

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МОЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №10**  
**по дисциплине «Базы данных»**  
**Тема: Реализация на языке SQL основных операций**  
**реляционной алгебры**

Студент гр. 8383

\_\_\_\_\_

Киреев К.А.

Преподаватель

\_\_\_\_\_

Фомичева Т.Г.

Санкт-Петербург

2020

## **Цель работы.**

Приобретение навыков написания запросов на языке SQL на примерах реализации набора операций манипуляционной составляющей реляционной модели.

## **Основные теоретические положения.**

Реляционная модель определяет не только принципы хранения данных, но и способы их обработки.

Реляционная модель состоит из трех частей:

- структурной,
- целостной,
- манипуляционной.

В основу манипуляционной части модели положена теория множеств.

Отношение — это множество, поэтому средства манипулирования отношениями могут базироваться на традиционных теоретико-множественных операциях, дополненных некоторыми операциями, специфичными для баз данных.

Набор операций манипуляционной составляющей реляционной модели содержит восемь операций, которые делятся на два класса:

- **теоретико-множественные операции:** объединение, пересечение, разность, декартово произведение;
- **специальные реляционные операции:** селекция, проекция, соединение, деление.

В состав реляционной алгебры включаются также операции присваивания и переименования атрибутов.

Операции объединения, пересечения и разности имеют смысл не для любых двух множеств. Поскольку результат каждой из них - отношение (множество однотипных строк), то операндами должны быть отношения с совместимыми, схемами.

Отношения-операнды должны иметь одинаковую степень, одинаковые типы соответствующих атрибутов. Имена атрибутов могут отличаться, тогда после переименования можно выполнить основную операцию.

Для операции декартового произведения надо, чтобы заголовки столбцов, были разными

Все четыре операции: объединение, пересечение, разность и декартово произведение являются ассоциативными, т.е.  $(A \# B) \# C = A \# (B \# C) = A \# B \# C$ , где  $A$ ,  $B$  и  $C$  - отношения,  $\#$  - любая операция.

Все операции, кроме разности, коммутативны, т.е.  $A \# B = B \# A$ .

Восемь выделенных Кодом операций реляционной алгебры определяют эффективность реляционной СУБД.

### **Постановка задачи.**

1. Выбрать предметную область и придумать, опираясь на примеры теоретической части методических указаний, собственные примеры для реализации каждой операции. Для операций объединение, пересечение и разность использовать один общий пример. Сформулировать задачу для каждой из 8-ми операций в терминах выбранной предметной области.
2. Создать структуру (определение) каждой таблицы.
3. Заполнить таблицы тестовыми данными.
4. Написать SQL- код, реализующий каждую из 8-ми операций.
5. Выполнить запросы и убедиться, что они реализуют соответствующие операции.

### **Ход работы.**

1. Предметная область – гостиница, ее посетители и работники.
2. Было создано 5 таблиц: Гости в данный момент, Праздники, Особые гости за всю историю, Работники и Должность. Структура таблиц представлена на рис. 1-5.

Имя поля	Тип данных
Фамилия	Короткий текст
Возраст	Числовой
Номер	Числовой
Дата_заезда	Дата и время
Дата_выезда	Дата и время

Рисунок 1 – Структура таблицы “Гости в данный момент”

Имя поля	Тип данных
Название	Короткий текст
Дата	Дата и время

Рисунок 2 – Структура таблицы “Праздники”

Имя поля	Тип данных
Фамилия	Короткий текст
Возраст	Числовой
Номер	Числовой
Дата_заезда	Дата и время
Дата_выезда	Дата и время

Рисунок 3 – Структура таблицы “Особые гости за всю историю”

Имя поля	Тип данных
Фамилия	Короткий текст
Должность	Короткий текст
Возраст	Числовой

Рисунок 4 – Структура таблицы “Работники”

Должность_рабо
Повар
Администратор
Хостес

Рисунок 5 – Структура таблицы “Должность”

3. Далее, каждая из таблиц была заполнена тестовыми данными, позволяющими продемонстрировать все 8 операций, данные таблиц представлены на рис. 6-10.

