

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт
з лабораторної роботи №5 з дисципліни
«Бази даних»

«Основи програмування з використанням мови SQL. Збережені процедури.
Курсори. Створення, програмування та керування тригерами.»

Варіант 3

Виконав студент ІП-13 Бондаренко Максим Вікторович
(шифр, прізвище, ім'я, по батькові)

Перевірів Марченко Олена Іванівна
(прізвище, ім'я, по батькові)

Лабораторна робота №5

Мета:

- Вивчити правила побудови ідентифікаторів, правила визначення змінних та типів. Визначити правила роботи з циклами та умовними конструкціями, роботу зі змінними типу Table.
- Вивчити синтаксис та семантику функцій та збережених процедур, способів їх ідентифікації, методів визначення та специфікації параметрів та повертаємих значень, виклик функцій та збережених процедур.
- Застосування команд для створення, зміни та видалення як скалярних, так і табличних функцій, збережених процедур.
- Вивчити призначення та типи курсорів, синтаксис та семантику команд мови SQL для створення курсорів, вибірки даних з курсорів, зміни даних із застосуванням курсорів.
- Вивчити призначення та типи тригерів, умов їх активації, синтаксису та семантики для їх створення, модифікації, перейменування, програмування та видалення.

Постановка задачі

При виконанні лабораторної роботи необхідно виконати наступні дії:

1) Збережені процедури:

- а. запит для створення тимчасової таблиці через змінну типу TABLE;
- б. запит з використанням умовної конструкції IF;
- в. запит з використанням циклу WHILE;
- г. створення процедури без параметрів;
- д. створення процедури з вхідним параметром;
- е. створення процедури з вхідним параметром та RETURN;
- ж. створення процедури оновлення даних в деякій таблиці БД;
- з. створення процедури, в котрій робиться вибірка даних.

2) Функції:

- а. створити функцію, котра повертає деяке скалярне значення;
- б. створити функцію, котра повертає таблицю з динамічним набором стовпців;
- в. створити функцію, котра повертає таблицю заданої структури.

3) Робота з курсорами:

- а. створити курсор;
- б. відкрити курсор;
- в. вибірка даних, робота з курсорами.

4) Робота з тригерами:

- a. створити тригер, котрий буде спрацьовувати при видаленні даних;
- b. створити тригер, котрий буде спрацьовувати при модифікації даних;
- c. створити тригер, котрий буде спрацьовувати при додаванні даних.

Виконання завдання

```
use db_labs;

drop procedure if exists create_temp_table;
create procedure create_temp_table()
begin
    drop table if exists temp_tbl;
    create temporary table temp_tbl as
        select *
        from client
        limit 10;
end;
call create_temp_table();

drop procedure if exists if_condition;
create procedure if_condition()
begin
    select e.first_name,
           e.last_name,
           c.salary,
           if(c.salary >= 3500, 'Big salary', 'Small salary') as salary_level
    from employee as e, contract as c
    where e.id = c.employee_id;
end;
call if_condition();

drop procedure if exists while_loop;
create procedure while_loop()
begin
    declare var int default 0;
    while var < 5 do
        select var;
        set var = var + 1;
    end while;
end;
call while_loop();

drop procedure if exists no_params;
create procedure no_params()
begin
    select *
    from client
    where LENGTH(payment_card_number) < 16;
end;
call no_params();

drop procedure if exists with_params;
create procedure with_params(in sex char)
begin
    select *
    from employee as e
    where e.sex = sex;
end;
call with_params('F');

drop procedure if exists in_out_procedure;
create procedure in_out_procedure(in value int, out res int)
begin
    select sum(salary)
    from contract
    where salary < value
```

«Бази даних»

```
    into res;

    select res;
end;
call in_out_procedure(4000, @res);

drop procedure if exists data_update;
create procedure data_update()
begin
    update contract
    set salary = salary + 500
    where start_date < date('2022-02-01');
end;
call data_update();

drop procedure if exists data_select;
create procedure data_select()
begin
    select month, sum(amount_of_payment)
    from monthly_payments
    group by month;
end;
call data_select();

drop function if exists scalar_function;
create function scalar_function(client_id int) returns decimal
deterministic
begin
    declare total decimal;
    select sum(amount_of_payment)
    from monthly_payments as mp
    where mp.client_id = client_id
    into total;
    return total;
end;
select scalar_function(1);

drop procedure if exists dynamic_columns;
create procedure dynamic_columns(in flag bool)
begin
    if flag = true then
        select concat(first_name, ' ', last_name), address, phone_number
        from client;
    else
        select *
        from client;
    end if;
end;
call dynamic_columns(true);

drop procedure if exists curs;
create procedure curs()
begin
    declare name varchar(50);
    declare i int default 0;
    declare cursor_client CURSOR FOR
        select concat(first_name, ' ', last_name)
        from client;

    open cursor_client;

    while i < (select count(*) from client) do
        fetch cursor_client into name;
        select name;
        set i = i + 1;
    end while;
end;
```

```
call curs();

drop trigger if exists trigger_on_delete;
create trigger trigger_on_delete before delete on client for each row begin
    if OLD.id in (select client_id from treaty) then
        signal sqlstate '45000'
        set message_text = 'You cant delete the user with active retail
outlet';
    end if;
end;
delete from client where id = 2;

drop trigger if exists trigger_on_update;
create trigger trigger_on_update before update on contract for each row
begin
    if new.salary > 6000 then
        set NEW.salary = 6000;
    end if;
end;
update contract
set salary = 7000
where id = 1;

drop trigger if exists trigger_on_insert;
create trigger trigger_on_insert before insert on contract for each row
begin
    if new.salary < 1000 then
        set NEW.salary = 1000;
    end if;
end;
insert into contract (salary)
values (500);
```

Висновок:

Протягом даної лабораторної роботи я вивчив правила побудови ідентифікаторів, правила визначення змінних та типів, визначив правила роботи з циклами та умовними конструкціями, вивчив синтаксис та семантику функцій та збережених процедур, способів їх ідентифікації, методів визначення та специфікації параметрів та повертаємих значень, виклик функцій та збережених процедур. Застосовував команди для створення, зміни та видалення як скалярних, так і табличних функцій, збережених процедур, вивчив призначення та типи курсорів, синтаксис та семантику команд мови SQL для створення курсорів, вибірки даних з курсорів, зміни даних із застосуванням курсорів, вивчив призначення та типи тригерів, умов їх активації, синтаксису та семантики для їх створення, модифікації, перейменування, програмування та видалення.