

## ЛАБОРАТОРНА РОБОТА № 6

### Процедури, що зберігаються. Тригери

**Мета роботи:** дослідження властивостей та можливостей процедур, що зберігаються та тригерів

#### Хід роботи:

1. Для БД за індивідуальним завданням реалізувати:

1. Процедури, що зберігаються - не менше 3 (для запитів на пошук обов'язково).
2. Тригери для таблиць – не менше 3.

2. Побудувати до БД «Рейтинг» процедури, що зберігаються:

1. Розрахунок середнього балу студентів за період – без урахування перездач.
2. Розрахунок середнього балу студентів за період – з урахування перездач.
3. Визначення студентів, що навчаються на 4 та 5.
4. Процедура виводить суму балів та її значення в національній системі та ECTS.
5. Тригер на вставку даних в таблицю студент – якщо код групи новий в таблицю додається група.
6. Тригер на модифікацію даних з таблиці студенти якщо більше немає студентів в групі група знищується.

3. Завдання на самостійну роботу

1. Реалізувати тригери для обробки подій таблиці Zakza

Реалізувати функцію визначення знижки в залежності від кількості товару

#### Завдання 1.1

```
CREATE PROC proc1
```

```
AS
```

```
UPDATE Faculty SET pole1 = 'qwerty' + 'ASD'
```

```
WHERE Phone LIKE '(098)%'
```

```
EXEC proc1
```

```
SELECT * FROM Faculty
```

	idFaculty	FacultyName	DeanName	Phone	pole1
1	5	Факультет інформаційно-комп'ютерних технологій	Надія Миколаївна Лобанчикова	(098)1234567	qwertyASD
2	6	Факультет комп'ютерно-інтегрованих технологій, м...	Олексій Андрійович Громовий	(063)1234567	asd
3	7	Факультет бізнесу та сфери обслуговування	Галина Миколаївна Тарасюк	(098)1234567	qwertyASD
4	8	Природно-екологічний факультет	Володимир Володимирович Котенко	(063)1234567	asd
5	9	Факультет публічного управління та права	Димитрій Олександрович Грицишен	(063)1234567	asd

Рис.1.1

					ДУ «Житомирська політехніка». 19.121.07.000 – Лр6						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Дубнюк А. Р.			Звіт з лабораторної роботи			Лім.	Арк.	Аркушів	
Перевір.		Коротун О.В.								1	5
Керівник								ФІКТ Гр. ПІ-60[1]			
Н. контр.											
Зав. каф.											

```
CREATE PROC proc2
AS
SELECT * FROM Students
WHERE EnterDay < '2020-12-08' AND TypeOfTraining = 'Контракт'
EXEC proc2
```

	idStudent	idGroup	StudentName	Surname	Birthday	HomeAddress	ResidenceAddress	EnterDay	TypeOfTraining
1	5	5	Олександр	Хілоножко	1990-01-01	вул. Шевченка 1010 кв.1011	вул. Довженка 10 кв.2	2020-12-06	Контракт
2	6	2	Вася	Белій	1990-01-07	вул. Шевченка 1010 кв.1011	вул. Довженка 10 кв.2	2020-12-07	Контракт

Рис.1.2

```
CREATE PROC proc3
AS
DECLARE @ENTERDAY0 date, @TypeOfTraining0 varchar(50)
SET @ENTERDAY0 = '2020-12-08'
SET @TypeOfTraining0 = 'Контракт'
SELECT * FROM Students
WHERE EnterDay < @ENTERDAY0 AND TypeOfTraining = @TypeOfTraining0
EXEC proc3
```

	idStudent	idGroup	StudentName	Surname	Birthday	HomeAddress	ResidenceAddress	EnterDay	TypeOfTraining
1	5	5	Олександр	Хілоножко	1990-01-01	вул. Шевченка 1010 кв.1011	вул. Довженка 10 кв.2	2020-12-06	Контракт
2	6	2	Вася	Белій	1990-01-07	вул. Шевченка 1010 кв.1011	вул. Довженка 10 кв.2	2020-12-07	Контракт

Рис.1.3

```
CREATE PROC proc4
AS
SELECT Students.StudentName, Students.Surname, Faculty.FacultyName
FROM Students INNER JOIN Groups ON Students.idGroup = Groups.idGroup INNER JOIN Faculty ON
Groups.idFaculty = Faculty.idFaculty
WHERE Students.Surname = 'Білоножко'
EXEC proc4
```

	StudentName	Surname	FacultyName
1	Артем	Білоножко	Факультет інформаційно-комп'ютерних технологій

