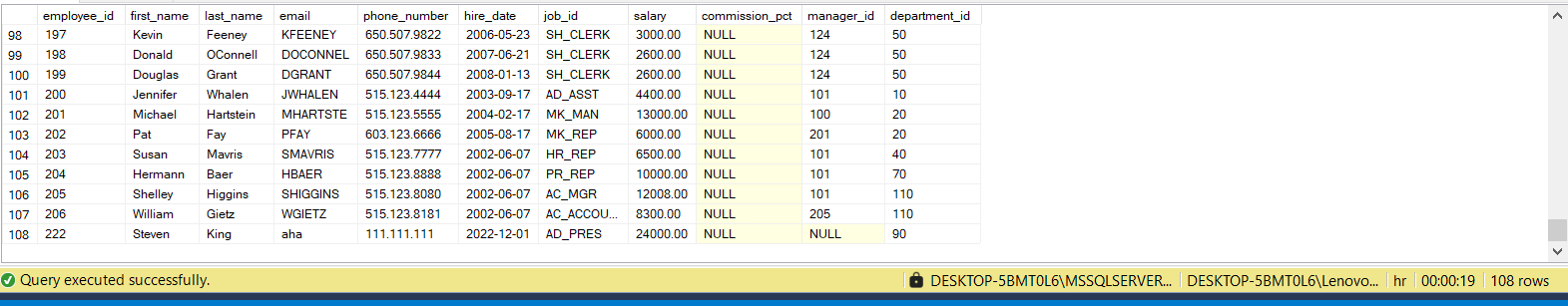
  
**8. W ramach sesji nr 1 wycofaj transakcję.**

rollback transaction;

sesja 1.



sesja 2.

  
**9. W ramach sesji nr 1 wyświetl listę blokad, za które odpowiedzialna jest ta sesja.**

select \*

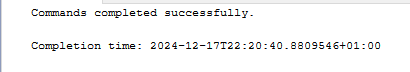
from sys.dm\_tran\_locks

where request\_session\_id = @@spid



**10. W ramach sesji nr 2 zatwierdź transakcję.**

commit transaction;



**11. W którym momencie udało Ci się wyświetlić wszystkich pracowników w ramach sesji nr 2?**

Po wycofaniu transakcji w kroku 8.

Zadanie 6.

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Ustaw maksymalną pensję na stanowisku President na 30000 zł.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Ustaw pensję pracownika o identyfikatorze równym 100 na 30000 zł.
* Zatwierdź transakcję.

begin transaction;

update jobs set max\_salary = 30000 where job\_title = 'President';

waitfor delay '00:00:15';

update employees set salary = 30000 where employee\_id = 100;

commit transaction;

go

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Ustaw pensję pracownika o identyfikatorze równym 100 na 30000 zł.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Ustaw maksymalną pensję na stanowisku President na 30000 zł.
* Zatwierdź transakcję.

begin transaction;

update employees set salary = 30000 where employee\_id = 100;

waitfor delay '00:00:15';

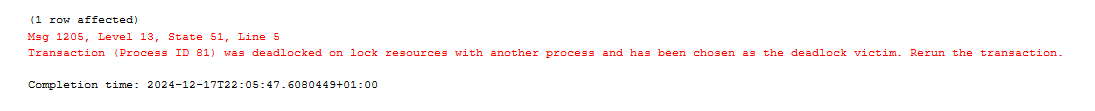
update jobs set max\_salary = 30000 where job\_title = 'President';

commit transaction;

go

1. **Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2 w ramach dwóch różnych sesji.**

**4. Poczekaj, aż zostanie wyświetlony odpowiedni komunikat.**



**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Zakleszczenie

PostgreSQL

Zadanie 1.

