

**Приклад протоколу**  
Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Інститут прикладного системного аналізу  
Кафедра математичних методів системного аналізу

**ЗВІТ**  
про виконання комп'ютерного практикуму № 1  
з дисципліни «Системи баз даних»

Виконав:

Студент ІІІ курсу

Групи КА-65

Іванов Д.С,

Перевірів:

Караюз І. В.

Київ – 2018

**Завдання:** Відповідно до моєї теми курсового проекту, буду працювати із таблицями «ЗАМОВЛЕННЯ» та «ПАРТНЕРИ».

### **1. Словесний опис:**

Завдання 1. Створити таблицю «Orders»

Таблиця «Orders» має містити такі поля:

- цілочисельне поле Id з автоінкрементом;
- рядкове поле Name\_custom
- рядкове поле Promo\_kod

Завдання 1.2. Створити таблицю «Partners» Таблиця «Partners» має містити такі поля:

- цілочисельне поле Id з автоінкрементом;
- рядкове поле FirstName;
- цілочисельне поле ReferralAmount

Завдання 2. Створити тригер для таблиці «Partners», який буде спрацьовувати після додавання запису в таблицю «Orders».

**1. Лістинг:**

```
CREATE DATABASE Laba;
USE Laba;
```

```
CREATE TABLE `Orders` (
  `Id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `Id_custom` INT NOT NULL,
  `Name_custom` varchar(20) NOT NULL,
  `Promo_kod` varchar(10)
);
```

```
CREATE TABLE `Partners` (
  `Id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `Promo_kod` varchar(10) NOT NULL,
  `FirstName` varchar(20) NOT NULL,
  `ReferralAmount` int
);
```

```
INSERT INTO Partners
VALUES
('1', 'promo1', 'Noname1', '0'),
('2', 'promo2', 'Noname2', '0'),
('3', 'promo3', 'Noname3', '0'),
('4', 'promo4', 'Noname4', '0'),
('5', 'promo5', 'Noname5', '0');
```

```
DELIMITER //
CREATE PROCEDURE orders_insert (
  IN Id INT,
  IN Id_custom INT,
  IN Name_custom varchar(20),
  IN Promo_kod varchar(10))
BEGIN
  INSERT INTO Orders (Id,Id_custom,Name_custom,Promo_kod)
  VALUES (Id,Id_custom,Name_custom,Promo_kod);
END
```

```
CALL orders_insert('6','6','Kate', '0');
```

```
DELIMITER //
CREATE PROCEDURE orders_update (
  IN Promo_kod_new varchar(10))
BEGIN
  UPDATE Orders
  SET Promo_kod=Promo_kod_new
  WHERE Promo_kod='0';
END
```

```
CALL orders_update('without');
```

```

DROP PROCEDURE IF EXISTS orders_update;

DELIMITER //
CREATE PROCEDURE orders_delete (
IN Id_delete INT)
BEGIN
    DELETE FROM Orders
    WHERE Id=Id_delete;
END

CALL orders_delete('3');

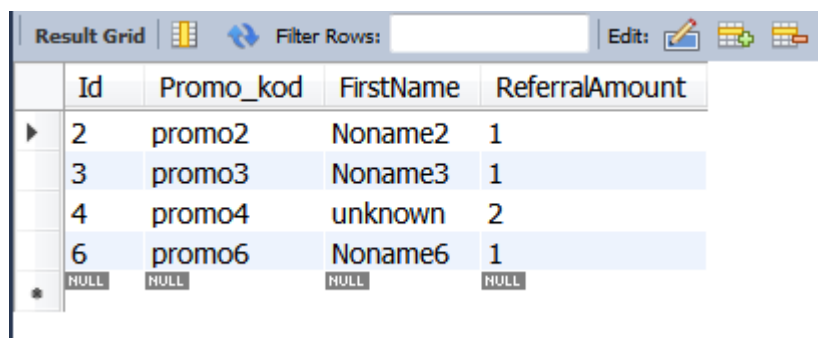
DELIMITER |
CREATE TRIGGER `after_insert_order`
AFTER INSERT ON `Orders`
FOR EACH ROW
    UPDATE Partners
    SET ReferralAmount = ReferralAmount + 1
    WHERE Promo_kod = NEW.Promo_kod;

DELIMITER $$
CREATE FUNCTION return_sum() RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE summa INT;
    SELECT SUM(ReferralAmount) INTO summa
    FROM Partners;
    RETURN summa;
END
DELIMITER $$

SELECT return_sum();

```

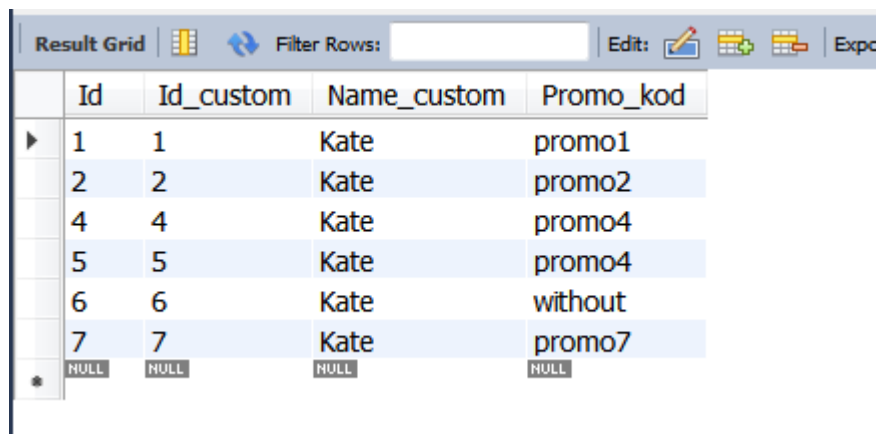
## 2. Приклад роботи:



The screenshot shows the MySQL Workbench interface with the 'Result Grid' tab active. It displays a table with four columns: Id, Promo\_kod, FirstName, and ReferralAmount. There are five rows of data, with the last row showing NULL values for all columns. The interface includes a 'Filter Rows' search bar and an 'Edit' button with a pencil icon.

	Id	Promo_kod	FirstName	ReferralAmount
▶	2	promo2	Noname2	1
	3	promo3	Noname3	1
	4	promo4	unknown	2
	6	promo6	Noname6	1
*	NULL	NULL	NULL	NULL

Рис. 3.1. Приклад роботи в MySQL Workbench (“Partners”)



	Id	Id_custom	Name_custom	Promo_kod
▶	1	1	Kate	promo1
	2	2	Kate	promo2
	4	4	Kate	promo4
	5	5	Kate	promo4
	6	6	Kate	without
	7	7	Kate	promo7
*	NULL	NULL	NULL	NULL

Рис. 3.2. Приклад роботи в MySQL Workbench (“Orders”)

### Висновки

Мною були отримані такі навички: створення таблиці, створення триггеру.