Фамилия ▾	Возраст ▾	Номер ▾	Дата_заезда ▾	Дата_выезда ▾
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

Рисунок 6 – Данные таблицы “Гости в данный момент”

Название ▾	Дата ▾
Всемирный день хоккея	01.12.2020
День матери	29.11.2020
Всемирный день доброты	13.11.2020
День бабушек и дедушек	28.12.2020
Черная пятница	27.11.2020

Рисунок 7 – Данные таблицы “Праздники”

Фамилия ▾	Возраст ▾	Номер ▾	Дата_заезда ▾	Дата_выезда ▾
Емельяненко	41	4	08.11.2020	17.12.2020
Чайка	22	6	13.11.2020	06.12.2020
Басков	43	99	17.11.2020	23.11.2020
Клюев	56	78	08.11.2020	10.11.2020

Рисунок 8 – Данные таблицы “Особые гости за всю историю”

Фамилия ▾	Должность ▾	Возраст ▾
Зайцев	Повар	30
Туров	Администратор	36
Волкова	Хостес	23

Рисунок 9 – Данные таблицы “Работники”

Должность_работника ▾
Повар
Администратор
Хостес

Рисунок 10 – Данные таблицы “Должность”

#### 4. Реализация SQL-кода.

##### Объединение ( $R3 = R1 \cup R2$ )

- Постановка задачи

Требуется получить всех людей, проживающих в гостинице в данный момент и всех особых гостей. Тогда R1 – люди, проживающие в данный момент, а R2 – особые гости за всю историю.

- Исходные данные

R1:

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

R2:

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Емельяненко	41	4	08.11.2020	17.12.2020
Чайка	22	6	13.11.2020	06.12.2020
Басков	43	99	17.11.2020	23.11.2020
Клюев	56	78	08.11.2020	10.11.2020

- SQL-код операции

```
SELECT Гости_в_данный_момент.* FROM
Гости_в_данный_момент
UNION SELECT Особые_гости_за_всю_историю.* FROM
Особые_гости_за_всю_историю;
```

- Результат

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезда
Басков	43	99	17.11.2020	23.11.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Иванов	25	10	01.12.2020	25.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Клюев	56	78	08.11.2020	10.11.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

### Пересечение ( $R3 = R1 \cap R2$ )

- Постановка задачи

Требуется получить всех людей, проживающих в гостинице в данный момент и являющихся особыми гостями. Тогда  $R1$  – люди, проживающие в данный момент, а  $R2$  – особые гости за всю историю.

- Исходные данные

$R1$ :

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезда
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

$R2$ :

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезда
Емельяненко	41	4	08.11.2020	17.12.2020
Чайка	22	6	13.11.2020	06.12.2020
Басков	43	99	17.11.2020	23.11.2020
Клюев	56	78	08.11.2020	10.11.2020

- SQL-код операции

```
SELECT Гости_в_данный_момент.*
FROM Гости_в_данный_момент, Особые_гости_за_всю_историю
WHERE (Гости_в_данный_момент.Фамилия =
Особые_гости_за_всю_историю.Фамилия) AND
```

```
(Гости_в_данный_момент.Возраст =
Особые_гости_за_всю_историю.Возраст) AND
(Гости_в_данный_момент.Номер =
Особые_гости_за_всю_историю.Номер) AND
(Гости_в_данный_момент.Дата_заезда =
Особые_гости_за_всю_историю.Дата_заезда) AND
(Гости_в_данный_момент.Дата_выезда =
Особые_гости_за_всю_историю.Дата_выезда);
```

- Результат

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Емельяненко	41	4	08.11.2020	17.12.2020
Чайка	22	6	13.11.2020	06.12.2020

### Разность ( $R3 = R1 \setminus R2$ )

- Постановка задачи

Требуется получить всех людей, проживающих в гостинице в данный момент, которые не являются особыми гостями. Тогда R1 – люди, проживающие в данный момент, а R2 – особые гости за всю историю.