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Zadeklaruj zmienną o nazwie *salary*.
* Przypisz jej wartość maksymalnej pensji na stanowisku Programmer.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Obniż o 1000 zł wartość zmiennej *salary*.
* Ustaw maksymalną pensję na stanowisku Programmer na wartość zmiennej *salary*.
* Wyświetl wartość zmiennej *salary*.
* Zatwierdź transakcję.

begin;

do $$

declare

salary numeric(6);

begin

select max\_salary into salary

from jobs where job\_title = 'Programmer';

perform pg\_sleep(15);

salary := salary - 1000;

update jobs set max\_salary = salary where job\_title = 'Programmer';

raise notice 'salary = %', salary;

end;

$$ LANGUAGE plpgsql;

commit;

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Zadeklaruj zmienną o nazwie *salary*.
* Przypisz jej wartość maksymalnej pensji na stanowisku Programmer.
* Obniż o 2000 zł wartość zmiennej *salary*.
* Ustaw maksymalną pensję na stanowisku Programmer na wartość zmiennej *salary*.
* Wyświetl wartość zmiennej *salary*.
* Zatwierdź transakcję.

begin;

do $$

declare

salary numeric(6);

begin

select max\_salary into salary

from jobs where job\_title = 'Programmer';

salary := salary - 2000;

update jobs set max\_salary = salary where job\_title = 'Programmer';

raise notice 'salary = %', salary;

end;

$$ LANGUAGE plpgsql;

commit;

**3. Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2 w ramach dwóch różnych sesji.**

Sesja dla skryptu 1.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Sesja dla skryptu 2.

Obraz zawierający tekst, zrzut ekranu, Czcionka, biały

Opis wygenerowany automatycznie

**4. Wyświetl maksymalną pensję na stanowisku Programmer.**

select max\_salary from jobs where job\_title = 'Programmer';

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Utracone modyfikacje (ang. lost updates)

**6. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

Repeatable Read

**Zadanie 2.**

1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl maksymalną pensję na stanowisku Programmer.**

begin;

select max\_salary from jobs where job\_title = 'Programmer';

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie**

**3. W ramach sesji nr 2 rozpocznij transakcję.  
4. W ramach sesji nr 2 podwyższ o 1000 zł maksymalną pensję na stanowisku Programmer.**

begin;

update jobs set max\_salary = max\_salary + 1000

where job\_title = 'Programmer';

**Obraz zawierający tekst, Czcionka, biały, zrzut ekranu

Opis wygenerowany automatycznie**

**5. W ramach sesji nr 2 zatwierdź transakcję.**

commit;

**Obraz zawierający tekst, Czcionka, biały, algebra

Opis wygenerowany automatycznie  
6. W ramach sesji nr 1 wyświetl maksymalną pensję na stanowisku Programmer.**

select max\_salary from jobs where job\_title = 'Programmer';

**Obraz zawierający tekst, Czcionka, zrzut ekranu, linia

Opis wygenerowany automatycznie  
7. W ramach sesji nr 1 zatwierdź transakcję.**

commit;

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie  
8. W ramach sesji nr 1 wyświetl maksymalną pensję na stanowisku Programmer.**

select max\_salary from jobs where job\_title = 'Programmer';

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie**

**9. Jakie zjawisko zaprezentowano w tym zadaniu?**

Niepowtarzalne odczyty (ang. Nonrepeatable (fuzzy) reads/inconsistent analysis)

**10. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

Repeatable Read

**Zadanie 3.**

1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl wszystkie stanowiska, na których minimalna pensja wynosi 2500 zł.**

begin;

select \* from jobs where min\_salary = 2500;

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie**

**3. W ramach sesji nr 2 rozpocznij transakcję.  
4. W ramach sesji nr 2 dodaj nowe stanowisko, na którym minimalna pensja wynosi 2500 zł.**

begin;

INSERT INTO jobs VALUES ( 'NOWE' , 'NNN' , 2500 , 4000 );

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie  
5. W ramach sesji nr 2 zatwierdź transakcję.**

commit;

**Obraz zawierający tekst, Czcionka, biały, linia

Opis wygenerowany automatycznie**

**6. W ramach sesji nr 1 wyświetl wszystkie stanowiska, na których minimalna pensja wynosi 2500 zł.**

select \* from jobs where min\_salary = 2500;

**Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie**

**7. W ramach sesji nr 1 zatwierdź transakcję.**

commit;

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie  
8. W ramach sesji nr 1 wyświetl wszystkie stanowiska, na których minimalna pensja wynosi 2500 zł.**

select \* from jobs where min\_salary = 2500;

**Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie**

**9. Jakie zjawisko zaprezentowano w tym zadaniu?**

Odczyt widmo (ang. Phantom reads/phantoms)

