



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

**Лабораторна робота №2**

**Інформаційно-комунікаційні системи:  
Бази даних**

**Варіант № 4**

**Підготував:**

студент 3 курсу

групи ФІ-84

Коломієць Андрій Юрійович

**E-mail:** [andkol-ipt22@lil.kpi.ua](mailto:andkol-ipt22@lil.kpi.ua)

## СТВОРЕННЯ ОБМЕЖЕНЬ БД: ФУНКЦІОНАЛЬНИХ, ЦІЛІСНИХ, ПОСИЛАЛЬНИХ

### 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**

1. Введіть обмеження на границі допустимих значень створеної вами бази даних (наприклад, Успішність – оцінка не повинна бути більшою за 12 балів. Номер семестра не повинен перевищувати 12).
2. Створіть зовнішні ключі у всіх таблицях, використовуючи опцію FOREIGN KEY, при цьому встановіть опцію каскадного видалення там, де це необхідно.
3. Відключіть обмеження зовнішнього ключа в таблиці . введіть в таблицю запис, значення поля якого порушує логічну цілісність таблиці (наприклад, у відношення Student, студента з неіснуючої групи). Спробуйте підключити раніше відключені обмеження.
4. Виконайте всі необхідні дії для того, щоб знову підключити обмеження, а всі дані у відношенні (наприклад, Student) відповідали умовам цілісності бази даних.
5. Змодельуйте ситуацію, коли необхідно відключити обмеження та розробіть заходи, які дозволять вам в подальшому привести базу даних в стан, що відповідає всім умовам цілісності.
6. Додати в одну з таблиць стовпець Single, тип даних VARCHAR(3), призначивши значення по замовчуванню «так». Видалити стовпець.
7. Перейменувати одну з таблиць.
8. Повернути попередню назву перейменованої таблиці.

## Query SQL Microsoft

```
USE Lab_2
```

```
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 NOT NULL,  
  
  ADDRESS_EMPLOYEE VARCHAR (25),  
)
```

```
CREATE TABLE MAIN
(
  EMPLOYEE_NAME INT,
  EMPLOYEE_DEGREE INT,
  EMPLOYEE_DEPARTMENT INT,
  EMPLOYEE_SUBJECT INT,
)
```

```
ALTER TABLE DEGREE ADD CONSTRAINT CHEAK_DEGREE CHECK (NAME_DEGREE!="");
```

```
ALTER TABLE SUBJECTS ADD CONSTRAINT CHEAK_SUBJECT CHECK (NAME_SUBJECT!="");
```

```
ALTER TABLE DEPARTMENT ADD CONSTRAINT CHEAK_DEPARTMENT CHECK (NAME_DEPARTMENT!="");
```

```
ALTER TABLE EMPLOYEE ADD CONSTRAINT CHEAK_EMPLOYEE_NAME CHECK (NAME_EMPLOYEE!="");
ALTER TABLE EMPLOYEE ADD CONSTRAINT CHEAK_EMPLOYEE_SURNAME CHECK (SURNAME_EMPLOYEE!="");
ALTER TABLE EMPLOYEE ADD CONSTRAINT CHEAK_EMPLOYEE_FATHER_NAME CHECK (FATHER_NAME_EMPLOYEE!="");
ALTER TABLE EMPLOYEE ADD CONSTRAINT CHEAK_EMPLOYEE_AGE CHECK (AGE_EMPLOYEE>=22 AND AGE_EMPLOYEE<=90);
ALTER TABLE EMPLOYEE ADD CONSTRAINT CHEAK_EMPLOYEE_ADRESS CHECK (FATHER_NAME_EMPLOYEE!="");
```

```
ALTER TABLE MAIN ADD CONSTRAINT FOREIGN_KEY_ID_EMPLOYEE FOREIGN KEY(EMPLOYEE_NAME) REFERENCES EMPLOYEE(ID_EMPLOYEE) ON DELETE CASCADE;
ALTER TABLE MAIN ADD CONSTRAINT FOREIGN_KEY_ID_DEGREE FOREIGN KEY(EMPLOYEE_DEGREE) REFERENCES DEGREE(ID_DEGREE) ON DELETE CASCADE;
ALTER TABLE MAIN ADD CONSTRAINT FOREIGN_KEY_ID_DEPARTMENT FOREIGN KEY(EMPLOYEE_DEPARTMENT) REFERENCES DEPARTMENT(ID_DEPARTMENT) ON DELETE CASCADE;
ALTER TABLE MAIN ADD CONSTRAINT FOREIGN_KEY_ID_SUBJECT FOREIGN KEY(EMPLOYEE_SUBJECT) REFERENCES SUBJECTS(ID_SUBJECT) ON DELETE CASCADE;
```

```
ALTER TABLE MAIN DROP CONSTRAINT FOREIGN_KEY_ID_EMPLOYEE;
ALTER TABLE MAIN DROP CONSTRAINT FOREIGN_KEY_ID_DEGREE;
ALTER TABLE MAIN DROP CONSTRAINT FOREIGN_KEY_ID_DEPARTMENT;
ALTER TABLE MAIN DROP CONSTRAINT FOREIGN_KEY_ID_SUBJECT;
```

```
ALTER TABLE MAIN ADD SINGLE VARCHAR(3) NOT NULL CONSTRAINT DEFINITION_SINGLE DEFAULT 'YES';
ALTER TABLE MAIN DROP CONSTRAINT DEFINITION_SINGLE;
ALTER TABLE MAIN DROP COLUMN SINGLE;
```

```
EXEC SP_RENAME 'MAIN','TOTAL COMMANDER';
EXEC SP_RENAME 'TOTAL COMMANDER','MAIN';
```