- Исходные данные

R1:

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

R2:

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Емельяненко	41	4	08.11.2020	17.12.2020
Чайка	22	6	13.11.2020	06.12.2020
Басков	43	99	17.11.2020	23.11.2020
Клюев	56	78	08.11.2020	10.11.2020

- SQL-код операции

```
SELECT Гости_в_данный_момент.*
FROM Гости_в_данный_момент LEFT JOIN
Особые_гости_за_всю_историю ON
```



```

(Гости_в_данный_момент.Дата_выезда =
Особые_гости_за_всю_историю.Дата_выезда) AND
(Гости_в_данный_момент.Дата_заезда =
Особые_гости_за_всю_историю.Дата_заезда) AND
(Гости_в_данный_момент.Номер =
Особые_гости_за_всю_историю.Номер) AND
(Гости_в_данный_момент.Возраст =
Особые_гости_за_всю_историю.Возраст) AND
(Гости_в_данный_момент.Фамилия =
Особые_гости_за_всю_историю.Фамилия)
WHERE (Особые_гости_за_всю_историю.Фамилия IS NULL) AND
(Особые_гости_за_всю_историю.Возраст IS NULL) AND
(Особые_гости_за_всю_историю.Номер IS NULL) AND
(Особые_гости_за_всю_историю.Дата_заезда IS NULL) AND
(Особые_гости_за_всю_историю.Дата_выезда IS NULL);

```

- Результат

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Степанова	33	5	02.12.2020	21.12.2020

## Декартово произведение ( $R3 = R1 \times R2$ )

- Постановка задачи

Требуется получить полный список обслуживающего персонала для каждого посетителя гостиницы. Обслуживающий персонал одинаковый для всех клиентов. Тогда  $R1$  – люди, проживающие в данный момент, а  $R2$  – работники.

- Исходные данные

$R1$ :

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

R2:

Фамилия	Должность	Возраст
Зайцев	Повар	30
Туров	Администратор	36
Волкова	Хостес	23

- SQL-код операции

```
SELECT Гости_в_данный_момент.*, Работники.*
FROM Гости_в_данный_момент, Работники;
```

- Результат

Гости_в_даг	Гости_в_даг	Номер	Дата_заезда	Дата_выезд	Работники.с	Должность	Работники.е
Иванов	25	10	01.12.2020	25.12.2020	Зайцев	Повар	30
Иванов	25	10	01.12.2020	25.12.2020	Туров	Администратор	36
Иванов	25	10	01.12.2020	25.12.2020	Волкова	Хостес	23
Петров	38	11	20.11.2020	15.12.2020	Зайцев	Повар	30
Петров	38	11	20.11.2020	15.12.2020	Туров	Администратор	36
Петров	38	11	20.11.2020	15.12.2020	Волкова	Хостес	23
Сидоров	19	14	10.10.2020	06.12.2020	Зайцев	Повар	30
Сидоров	19	14	10.10.2020	06.12.2020	Туров	Администратор	36
Сидоров	19	14	10.10.2020	06.12.2020	Волкова	Хостес	23
Киреев	27	17	29.11.2020	11.12.2020	Зайцев	Повар	30
Киреев	27	17	29.11.2020	11.12.2020	Туров	Администратор	36
Киреев	27	17	29.11.2020	11.12.2020	Волкова	Хостес	23
Кузьмин	45	21	30.11.2020	30.12.2020	Зайцев	Повар	30
Кузьмин	45	21	30.11.2020	30.12.2020	Туров	Администратор	36
Кузьмин	45	21	30.11.2020	30.12.2020	Волкова	Хостес	23
Емельяненко	41	4	08.11.2020	17.12.2020	Зайцев	Повар	30
Емельяненко	41	4	08.11.2020	17.12.2020	Туров	Администратор	36
Емельяненко	41	4	08.11.2020	17.12.2020	Волкова	Хостес	23
Степанова	33	5	02.12.2020	21.12.2020	Зайцев	Повар	30
Степанова	33	5	02.12.2020	21.12.2020	Туров	Администратор	36
Степанова	33	5	02.12.2020	21.12.2020	Волкова	Хостес	23
Чайка	22	6	13.11.2020	06.12.2020	Зайцев	Повар	30
Чайка	22	6	13.11.2020	06.12.2020	Туров	Администратор	36
Чайка	22	6	13.11.2020	06.12.2020	Волкова	Хостес	23