**10. Na jakim minimalnym poziomie izolacji nie mogłoby dojść do jego wystąpienia?**

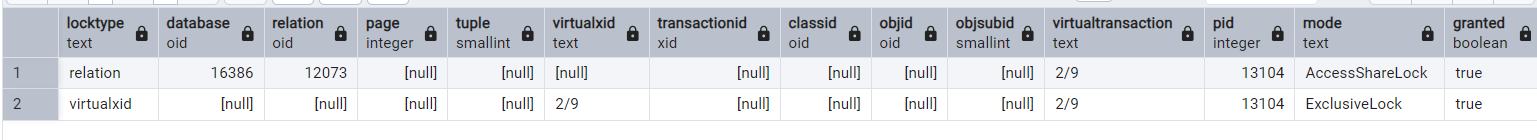
Repeatable Read

**Zadanie 4.**

1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

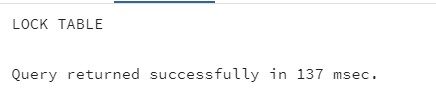
begin;

SELECT \* FROM pg\_locks;



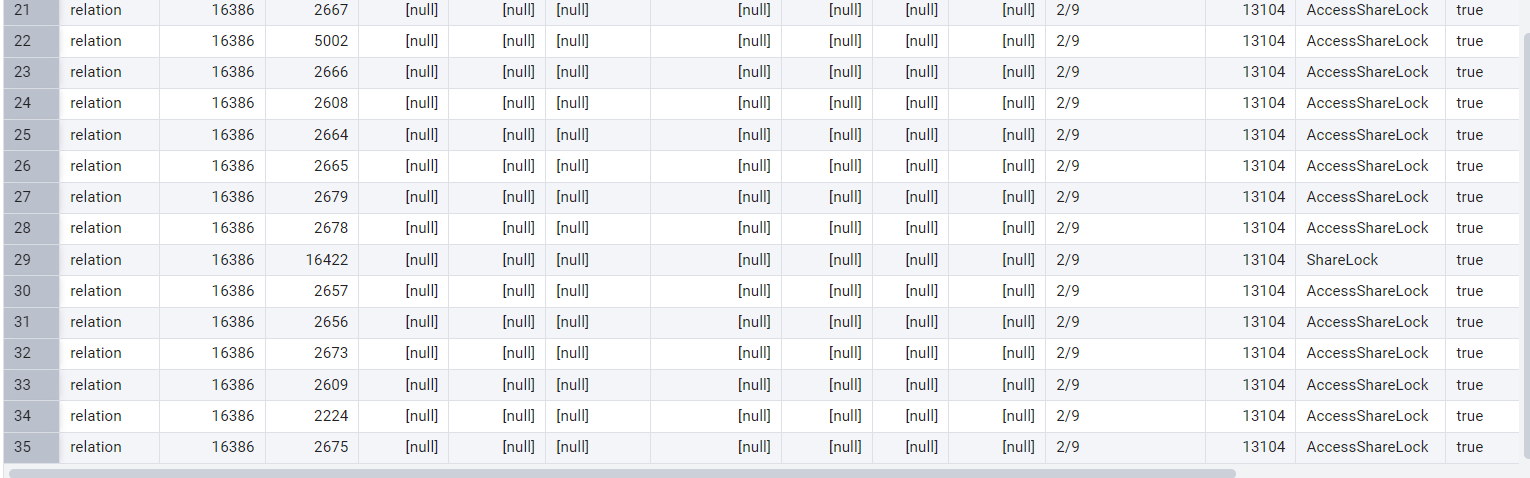
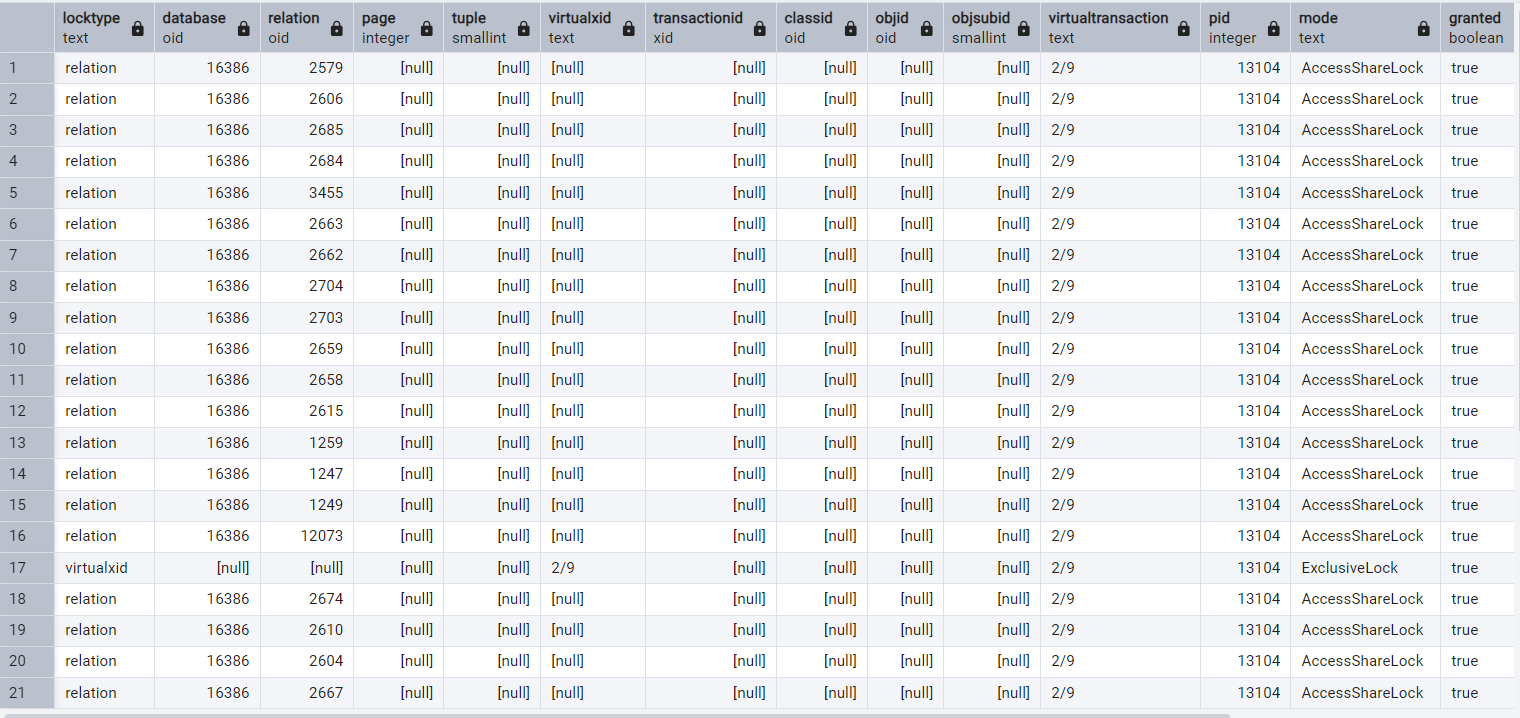
1. **W ramach sesji nr 1 na cały czas trwania transakcji załóż blokadę dzieloną na tabeli stanowisk.**

