

МІНІСТЕРСТВО ОСВІТИ, НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ» ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Лабораторна робота №5

Інформаційно-комунікаційні системи: Бази даних

Варіант № 4

Підготував:

студент 3 курсу

групи ФІ-84

Коломієць Андрій Юрійович

E-mail: andkol-ipt22@lll.kpi.ua

ПІДЗАПИТИ: ПРОСТИ і КОРЕЛЬОВАНІ

Завдання

Предметна область:

Навчально-методичне управління (професорсько-викладацький склад).

Основні предметно-значущі сутності:

Співробітники, Підрозділи, Дисципліни.

Основні предметно-значущі атрибути сутностей:

- Співробітники прізвище, ім'я, по батькові, стать, дата народження, адреса прописки, посада, підрозділ;
- Підрозділи назва, вид підрозділу;
- Дисципліни назва.

Основні вимоги до функцій системи:

- Вибрати дисципліни, що читаються співробітниками або певним співробітником;
- Вибрати список співробітників по підрозділам або певного підрозділу;
- Вибрати дисципліни, що читаються співробітниками по підрозділам або певного підрозділу.

Тригери:

- 1. На додавання запису в таблицю «Працівники». Якщо в таблиці вже існує запис про співробітника з збігаються предметно-значущими атрибутами, заборонити додавання нового запису.
- 2. Створити представлення «Дисципліни» з полями «Код_сотрудніка», «ФІО_сотрудніка», «Дисципліна». Оновлювати представлення «Дисципліни».

Процедура:

Процедура повинна повертати кількість дисциплін, що читаються кожним співробітником зазначеного підрозділу.

Необхідно додатково

Проаналізувати предметну область вибраного варіанта завдання і сформувати запити з підзапитами:

- 1. Вивід списку атрибутів відношення, кількісні оцінки якого вище середнього, наприклад, імена студентів, у яких оцінка по тій чи іншій дисципліни вище середньої оцінки по цій самій дисципліни.
- 2. Вивід списку атрибутів відношення, кількісні оцінки якого максимальні або мінімальні, наприклад, імена студентів з максимальним сумарним балом.
- 3. Вивід списку атрибутів відношення, яке відповідає деякій умові з підзапиту, наприклад, вивести імена студентів відмінників і назви предметів, по яким вони здавали екзамен.

Виконання завдання

Microsoft SQL

```
USE Lab_5
DROP TABLE IF EXISTS MAIN, EMPLOYEE, DEGREE, DEPARTMENT, SUBJECTS;
CREATE TABLE DEGREE
 ID_DEGREE INT IDENTITY PRIMARY KEY NOT NULL,
 NAME_DEGREE VARCHAR (20) NOT NULL,
);
CREATE TABLE SUBJECTS
  ID_SUBJECT INT IDENTITY PRIMARY KEY NOT NULL,
  NAME_SUBJECT VARCHAR (20) NOT NULL,
);
CREATE TABLE DEPARTMENT
  ID_DEPARTMENT INT IDENTITY PRIMARY KEY NOT NULL,
  NAME_DEPARTMENT VARCHAR (20) NOT NULL,
);
```

```
CREATE TABLE EMPLOYEE
  ID_EMPLOYEE INT IDENTITY PRIMARY KEY NOT NULL,
  NAME_EMPLOYEE VARCHAR (20) NOT NULL,
  SURNAME_EMPLOYEE VARCHAR (20) NOT NULL,
  FATHER_NAME_EMPLOYEE VARCHAR (20) NOT NULL,
  AGE_EMPLOYEE INT,
  ADDRESS_EMPLOYEE VARCHAR (25),
);
CREATE TABLE MAIN
  EMPLOYEE_NAME INT,
  EMPLOYEE_DEGREE INT,
  EMPLOYEE_DEPARTMENT INT,
  EMPLOYEE_SUBJECT INT,
  FOREIGN KEY(EMPLOYEE_NAME) REFERENCES EMPLOYEE(ID_EMPLOYEE) ON DELETE CASCADE,
  FOREIGN KEY(EMPLOYEE_DEGREE) REFERENCES DEGREE(ID_DEGREE) ON DELETE CASCADE,
  FOREIGN KEY(EMPLOYEE_DEPARTMENT) REFERENCES DEPARTMENT(ID_DEPARTMENT) ON DELETE CASCADE,
  FOREIGN KEY(EMPLOYEE_SUBJECT) REFERENCES SUBJECTS(ID_SUBJECT) ON DELETE CASCADE
);
```

```
('DEGREE_1'),
 ('DEGREE_2'),
 ('DEGREE_3'),
 ('DEGREE_4'),
 ('DEGREE_5'),
 ('DEGREE_6'),
 ('DEGREE_7'),
 ('DEGREE_8'),
 ('DEGREE_9'),
 ('DEGREE_10');
INSERT INTO DEPARTMENT VALUES
 (' DEPARTMENT_1'),
 (' DEPARTMENT_2'),
 (' DEPARTMENT_3'),
 (' DEPARTMENT_4'),
 (' DEPARTMENT_5'),
 (' DEPARTMENT_6'),
 (' DEPARTMENT_7'),
 (' DEPARTMENT_8'),
 (' DEPARTMENT_9'),
 (' DEPARTMENT_10');
```