Деление ( $R3 = R1 \div R2$ )

- Постановка задачи

Модернизируем таблицу из прошлого пункта и удалим некоторых работников для нескольких посетителей и оставим только столбцы с фамилией гостя и должностью работника, который его обслуживает, полученную таблицу назовем “Предоставление сервиса гостиницы”. Также добавим гостям Емельяненко и Чайка новых работников, не входящих в список делителя: личный прислуга и горничная соответственно. Требуется получить список имен всех гостей, которых обслуживают все работники из списка должностей. Тогда R1

– “Предоставление сервиса гостиницы”, а R2 – должности работников.

- Исходные данные

R1:

Фамилия_гостя	Должность_работника
Емельяненко	Хостес
Емельяненко	Личный прислуга
Емельяненко	Повар
Иванов	Хостес
Иванов	Повар
Киреев	Администратор
Киреев	Хостес
Киреев	Повар
Кузьмин	Хостес
Кузьмин	Администратор
Кузьмин	Повар
Петров	Хостес
Петров	Администратор
Сидоров	Хостес
Сидоров	Повар
Степанова	Хостес
Чайка	Повар
Чайка	Администратор
Чайка	Горничная

R2:

Должность_рабо
Повар
Администратор
Хостес

- SQL-код операции

```
SELECT DISTINCT Фамилия_гостя
FROM Предоставление_сервиса_гостиницы AS X
WHERE NOT EXISTS
(SELECT Должность_работника FROM Должность
WHERE Должность_работника NOT IN
(SELECT Должность_работника
FROM Предоставление_сервиса_гостиницы AS Y
WHERE X.Фамилия_гостя = Y.Фамилия_гостя));
```

- Результат

Фамилия_гостя ▾
Киреев
Кузьмин

### Проекция ( $R3 = \pi_{i1, i2 \dots i3}(R1)$ )

- Постановка задачи

Требуется получить всех посетителей гостиницы в текущий момент и номера, в которых они проживают. Тогда R1 – таблица, полученная декартовым произведением выше, содержащая сведения (все атрибуты) о каждом госте. Применяемая операция над R1:  $\pi_{1,3}$ .

- Исходные данные

R1:

Гости_в_дан ▾	Гости_в_дан ▾	Номер ▾	Дата_заезд ▾	Дата_выезд ▾	Работники.С ▾	Должность ▾	Работники.Е ▾
Иванов	25	10	01.12.2020	25.12.2020	Зайцев	Повар	30
Иванов	25	10	01.12.2020	25.12.2020	Туров	Администратор	36
Иванов	25	10	01.12.2020	25.12.2020	Волкова	Хостес	23
Петров	38	11	20.11.2020	15.12.2020	Зайцев	Повар	30
Петров	38	11	20.11.2020	15.12.2020	Туров	Администратор	36
Петров	38	11	20.11.2020	15.12.2020	Волкова	Хостес	23
Сидоров	19	14	10.10.2020	06.12.2020	Зайцев	Повар	30
Сидоров	19	14	10.10.2020	06.12.2020	Туров	Администратор	36
Сидоров	19	14	10.10.2020	06.12.2020	Волкова	Хостес	23
Киреев	27	17	29.11.2020	11.12.2020	Зайцев	Повар	30
Киреев	27	17	29.11.2020	11.12.2020	Туров	Администратор	36
Киреев	27	17	29.11.2020	11.12.2020	Волкова	Хостес	23
Кузьмин	45	21	30.11.2020	30.12.2020	Зайцев	Повар	30
Кузьмин	45	21	30.11.2020	30.12.2020	Туров	Администратор	36
Кузьмин	45	21	30.11.2020	30.12.2020	Волкова	Хостес	23
Емельяненко	41	4	08.11.2020	17.12.2020	Зайцев	Повар	30
Емельяненко	41	4	08.11.2020	17.12.2020	Туров	Администратор	36
Емельяненко	41	4	08.11.2020	17.12.2020	Волкова	Хостес	23
Степанова	33	5	02.12.2020	21.12.2020	Зайцев	Повар	30
Степанова	33	5	02.12.2020	21.12.2020	Туров	Администратор	36
Степанова	33	5	02.12.2020	21.12.2020	Волкова	Хостес	23
Чайка	22	6	13.11.2020	06.12.2020	Зайцев	Повар	30
Чайка	22	6	13.11.2020	06.12.2020	Туров	Администратор	36
Чайка	22	6	13.11.2020	06.12.2020	Волкова	Хостес	23

