

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА  
ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота**  
з дисципліни  
«Організація баз даних та знань»

**Виконав:**  
студент групи КН-209

Гавришків Олексій

**Викладач:**  
Мельникова Н. І.

Львів – 2020 р.

## Лабораторна робота № 10

**Тема:** Написання збережених процедур на мові SQL.

**Мета:** Навчитися розробляти та виконувати збережені процедури та функції у MySQL.

### Хід роботи:

1. Створимо дві функції для кодування і декодування стрічок з типом VARCHAR(60).

```
CREATE FUNCTION password_encode (password VARCHAR(60))
RETURNS TINYBLOB
RETURN AES_ENCRYPT(password, 'sH2jrliIPS');

CREATE FUNCTION password_decode (encoded_phrase TINYBLOB)
RETURNS VARCHAR(60)
RETURN AES_DECRYPT(encoded_phrase, 'sH2jrliIPS')
```

2. Створимо користувача і закодуємо його пароль створеною функцією.

```
INSERT INTO `user` (`first_name`, `last_name`, `email`, `username`,
`password`) VALUES ('Oleksii', 'Havryshkiv', 'test@gmail.com',
'test', password_encode('secret_password'));
```

3. Тепер дістанемо запис і розкодуємо пароль

```
SELECT email, password_decode(password) as password FROM user;
```

#	email	password
1	test@gmail.com	secret_password

4. Створимо процедуру для заповнення тестових юзерів.

```
DELIMITER $$
CREATE PROCEDURE fillUsers()
BEGIN
    SET @i = 1;
    WHILE (@i < 20) DO
        INSERT INTO `user` (`first_name`, `last_name`, `email`,
`username`, `password`) VALUES (CONCAT('test',@i), CONCAT('test',@i),
CONCAT('test',@i,'@gmail.com'), CONCAT('test',@i),
password_encode('secret_password'));
        SET @i = @i + 1;
    END WHILE;
END$$
DELIMITER ;
```

5. Аналогічно створимо процедуру для додавання учасників до проєктів:

```
DELIMITER $$
CREATE PROCEDURE fillParticipants()
BEGIN
```

```

SET @i = 3;
WHILE (@i < 22) DO
    INSERT INTO `project_participant`
    (`project_id`, `user_id`, `role`) VALUES ((@i - 1)/2, @i, 0);
    SET @i = @i + 1;
END WHILE;
END$$
DELIMITER ;

```

## 6. Виконаємо процедури:

```

call docker_example.fillUsers();
call docker_example.fillParticipants();

```

## 7. Створимо і викличимо процедуру для визначення к-сті учасників в кожному проекті

```

DELIMITER $$
CREATE PROCEDURE getParticipantsCountByProject()
BEGIN
    SELECT count(user.id) AS participants_count, project.name AS
project
    FROM ((project_participant INNER JOIN user)
    INNER JOIN project
    ON project.id = project_participant.project_id
    AND user.id = project_participant.user_id)
    GROUP BY project.name
    ORDER BY participants_count DESC;
END$$
DELIMITER ;

```

```

call getParticipantsCountByProject();

```

#	participants_count	project
1	3	Project 2
2	2	Project 3
3	2	Project 4
4	2	Project 5
5	2	Project 6
6	2	Project 7
7	2	Project 8
8	2	Project 9
9	2	Project 10
10	1	Project 1

**Висновок:** виконуючи цю лабораторну роботу я навчився створювати процедури і функції в SQL і розглянув їх на прикладі системи управління проектами.