Рис.1.4

## Завдання1.2

```
--1--
CREATE TRIGGER trigger0
ON Faculty
AFTER INSERT
AS
INSERT INTO History(TempId, Operation)
SELECT idFaculty, 'Додавлен факультет: ' + FacultyName
FROM INSERTED
INSERT INTO Faculty
(FacultyName, DeanName, Phone) VALUES
('Факультет інформаційних технологій', 'Надія Миколаївна Лобанчикова', '(063)1234567')
SELECT * FROM History
--2--
CREATE TRIGGER trigger1
ON Faculty
AFTER DELETE
AS
INSERT INTO History(TempId, Operation)
SELECT idFaculty, 'Удален факультет' + FacultyName
FROM DELETED

DELETE FROM Faculty WHERE idFaculty=6
SELECT * FROM History
```

	Id	TempId	Operation	CreateAt
1	1	19	Добавлен факультет: Факультет інформаційних техно...	2020-05-15 16:46:15.953
2	2	6	Удален факультетФакультет комп'ютерно-інтегрован...	2020-05-15 16:48:33.710

Рис.1.5

```
CREATE TRIGGER trigger2
ON Faculty
AFTER UPDATE
AS
INSERT INTO History(TempId, Operation)
SELECT idFaculty, 'Факультет изменен: ' + FacultyName
FROM INSERTED
UPDATE Faculty
SET pole1 = 'qwertyф'
WHERE Phone LIKE '(098)%'
SELECT * FROM History
```

	Id	TempId	Operation	CreateAt
1	1	19	Добавлен факультет: Факультет інформаційних техно...	2020-05-15 16:46:15.953
2	2	6	Удален факультетФакультет комп'ютерно-інтегрован...	2020-05-15 16:48:33.710
3	3	7	Факультет изменен: Факультет бізнесу та сфери обс...	2020-05-15 16:52:18.353
4	4	5	Факультет изменен: Факультет інформаційно-комп'ю...	2020-05-15 16:52:18.353

Рис.1.6

## Завдання 2

```
--1--
CREATE PROC serBalBezPer @from DATE, @to DATE AS
SELECT Rozklad_pids.DATE, AVG(reiting.Reiting) AS avgReiting
FROM Rozklad_pids INNER JOIN reiting ON Rozklad_pids.K_zapis=reiting.K_zapis
WHERE Rozklad_pids.DATE BETWEEN @from AND @to
GROUP BY Rozklad_pids.DATE

EXECUTE serBalBezPer '2017-01-06', '2017-06-01'
```

Date	avgReiting
2017-01-06	63
2017-06-01	71

Рис.1.7. Процедура для розрахунку середнього балу студентів за період – без урахування перездач.

```
--3--
CREATE PROC Student4i5 AS
SELECT DISTINCT dbo_student.Name_ini
FROM reiting INNER JOIN dbo_student ON reiting.Kod_student=dbo_student.Kod_stud
WHERE reiting.Reiting BETWEEN 72 AND 100
ORDER BY dbo_student.Name_ini

EXECUTE Student4i5
```

	Name_ini		
1	Бобунов І.А.	14	Петрович О.М.
2	Головко В.В.	15	Прищепа В.А.
3	Горницький О.В.	16	Прокопович С.В.
4	Гриневич Д.М.	17	Родіонова Л.В.
5	Дерпак Ю.І.	18	Романчук Ю.В.
6	Клепус О.І.	19	Сарган О.В.
7	Кравченко К.С.	20	Сащук І.І.
8	Крамарчук О.В.	21	Сікорський Д.С.
9	Краснюк С.І.	22	Скрипник М.М.
10	Малыренко Е.О.	23	Сорочук О.Д.
11	Марченко О.Ю.	24	Халімончук А.О.
12	Мінчук Д.В.	25	Чистяков К.О.
13	Нижник В.М.	26	Шелуха О.О.

Рис.1.8. Процедура для визначення студентів, що навчаються на 4 та 5.

```
--4--
CREATE PROC mark AS
DECLARE @n INT
DECLARE @e varchar(2)
SELECT dbo_student.Name_ini, AVG(reiting.Reiting) AS AvgReiting,
CASE
    WHEN AVG(reiting.Reiting) BETWEEN 90 AND 100 then 5
    WHEN AVG(reiting.Reiting) BETWEEN 85 AND 89 then 4
    WHEN AVG(reiting.Reiting) BETWEEN 80 AND 84 then 3
    WHEN AVG(reiting.Reiting) BETWEEN 1 AND 49 then 2
END AS NationalMark,
CASE
    WHEN AVG(reiting.Reiting) BETWEEN 90 AND 100 then 'A'
    WHEN AVG(reiting.Reiting) BETWEEN 80 AND 89 then 'B'
    WHEN AVG(reiting.Reiting) BETWEEN 65 AND 79 then 'C'
    WHEN AVG(reiting.Reiting) BETWEEN 55 AND 64 then 'D'
    WHEN AVG(reiting.Reiting) BETWEEN 50 AND 54 then 'E'
    WHEN AVG(reiting.Reiting) BETWEEN 35 AND 49 then 'FX'
    WHEN AVG(reiting.Reiting) BETWEEN 1 AND 34 then 'F'
END AS ECTS
FROM reiting INNER JOIN dbo_student ON reiting.Kod_student = dbo_student.Kod_stud
GROUP BY dbo_student.Name_ini

EXECUTE mark
```