LOCK TABLE jobs IN SHARE MODE;



**4. W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

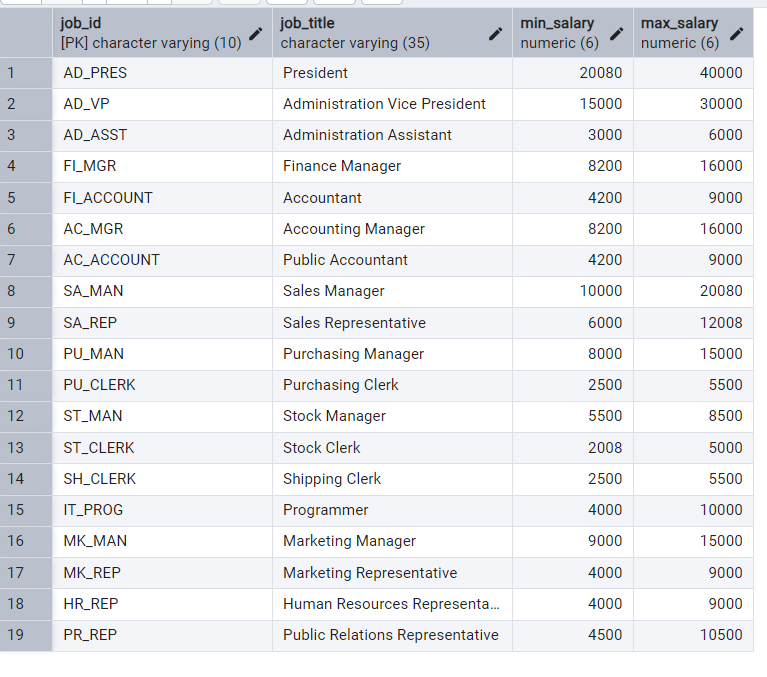
SELECT \* FROM pg\_locks;