INSERT INTO DEGREE VALUES

```
INSERT INTO SUBJECTS VALUES
```

```
('SUBJECTS_1'),

('SUBJECTS_2'),

('SUBJECTS_3'),

('SUBJECTS_4'),

('SUBJECTS_5'),

('SUBJECTS_6'),

('SUBJECTS_7'),

('SUBJECTS_8'),

('SUBJECTS_9'),

('SUBJECTS_10');
```

INSERT INTO EMPLOYEE VALUES

```
('Name_1', 'Surname_1', 'Father_Name_1', '20', 'Street_1'),
('Name_2', 'Surname_2', 'Father_Name_2', '66', 'Street_2'),
('Name_3', 'Surname_3', 'Father_Name_3', '25', 'Street_3'),
('Name_4', 'Surname_4', 'Father_Name_4', '34', 'Street_4'),
('Name_5', 'Surname_5', 'Father_Name_5', '61', 'Street_5'),
('Name_6', 'Surname_6', 'Father_Name_6', '27', 'Street_6'),
('Name_7', 'Surname_7', 'Father_Name_7', '60', 'Street_7'),
('Name_8', 'Surname_8', 'Father_Name_8', '46', 'Street_8'),
('Name_9', 'Surname_9', 'Father_Name_9', '35', 'Street_9'),
('Name_10', 'Surname_10', 'Father_Name_10', '24', 'Street_10');
```

INSERT INTO MAIN VALUES

(1,2,3,1),

(2,3,2,5),

(3,2,1,2),

(4,3,5,1),

(5,2,3,2),

(6,3,3,10),

(7,7,3,1),

(8,8,3,3),

(9,2,1,6),

(10, 10, 2, 4);

-- Завдання:

---- Вивід списку атрибутів відношення, кількісні оцінки якого вище середнього, наприклад, імена студентів, у яких оцінка по тій чи іншій дисципліни вище середньої оцінки по цій самій дисципліни.

---- Вивід списку атрибутів відношення, кількісні оцінки якого максимальні або мінімальні, наприклад, імена студентів з максимальним сумарним балом.

---- Вивід списку атрибутів відношення, яке відповідає деякій умові з підзапиту, наприклад, вивести імена студентів відмінників і назви предметів, по яким вони здавали екзамен.

- -- Подібно до цього, можна провести аналогію:
- -- Виводимо працівників, вік котрих більший за середній вік всіх працівників та працюють на позиції "DEGREE_2"

SELECT EMPLOYEE.NAME_EMPLOYEE, EMPLOYEE.SURNAME_EMPLOYEE FROM EMPLOYEE, MAIN

	NAME_EMPLOYEE	SURNAME_EMPLOYEE
1	Name_7	Surname_7
2	Name_8	Surname_8

WHERE MAIN.EMPLOYEE_DEGREE>3 AND MAIN.EMPLOYEE_NAME=EMPLOYEE.ID_EMPLOYEE AND EMPLOYEE.AGE_EMPLOYEE > (SELECT AVG(AGE_EMPLOYEE) FROM EMPLOYEE)

-- Виводимо список працівників вік котрих є мінімальним

	NAME_EMPLOYEE	SURNAME_EMPLOYEE
1	Name_1	Surname_1

SELECT EMPLOYEE.NAME_EMPLOYEE,EMPLOYEE.SURNAME_EMPLOYEE FROM EMPLOYEE, MAIN

WHERE EMPLOYEE.AGE_EMPLOYEE = (SELECT MIN(EMPLOYEE.AGE_EMPLOYEE) FROM EMPLOYEE) AND EMPLOYEE.ID_EMPLOYEE=MAIN.EMPLOYEE_NAME

-- Виводимо працівників вік котрих є максимальним

	NAME_EMPLOYEE	SURNAME_EMPLOYEE
1	Name_2	Surname_2

SELECT EMPLOYEE.NAME_EMPLOYEE, EMPLOYEE.SURNAME_EMPLOYEE FROM EMPLOYEE, MAIN

WHERE EMPLOYEE.AGE_EMPLOYEE = (SELECT MAX(EMPLOYEE.AGE_EMPLOYEE) FROM EMPLOYEE) AND EMPLOYEE.ID_EMPLOYEE=MAIN.EMPLOYEE_NAME

-- Виводимо працівників та дисципліни, котрі вони викладають, вік яких менший за 30 або більший за 50

SELECT EMPLOYEE.NAME_EMPLOYEE, EMPLOYEE.SURNAME_EMPLOYEE FROM MAIN

JOIN EMPLOYEE ON EMPLOYEE.ID_EMPLOYEE=MAIN.EMPLOYEE_NAME

JOIN SUBJECTS ON SUBJECTS.ID_SUBJECT=MAIN.EMPLOYEE_SUBJECT

	NAME_EMPLOYEE	SURNAME_EMPLOYEE
1	Name_1	Surname_1
2	Name_2	Surname_2
3	Name_3	Surname_3
4	Name_5	Surname_5
5	Name_6	Surname_6
6	Name_7	Surname_7
7	Name_10	Surname_10

WHERE (EMPLOYEE.AGE_EMPLOYEE < 30 OR EMPLOYEE.AGE_EMPLOYEE>50) AND EMPLOYEE.ID_EMPLOYEE=MAIN.EMPLOYEE_NAME