- SQL-код операции

```
SELECT DISTINCT
Полный_сервис.Гости_в_данный_момент_Фамилия,
Полный_сервис.Номер
FROM Полный_сервис;
```

- Результат

Гости_в_дан	Номер
Емельяненко	4
Иванов	10
Киреев	17
Кузьмин	21
Петров	11
Сидоров	14
Степанова	5
Чайка	6

### Селекция (выбор) ( $R3 = \sigma_F(R1)$ )

- Постановка задачи

Требуется получить список всех гостей, старше 35 лет. Тогда R1 – люди, проживающие в данный момент. Применяемая операция над R1:  $\sigma_F$ , где  $\sigma_F$  – атрибут «Возраст» больше 35.

- Исходные данные

R1:

Фамилия	Возраст	Номер	Дата_заезда	Дата_выезд
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

- SQL-код операции

```
SELECT Гости_в_данный_момент.*
FROM Гости_в_данный_момент
WHERE Гости_в_данный_момент.Возраст > 35;
```

- Результат

Фамилия ▾	Возраст ▾	Номер ▾	Дата_заезда ▾	Дата_выезда ▾
Петров	38	11	20.11.2020	15.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020

### Эквисоединение ( $R3 = R1 \bowtie_{i=j} R2$ )

- Постановка задачи

Требуется получить список всех гостей, дата заезда которых совпадает с одним из праздников. Тогда R1 – люди, проживающие в данный момент. R2 – праздники. Условие соединения таблиц – равенство даты заезда посетителя и даты праздника. В результирующем отношении появляется два атрибута с одинаковыми значениями.

- Исходные данные

R1:

Фамилия ▾	Возраст ▾	Номер ▾	Дата ▾	Дата_выезда ▾
Иванов	25	10	01.12.2020	25.12.2020
Петров	38	11	20.11.2020	15.12.2020
Сидоров	19	14	10.10.2020	06.12.2020
Киреев	27	17	29.11.2020	11.12.2020
Кузьмин	45	21	30.11.2020	30.12.2020
Емельяненко	41	4	08.11.2020	17.12.2020
Степанова	33	5	02.12.2020	21.12.2020
Чайка	22	6	13.11.2020	06.12.2020

R2:

Название ▾	Дата ▾
Всемирный день хоккея	01.12.2020
День матери	29.11.2020
Всемирный день доброты	13.11.2020
День бабушек и дедушек	28.12.2020
Черная пятница	27.11.2020

- SQL-код операции

```
SELECT Гости_в_данный_момент.*, Праздники.*
FROM Праздники INNER JOIN Гости_в_данный_момент ON
Праздники.Дата = Гости_в_данный_момент.Дата;
```

- Результат

Фамилия	Возраст	Номер	Гости_в_данный_момент.Дата	Дата_выезд	Название	Праздники.Дата
Иванов	25	10	01.12.2020	25.12.2020	Всемирный день хоккея	01.12.2020
Киреев	27	17	29.11.2020	11.12.2020	День матери	29.11.2020
Чайка	22	6	13.11.2020	06.12.2020	Всемирный день доброты	13.11.2020

## Выводы.

В ходе выполнения данной работы были изучены основы составления SQL запросов и закреплены знания реляционной алгебры.