**5. W ramach sesji nr 2 rozpocznij transakcję.  
6. W ramach sesji nr 2 wyświetl wszystkie stanowiska.**

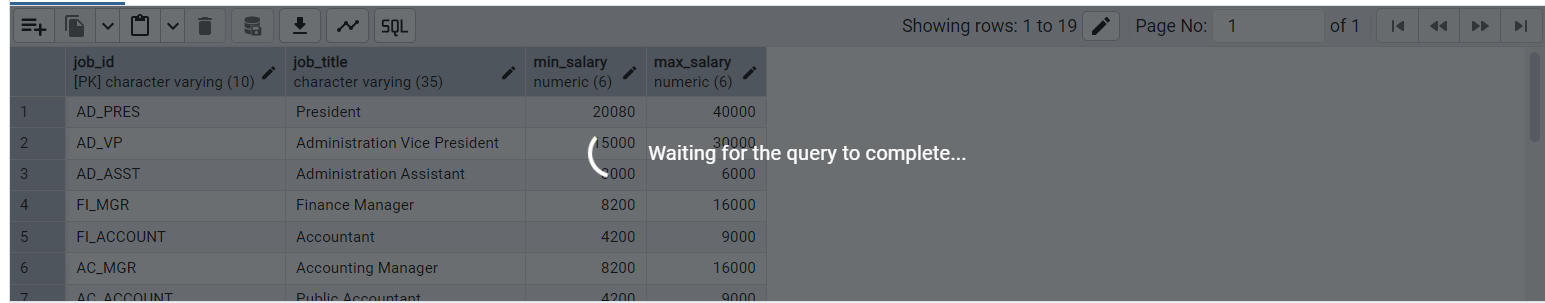
begin;

select \* from jobs;



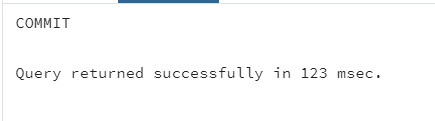
**7. W ramach sesji nr 2 podwyższ o 100 zł maksymalną pensję na wszystkich stanowiskach.**

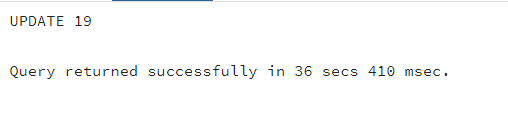
update jobs set max\_salary = max\_salary + 100;



**8. W ramach sesji nr 1 zatwierdź transakcję.**

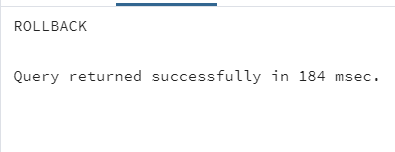
commit;



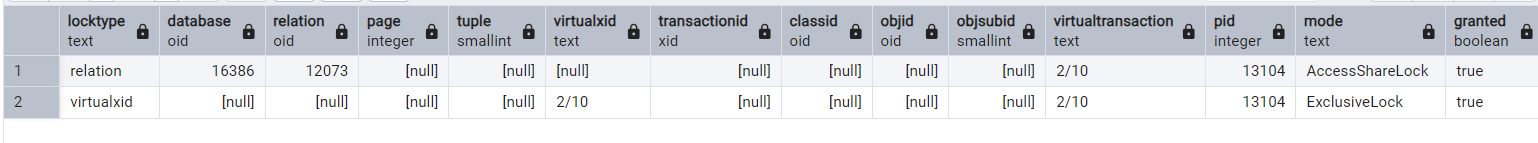


**9. W ramach sesji nr 2 wycofaj transakcję.**

rollback;

  
**10. W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

SELECT \* FROM pg\_locks;



