

## 1. Реализовать практические задания на примере других таблиц и запросов.

```
Command Prompt - mysql.exe -uroot -p

| Senior Staff |
| Assistant Engineer |
| Technique Leader |
| Manager |
+-----+
7 rows in set (9.97 sec)

mysql> SELECT DISTINCT title FROM employees.titles;
+-----+
| title |
+-----+
| Senior Engineer |
| Staff |
| Engineer |
| Senior Staff |
| Assistant Engineer |
| Technique Leader |
| Manager |
+-----+
7 rows in set (43.88 sec)

mysql> DELETE FROM employees WHERE emp_no = (SELECT emp_no FROM salaries ORDER BY salary DESC LIMIT 1)
-> ;
ERROR 1046 (3D000): No database selected
mysql> use employees
Database changed
mysql> DELETE FROM employees WHERE emp_no = (SELECT emp_no FROM salaries ORDER BY salary DESC LIMIT 1);
Query OK, 1 row affected (11.38 sec)

mysql>

Command Prompt - mysql.exe -uroot -p

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use employees
Database changed
mysql> lock tables employees read
-> ;
Query OK, 0 rows affected (0.00 sec)

mysql> lock tables employees write
-> ;
Query OK, 0 rows affected (0.00 sec)

mysql> unlock tables
-> ;
Query OK, 0 rows affected (0.00 sec)

mysql> lock tables employees write;
Query OK, 0 rows affected (0.00 sec)

mysql> unlock tables;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

2. Подумать, какие операции являются транзакционными, и написать несколько примеров с транзакционными запросами.

Повышаем сотрудника с id 10002 с Staff до Senior Staff следующей транзакцией:

```
BEGIN;

UPDATE `employees`.`titles`

SET

    `to_date` = CURDATE()

WHERE

    (`emp_no` = '10002')

    AND (`title` = 'Staff')

    AND (`from_date` = '1996-08-03');

INSERT INTO `employees`.`titles` VALUES('10002', 'Senior Staff', CURDATE(), '9999-01-01');

COMMIT;
```

Переводим работника из QA в DEV:

```
BEGIN;

UPDATE `employees`.`dept_emp`

SET

    `to_date` = CURDATE()

WHERE

    (`emp_no` = '10009')

    AND (`dept_no` = 'd006')

    AND (`from_date` = '1985-02-18');

INSERT INTO `employees`.`dept_emp` VALUES('10009', 'd005', CURDATE(), '9999-01-01');

COMMIT;
```

Проанализировать несколько запросов с помощью EXPLAIN.

Запрос из методички:

```
--
33 • EXPLAIN SELECT d.*, e.first_name, e.last_name
34 FROM departments d
35     LEFT JOIN dept_manager m
36         ON d.dept_no = m.dept_no
37     LEFT JOIN employees e
38         ON m.emp_no = e.emp_no;
39
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
	1	SIMPLE	d	NULL	index	NULL	dept_name	162	NULL	9	100.00	Using index
	1	SIMPLE	m	NULL	ref	dept_no	dept_no	16	employees.d.dept_no	2	100.00	Using index
	1	SIMPLE	e	NULL	eq_ref	PRIMARY	PRIMARY	4	employees.m.emp_no	1	100.00	NULL

Для оптимизации нужно добавить индексы для таблицы employees по полю emp\_no.

### Запрос из предыдущего ДЗ:

```
42 SELECT
43     CONCAT(e.last_name, ' ', e.first_name) AS 'Manager Name'
44 FROM
45     employees AS e
46 WHERE
47     e.emp_no = (SELECT
48                 d.emp_no
49                 FROM
50                     dept_manager AS d
51                 WHERE
52                     dept_no = (SELECT
53                               d.dept_no
54                               FROM
55                                   employees.employees AS e
56                               JOIN
57                                   dept_emp AS d ON e.emp_no = d.emp_no
58                               WHERE
59                                   e.first_name = 'Duangkaew' AND e.last_name = 'Piveteau'
60                                   AND d.to_date = '9999-01-01')
61                 AND d.to_date = '9999-01-01');
```

Result Grid												
Filter Rows: <input type="text"/> Export:  Wrap Cell Content:												
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
▶	1	PRIMARY	e		const	PRIMARY	PRIMARY	4	const	1	100.00	
	2	SUBQUERY	d		ref	dept_no	dept_no	16	const	4	10.00	Using where
	3	SUBQUERY	e		ALL	PRIMARY				299069	1.00	Using where
	3	SUBQUERY	d		ref	PRIMARY,em...	PRIMARY	4	employees.e.emp_no	1	10.00	Using where

Для следующего запроса, так-же видно, что не хватает индексов для таблицы employees по полю emp\_no.

А также:

- для dept\_emp и dept\_manager индексы для to\_date
- для employees индексы для first\_name и last\_name