



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский технологический университет»

МИРЭА

Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения
(ИППО)

**Отчет по лабораторной работе
по дисциплине
«Управление данными»**

**Тема: «Агрегирование данных с помощью групповых функций.
Подзапросы. Создание таблиц и управление ими»
Вариант 1**

Выполнил студент группы
ИСБОп-01-14

Карих Дмитрий Степанович

Принял
Старший преподаватель

Матчин Василий Тимофеевич

Москва 2017

Цель работы

Целью работы является получение навыков агрегирования данных с помощью групповых функций, создание подзапросов и создание таблиц и управление ими.

Теоретическое введение

Основным действием выполняемым с помощью SQL запросов является выборка данных для их вывода в виде таблицы в области результатов запроса.

- **SELECT** — Часть команды выборки, в которой задаются колонки выводимой таблицы. В ней через запятую перечисляются названия полей, из которых идет выборка.
- **FROM** — Часть строки выборки, в которой через запятую перечисляются запрашиваемые данным.
- **WHERE** — условия отбора, в соответствии с которым отбираются данные из таблицы.
- **AS** — Создает для столбца таблицы произвольное название. Ставится после названия вводимого поля, либо выражения.

Выполнение лабораторной работы

1. *Задание:* Напишите запрос для вывода должности (job_id) и количества сотрудников, занимающих каждую должность.

Решение:

```
SELECT `job`.`job_id`, COUNT(`employee`.`employee_id`)
FROM `job`, `employee`
FROM `job`.`job_id` = `employee`.`employee_id`;
```

1. *Задание:* Напишите запрос для вывода названия отдела, количества сотрудников и средней зарплаты по этому отделу. Назовите столбцы Department Name, Number of People и Salary.

Решение:

```
SELECT `department`.`name` AS "Department Name",
COUNT (`employee`.`employee_id`) AS "Number of People»,
AVG (`employee`.`salary`) AS "Salary"
FROM `department`, `employee`
WHERE `department`.`department_id` =
`employee`.`department_id`;
```

2. *Задание:* Напишите запрос для вывода фамилии и даты найма каждого сотрудника, работающего в одном отделе с сотрудником по фамилии Jones. Исключите Jones из выходных данных.

Решение:

```
SELECT `employee`.`last_name`, `employee`.`hire_date`
FROM `employee`, `department`, `employee` AS `jones`
WHERE
    `jones`.`last_name` = "Jones" AND
    `jones`.`department_id` = `employee`.`department_id`
AND
    `employee`.`last_name` <> "Jones";
```

3. *Задание:* Напишите запрос для получения кода отдела, фамилии и должности для каждого сотрудника, работающего в отделе исследований (Research).

Решение:

```
SELECT
    `department`.`department_id`,
    `employee`.`last_name`, `job`.`function`
FROM `department`, `employee`, `job`
WHERE `department`.`name` = "Research";
```

4. *Задание:* Создайте таблицу DEPT на основе экземпляра таблицы приведенного ниже.

Имя столбцов	ID	NAME	BIRTH	DESCRIPT	JOB
Тип ключа	PK	NO			
Nulls	NO		NO	YES	YES
Тип данных	INTEGER	VARCHAR	DATA	VARCHAR	VARCHAR
Длина	7	50		255	50

Решение:

```
CREATE TABLE `DEPT` (
    `ID` int(7) NOT NULL,
    `NAME` varchar(50) NOT NULL,
    `BIRTH` date NOT NULL,
    `DESCRIPT` varchar(255) DEFAULT NULL,
```

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
☐	1	ID	int(7)		Нет	Нет			Изменить Удалить Ещё
☐	2	NAME	varchar(50)	latin1_swedish_ci	Нет	Нет			Изменить Удалить Ещё
☐	3	BIRTH	date		Нет	Нет			Изменить Удалить Ещё
☐	4	DESCRIPT	varchar(255)	latin1_swedish_ci	Да	NULL			Изменить Удалить Ещё
☐	5	JOB	varchar(50)	latin1_swedish_ci	Да	NULL			Изменить Удалить Ещё

☐ Отметить все С отмеченными: ☐ Обзор ☐ Изменить ☐ Удалить ☐ Первичный ☐ Уникальный ☐ Индекс

Рис. 1 — Структура созданной таблицы по запросу

5. **Задание:** Создайте таблицу EMP на основе строк из таблицы EMPLOYEE. Необходимо включить только следующие столбцы EMPLOYEE_ID, FIRST_NAME, HIRE_DATE, SALARY. Столбцы должны называться ID, NAME, HDATE и SAL соответственно.

Решение:

```
CREATE TABLE `EMP`
SELECT
  `employee`.`employee_id` AS `ID`,
  `employee`.`first_name` AS `NAME`,
  `employee`.`hire_date` AS `HDATE`,
  `employee`.`salary` AS `SAL`
FROM `employee`;
```

6. **Задание:** Добавьте столбец LOC для таблицы DEPT, с типом данных VARCHAR и длиной 18. Созданный столбец должен быть уникальным. После этого удалите таблицы DEPT и EMP.

Решение:

```
ALTER TABLE `DEPT`
ADD COLUMN `LOC` VARCHAR(18) UNIQUE;
DROP TABLE `DEPT`;
DROP TABLE `EMP`;
```

Вывод по работе

В результате проведения лабораторной работы мы научились писать подзапросы и создавать таблицы, управлять ими. Было составлено четыре запроса, в которых были отработаны навыки агрегирования данных и создано две таблицы.