**ЛАБОРАТОРНА РОБОТА № 12**

з дисципліни

«Організація баз даних»

Шейко Р. О.

Групи КІ-21-2

**Завдання на лабораторну роботу**

1. Создать таблицу, которая будет хранить следующую информацию о группе:

· идентификатор группы (использовать как первичный ключ) – group\_id,

· факультет – department,

· число студентов в группе – students\_count,

· вложенную таблицу - lectures -, включающую следующие данные о читаемых лекциях:

o название лекции, читаемой в данной группе - subject,

o данные преподавателя, читающего лекцию (фамилия) - lecturer,

o количество часов, выделенных для преподавания данной лекции – read\_hours.

2. Написать процедуру для просмотра данных о лекциях, читаемых каждой группе. Запускать ее на выполнение после каждого из следующих пунктов задания.

3. Ввести данные для двух групп любых факультетов. Для первой группы данные для вложенной таблицы ввести одновременно с данными для основной таблицы. Для второй группы часть данных ввести с помощью объектной переменной, а часть с помощью оператора THE.

4. Используя оператор THE, изменить данные какого-либо преподавателя, читающего лекции второй группе.

5. Удалить все данные о лекциях, читаемых первой группе. Проверить, будет ли вложенная таблица, данные которой были удалены, NULL-таблицей (будет ли выполняться условие lectures IS NULL).

**Ілюстрації до роботи:**

Зображення, що містить текст, знімок екрана, програмне забезпечення, монітор

Автоматично згенерований опис

Рис 1. Результат

***Код до завдання:***

CREATE TYPE LectureType AS OBJECT (

subject VARCHAR2(50),

lecturer VARCHAR2(50),

read\_hours NUMBER

);

CREATE TYPE LectureTableType AS TABLE OF LectureType;

CREATE TABLE GroupTable (

group\_id NUMBER PRIMARY KEY,

department VARCHAR2(50),

students\_count NUMBER,

lectures LectureTableType

);

CREATE PROCEDURE ViewLectures(group\_id\_in IN NUMBER) IS

lectures\_table LectureTableType;

BEGIN

SELECT lectures INTO lectures\_table

FROM GroupTable

WHERE group\_id = group\_id\_in;

DBMS\_OUTPUT.PUT\_LINE('Lectures for group ' || group\_id\_in || ':');

FOR i IN 1..lectures\_table.COUNT LOOP

DBMS\_OUTPUT.PUT\_LINE(

'Subject: ' || lectures\_table(i).subject ||

', Lecturer: ' || lectures\_table(i).lecturer ||

', Read hours: ' || lectures\_table(i).read\_hours

);

END LOOP;

END;

-- Ввод данных для первой группы

DECLARE

lectures\_table LectureTableType := LectureTableType(

LectureType('SQL', 'Hvenko', 20),

LectureType('C', 'Dolonko', 25)

);

BEGIN

INSERT INTO GroupTable VALUES (1, 'Mathematics', 30, lectures\_table);

END;

-- Ввод данных для второй группы

DECLARE

lectures\_table LectureTableType := LectureTableType();

BEGIN

lectures\_table.EXTEND;

lectures\_table(lectures\_table.LAST) := LectureType('Biology', 'Sidorov', 30);

lectures\_table.EXTEND;

lectures\_table(lectures\_table.LAST) := LectureType('Chemistry', 'Kuznetsov', 30);

INSERT INTO GroupTable VALUES (2, 'Biology', 25, lectures\_table);

END;

-- Изменение данных для второй группы с помощью оператора THE

DECLARE

lectures\_table LectureTableType;

BEGIN

SELECT lectures INTO lectures\_table

FROM GroupTable

WHERE group\_id = 2;

lectures\_table(2) := THE LectureType(

'Chemistry', 'Ivanova', 30

);

UPDATE GroupTable

SET lectures = lectures\_table

WHERE group\_id = 2;

END;

-- Удаление данных о лекциях, читаемых первой группе

DELETE FROM TABLE (

SELECT lectures

FROM GroupTable

WHERE group\_id = 1

);

-- Проверка, что вложенная таблица пуста

DECLARE

lectures\_table LectureTableType;

BEGIN

SELECT lectures INTO lectures\_table

FROM GroupTable

WHERE group\_id = 1;

IF lectures\_table IS NULL THEN

DBMS\_OUTPUT.PUT\_LINE('Lectures for group 1 is empty');

END IF;

END;