**11. W którym momencie udało Ci się podwyższyć maksymalną pensję na wszystkich stanowiskach w ramach sesji nr 2?**

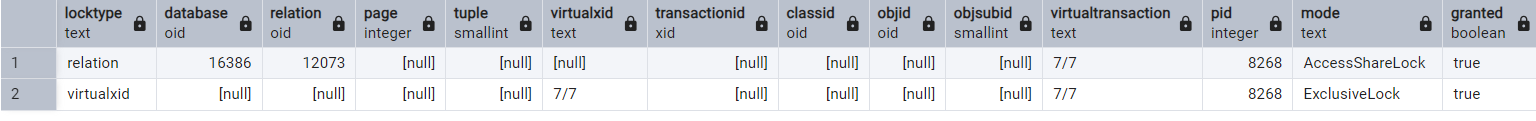
Po zatwierdzeniu transakcji w kroku 8.

**Zadanie 5.**

1. **W ramach sesji nr 1 rozpocznij transakcję.**
2. **W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

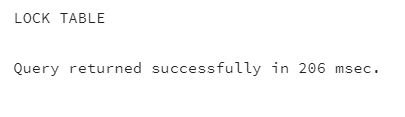
begin;

SELECT \* FROM pg\_locks;



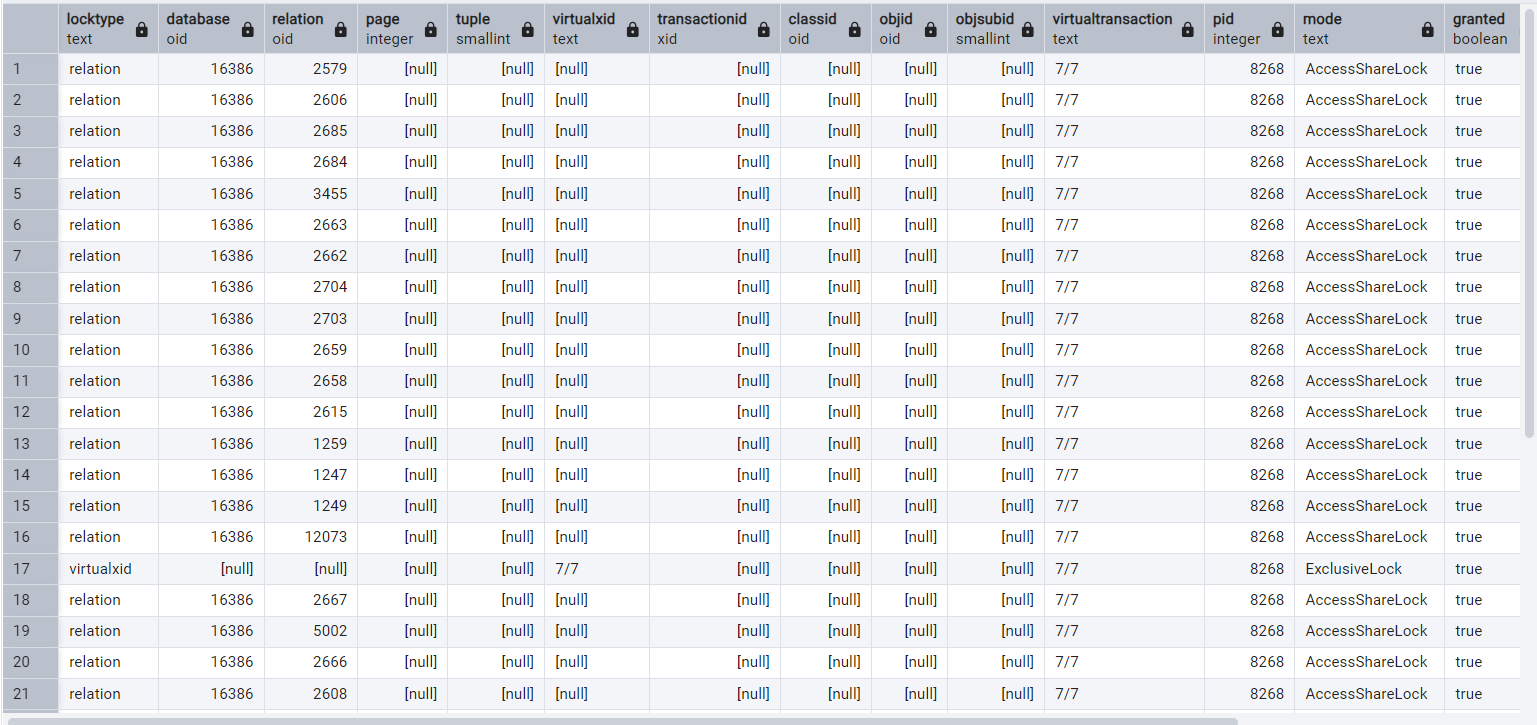
1. **W ramach sesji nr 1 na cały czas trwania transakcji załóż blokadę wyłączną na tabeli stanowisk.**

