

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС  
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»  
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО  
АНАЛІЗУ

Лабораторна робота №2

З дисципліни «Організація баз даних»

**Виконав:**

студент 3-го курсу

групи КА-77

Щедров М. О.

**Прийняв:** Мухін В.Є.

Київ 2020р.

### Задание 2.1.

Создать хранимую функцию, подсчитывающую количество врачей в таблице Врачи и количество договоров в таблице Договора.

#### Результаты работы:

```
CREATE FUNCTION count_doctors () returns INT
] BEGIN
    DECLARE n INT;
    SELECT count(idDoctor) INTO n FROM Doctors;
    return n;
- END $$

delimiter ;

delimiter $$

CREATE FUNCTION count_agreements () returns INT
] BEGIN
    DECLARE n INT;
    SELECT count(idAgreement) INTO n FROM Agreements;
    return n;
- END $$

delimiter ;

select count_doctors(), count_agreements();
```

### Задание 2.2.

Написать процедуру добавления договора врачу – базовые таблицы Врачи, Договора

#### Результаты работы:

```

CREATE PROCEDURE setAgreement (idDoctorIn INT, valueAgreementIn INT, dateAgreementIn text, idAgreementIn INT)
/* параметры: номер задания, идентификатор сотрудника, номер проекта, название проекта. Возвращает 0, если зад
1 BEGIN
  DECLARE idA INT default 0; /*идентификатор проекта*/
  SELECT Doctors.idAgreement INTO idA FROM Doctors
    JOIN Agreements ON (Doctors.idAgreement=Agreements.idAgreement)
    WHERE idAgreementIn=Agreements.idAgreement and idDoctorIn = Doctors.idDoctor;

  IF (idA=0) /* если задание не назначено, то назначаем его*/
1 THEN BEGIN
  DECLARE fn text;
  DECLARE pos text;
  DECLARE cab INT;
  DECLARE bh text;
  DECLARE serv text;
  DECLARE sal text;
  SELECT Doctors.FullName INTO fn FROM Doctors
    WHERE idDoctorIn = idDoctor;
  SELECT Doctors.Position INTO pos FROM Doctors
    WHERE idDoctorIn = idDoctor;
  SELECT Doctors.Cabinet INTO cab FROM Doctors
    WHERE idDoctorIn = idDoctor;
  SELECT Doctors.BusinessHours INTO bh FROM Doctors
    WHERE idDoctorIn = idDoctor;
  SELECT Doctors.Service INTO serv FROM Doctors
    WHERE idDoctorIn = idDoctor;
  SELECT Doctors.Salary INTO sal FROM Doctors
    WHERE idDoctorIn = idDoctor;

  INSERT INTO Agreements VALUES (idAgreementIn, dateAgreementIn, valueAgreementIn);
  INSERT INTO Doctors VALUES (11, fn, pos, cab, bh, serv, sal, idAgreementIn);
  SELECT "Agreement inserted successfully";
  END;
  ELSE
    SELECT concat("This is already attached to the doctor ", idA);
  END IF;
  END$$

delimiter ;

call setAgreement (1, 1900, "2015-12-03", 10);

```

### Задание 1.3.

Написать процедуру, которая изменяет записи таблицы Врачи, таким образом, чтобы должности врачей становились написанными с большой буквы.

### Результаты работы:

delimiter \$\$

CREATE PROCEDURE BigName()

BEGIN

DECLARE id\_p INT; DECLARE name\_p TINYTEXT;

DECLARE is\_end INT DEFAULT 0; #флаг окончания работы

/\*объявляем курсор для запроса, который выбирает из таблицы идентификатор заг

DECLARE cur\_p\_name CURSOR FOR

SELECT idDoctor, concat(ucase(substring(FullName,1,1)),substring(FullName,2))

FROM Doctors;

# /\*объявляем обработчик для ситуации, когда курсор достигает конца результтирук

DECLARE CONTINUE HANDLER FOR NOT FOUND SET is\_end=1;

OPEN cur\_p\_name;

/\*в цикле читаем данные курсора и редактируем записи\*/

wet: LOOP

FETCH cur\_p\_name into id\_p, name\_p;

IF is\_end THEN LEAVE wet;

END IF;

UPDATE Doctors SET FullName=name\_p WHERE idDoctor=id\_p;

END LOOP wet;

CLOSE cur\_p\_name;

END\$\$

delimiter ;

call BigName();

select \* from Doctors;