**ЛАБОРАТОРНА РОБОТА № 8**

з дисципліни

«Організація баз даних»

Шейко Р. О.

Групи КІ-21-2

**Завдання на лабораторну роботу**

1. Создать хранимую функцию FIO\_QUANTITY\_MARK (, которая принимает параметры: id предмета (sb\_id) и оценку (mark), возвращает количество этих оценок, полученных на экзамене (используя таблицу FIO\_MARKS).

2. Написать программу, в которой создать процедуру, выводящую на экран фамилию преподавателя, название предмета, оценку, количество данных оценок, поставленных на экзамене (используя таблицы FIO\_SUBJECTS, FIO\_TEACHERS и функцию FIO\_QUANTITY\_MARK)

**Ілюстрації до роботи:**

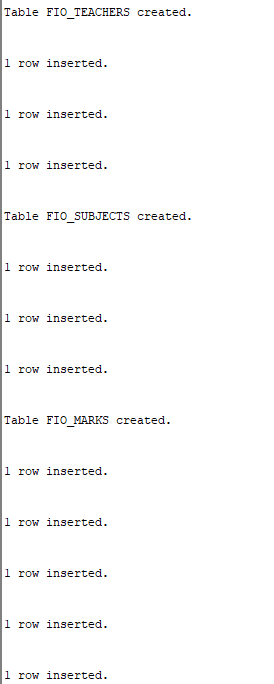


Рис 1. Список дій

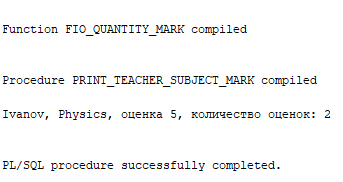


Рис 2. Список дій та вивід

***Код до завдання:***

DROP TABLE FIO\_MARKS;

DROP TABLE FIO\_TEACHERS;

DROP TABLE FIO\_SUBJECTS;

SET SERVEROUTPUT ON;

-- Создание таблицы FIO\_TEACHERS

CREATE TABLE FIO\_TEACHERS (

TEACHER\_ID NUMBER PRIMARY KEY,

SURNAME VARCHAR2(50) NOT NULL,

NAME VARCHAR2(50) NOT NULL,

PATRONYMIC VARCHAR2(50)

);

-- Заполнение таблицы FIO\_TEACHERS данными

INSERT INTO FIO\_TEACHERS (TEACHER\_ID, SURNAME, NAME, PATRONYMIC)

VALUES (1, 'Ivanov', 'Ivan', 'Ivanovich');

INSERT INTO FIO\_TEACHERS (TEACHER\_ID, SURNAME, NAME, PATRONYMIC)

VALUES (2, 'Petrov', 'Petr', 'Petrovich');

INSERT INTO FIO\_TEACHERS (TEACHER\_ID, SURNAME, NAME, PATRONYMIC)

VALUES (3, 'Sidorov', 'Sidor', 'Sidorovich');

-- Создание таблицы FIO\_SUBJECTS

CREATE TABLE FIO\_SUBJECTS (

SUBJECT\_ID NUMBER PRIMARY KEY,

SUBJECT\_NAME VARCHAR2(50) NOT NULL

);

-- Заполнение таблицы FIO\_SUBJECTS данными

INSERT INTO FIO\_SUBJECTS (SUBJECT\_ID, SUBJECT\_NAME)

VALUES (1, 'Math');

INSERT INTO FIO\_SUBJECTS (SUBJECT\_ID, SUBJECT\_NAME)

VALUES (2, 'Physics');

INSERT INTO FIO\_SUBJECTS (SUBJECT\_ID, SUBJECT\_NAME)

VALUES (3, 'Chemistry');

-- Создание таблицы FIO\_MARKS

CREATE TABLE FIO\_MARKS (

MARK\_ID NUMBER PRIMARY KEY,

STUDENT\_ID NUMBER NOT NULL,

TEACHER\_ID NUMBER NOT NULL,

SUBJECT\_ID NUMBER NOT NULL,

MARK NUMBER NOT NULL,

CONSTRAINT FK\_STUDENT FOREIGN KEY (STUDENT\_ID) REFERENCES FIO\_STUDENTS (STUDENT\_ID),

CONSTRAINT FK\_TEACHER FOREIGN KEY (TEACHER\_ID) REFERENCES FIO\_TEACHERS (TEACHER\_ID),

CONSTRAINT FK\_SUBJECT FOREIGN KEY (SUBJECT\_ID) REFERENCES FIO\_SUBJECTS (SUBJECT\_ID)

);

-- Заполнение таблицы FIO\_MARKS данными

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (1, 1, 1, 1, 4);

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (2, 2, 1, 1, 5);

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (3, 1, 2, 2, 3);

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (4, 2, 2, 2, 5);

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (5, 1, 3, 3, 4);

INSERT INTO FIO\_MARKS (MARK\_ID, STUDENT\_ID, TEACHER\_ID, SUBJECT\_ID, MARK)

VALUES (6, 2, 3, 3, 3);

-- Создаем хранимую функцию FIO\_QUANTITY\_MARK

CREATE OR REPLACE FUNCTION FIO\_QUANTITY\_MARK(sb\_id IN NUMBER, mark IN NUMBER)

RETURN NUMBER

IS

quantity\_mark NUMBER;

BEGIN

SELECT COUNT(\*) INTO quantity\_mark

FROM FIO\_MARKS

WHERE SUBJECT\_ID = sb\_id AND MARK = mark;

RETURN quantity\_mark;

END;

/

-- Создаем процедуру для вывода на экран нужных данных

CREATE OR REPLACE PROCEDURE PRINT\_TEACHER\_SUBJECT\_MARK(teach\_id IN NUMBER, sb\_id IN NUMBER, mark IN NUMBER)

IS

teach\_surname FIO\_TEACHERS.SURNAME%TYPE;

subject\_name FIO\_SUBJECTS.SUBJECT\_NAME%TYPE;

mark\_quantity NUMBER;

BEGIN

-- Получаем фамилию преподавателя

SELECT SURNAME INTO teach\_surname

FROM FIO\_TEACHERS

WHERE TEACHER\_ID = teach\_id;

-- Получаем название предмета

SELECT SUBJECT\_NAME INTO subject\_name

FROM FIO\_SUBJECTS

WHERE SUBJECT\_ID = sb\_id;

-- Получаем количество оценок по нужному предмету

mark\_quantity := FIO\_QUANTITY\_MARK(sb\_id, mark);

-- Выводим данные на экран

DBMS\_OUTPUT.PUT\_LINE(teach\_surname || ', ' || subject\_name || ', оценка ' || mark || ', количество оценок: ' || mark\_quantity);

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Данные не найдены');

END;

/

-- Вызываем процедуру с нужными параметрами

BEGIN

PRINT\_TEACHER\_SUBJECT\_MARK(1, 2, 5);

END;

/

/\*

В этом примере мы передаем в процедуру следующие параметры:

teach\_id = 1 - идентификатор преподавателя

sb\_id = 2 - идентификатор предмета

mark = 5 - оценка, которую мы ищем в таблице FIO\_MARKS

Процедура выведет на экран фамилию преподавателя, название предмета, оценку и количество оценок, соответствующих нашему запросу.

\*/