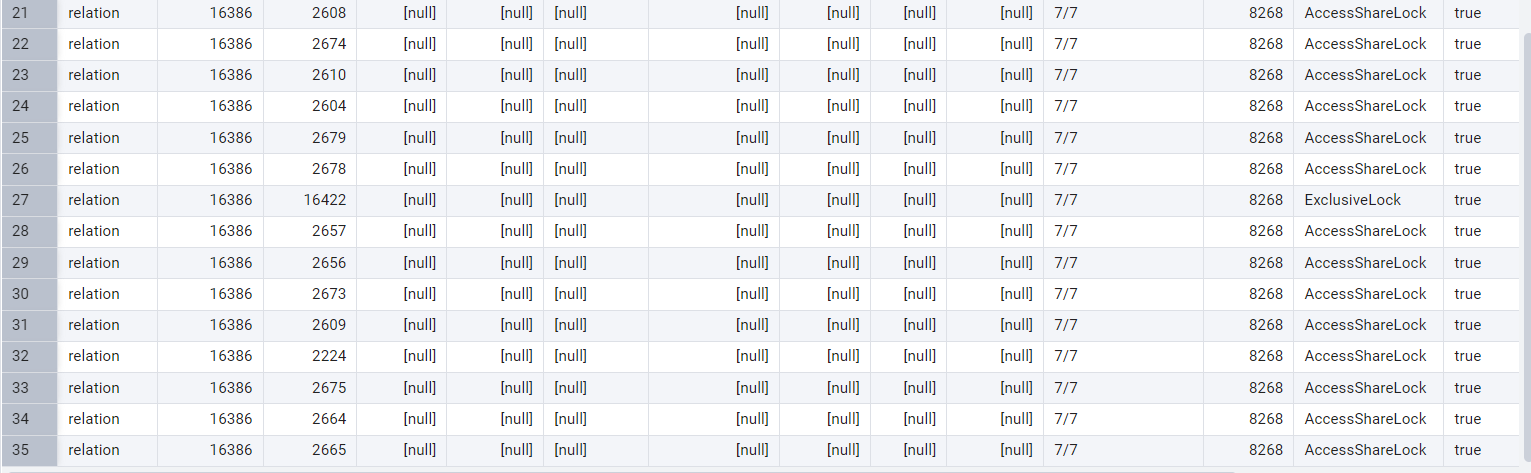
LOCK TABLE jobs IN EXCLUSIVE MODE;



1. **W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

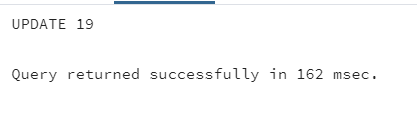
SELECT \* FROM pg\_locks;





1. **W ramach sesji nr 1 podwyższ o 100 zł maksymalną pensję na wszystkich stanowiskach.**