	Name_ini	AvgReiting	NationalMark	ECTS	19	Мелько В.М.	42	2	FX
1	Базелюк О.В.	43	2	FX	20	Мінчук Д.В.	70	4	C
2	Білявська В.С.	66	4	C	21	Нижник В.М.	71	4	C
3	Бобунов І.А.	73	4	C	22	Петрович О.М.	74	4	C
4	Герасимович В.Д.	54	3	E	23	Прищеп В.А.	74	4	C
5	Головко В.В.	71	4	C	24	Прокопович С.В.	73	4	C
6	Горницький О.В.	61	3	D	25	Радіонов А.Ю.	41	2	FX
7	Гриневич Д.М.	89	4	B	26	Радчук О.І.	37	2	FX
8	Дерпак Ю.І.	71	4	C	27	Родіонова Л.В.	66	4	C
9	Зосименко О.І.	59	3	D	28	Романчук Ю.В.	64	3	D
10	Клепус О.І.	81	4	B	29	Сарган О.В.	93	5	A
11	Кравченко К.С.	65	4	C	30	Сащук І.І.	70	4	C
12	Крамарчук О.В.	65	4	C	31	Сікорський Д.С.	77	4	C
13	Краснюк С.І.	67	4	C	32	Скрипник М.М.	70	4	C
14	Кузьменко О.О.	52	3	E	33	Сорочук О.Д.	68	4	C
15	Кухарець А.О.	44	2	FX	34	Халімончук А.О.	60	3	D
16	Лайчук О.В.	50	3	E	35	Чернецька І.С.	37	2	FX
17	Маляренко Є.О.	95	5	A	36	Чистяков К.О.	70	4	C
18	Марченко О.Ю.	68	4	C	37	Шелуха О.О.	69	4	C
					38	Шпаківський А.В.	42	2	FX

Рис.1.9. Процедура що виводить суму балів та її значення в національній системі та ECTS.

```
CREATE TRIGGER NewGroup ON dbo_student FOR INSERT AS
IF @@ROWCOUNT=1
BEGIN
    IF EXISTS(
        SELECT * FROM inserted
        WHERE inserted.Kod_group NOT IN (SELECT dbo_groups.Kod_group FROM dbo_groups))
    BEGIN
        INSERT into dbo_groups (Kod_group, Kod_men, Kod_zhurn, K_navch_plan, kilk)
        SELECT inserted.Kod_group, 2, 2, 17, 20
        FROM inserted
    END
END
```

```

CREATE TRIGGER onEmptyGroup ON dbo_groups FOR DELETE AS
IF @@ROWCOUNT=1
BEGIN
    IF EXISTS(
        SELECT * FROM dbo_groups WHERE dbo_groups.kilk = 0)
    BEGIN
        DELETE FROM dbo_groups WHERE dbo_groups.kilk = 0
    END
END
--2--
CREATE PROC SerBalVyzTeac @from Date, @to Date, @teach int AS
SELECT Rozklad_pids.Date, AVG(reiting.Reiting) AS avgReiting
FROM Rozklad_pids INNER JOIN reiting ON Rozklad_pids.K_zapis=reiting.K_zapis
WHERE Rozklad_pids.Date BETWEEN @from AND @to AND Rozklad_pids.k_vilkad = 2
GROUP BY Rozklad_pids.Date

EXECUTE SerBalVyzTeac '2017-01-06', '2017-06-01', 2

```

Date	avgReiting
2017-01-06	62
2017-06-01	71

Рис.1.10. Процедура розрахунку середнього балу студентів за період за визначеним викладачем.

### Завдання 3.1. тригер для обробки подій таблиці Zakaz, якщо дата замовлення пізніша за дату доставки.:

```

CREATE TRIGGER dateError ON zakaz FOR INSERT AS
IF @@ROWCOUNT=1
BEGIN
    IF EXISTS(
        SELECT * FROM inserted
        WHERE inserted.date_rozm > inserted.date_naznach)
    BEGIN
        ROLLBACK TRAN
    END
END

```

### Завдання 3.2. функція визначення знижки в залежності від кількості товару:

```

CREATE PROC Znigka @discount FLOAT OUTPUT, @amount int AS
IF @amount between 1 and 20 BEGIN SET @discount=0.1 END
IF @amount between 21 and 30 BEGIN SET @discount=0.2 END
IF @amount between 31 and 50 BEGIN SET @discount=0.3 END
IF @amount between 51 and 80 BEGIN SET @discount=0.4 END
IF @amount > 80 BEGIN SET @discount=0.5 END
DECLARE @disc FLOAT
DECLARE @kilkist int = 50
EXECUTE getDiscount @disc OUTPUT, @kilkist
PRINT @disc

```

**Висновок:** дослідженні властивості та можливості процедур, що зберігаються та тригерів