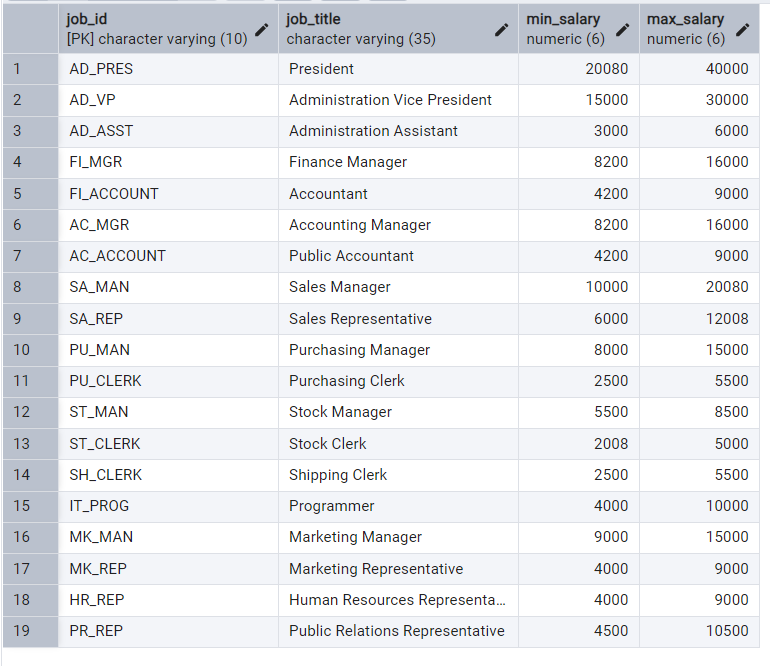
update jobs set max\_salary = max\_salary + 100;



1. **W ramach sesji nr 2 rozpocznij transakcję.**
2. **W ramach sesji nr 2 wyświetl wszystkie stanowiska.**

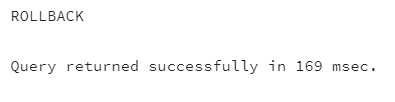
begin;

select \* from jobs;



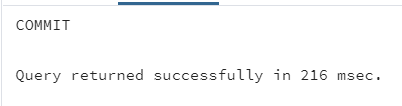
1. **W ramach sesji nr 1 wycofaj transakcję.**

rollback;



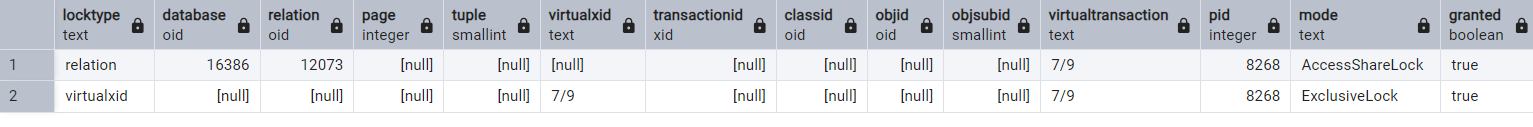
1. **W ramach sesji nr 2 zatwierdź transakcję.**

commit;



1. **W ramach sesji nr 1 wyświetl listę wszystkich blokad.**

SELECT \* FROM pg\_locks;



**11. W którym momencie udało Ci się wyświetlić wszystkie stanowiska w ramach sesji nr 2?**

Przy wykonywaniu zapytania select w kroku 7.

**Zadanie 6.**

**1. Przygotuj skrypt nr 1:**

* Rozpocznij transakcję.
* Ustaw minimalną pensję na stanowisku Programmer na 5000 zł.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Ustaw pensję pracownika o identyfikatorze równym 103 na 5000 zł.
* Zatwierdź transakcję.

begin;

do $$

begin

update jobs set min\_salary = 5000 where job\_title = 'Programmer';

perform pg\_sleep(15);

update employees set salary = 5000 where employee\_id = 103;

end;

$$ LANGUAGE plpgsql;

commit;

**2. Przygotuj skrypt nr 2:**

* Rozpocznij transakcję.
* Ustaw pensję pracownika o identyfikatorze równym 103 na 5000 zł.
* Wstrzymaj wykonanie skryptu na 15 sekund.
* Ustaw minimalną pensję na stanowisku Programmer na 5000 zł.
* Zatwierdź transakcję.

begin;

do $$

begin

update employees set salary = 5000 where employee\_id = 103;

perform pg\_sleep(15);

update jobs set min\_salary = 5000 where job\_title = 'Programmer';

end;

$$ LANGUAGE plpgsql;

commit;

**3. Uruchom skrypt nr 1 i zaraz po nim skrypt nr 2 w ramach dwóch różnych sesji.**

**4. Poczekaj, aż zostanie wyświetlony odpowiedni komunikat.**

**Obraz zawierający tekst, Czcionka, zrzut ekranu

Opis wygenerowany automatycznie**

**5. Jakie zjawisko zaprezentowano w tym zadaniu?**

Zakleszczenie