

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»

Факультет инфокоммуникационных технологий
09.03.03 «Мобильные и сетевые технологии»

Дисциплина
«ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ БАЗ ДАННЫХ»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4
«SQL запросы к базе данных»

Выполнил: Пузырев Д.А.
Студент группы К3241

Проверил: Говоров А.И.
Преподаватель

Санкт-Петербург
2020 г.

Цель работы

Овладеть практическими навыками создания SQL запросов к базе данных.

Описание варианта

Система должна обеспечивать хранение сведений об имеющихся в гостинице номерах, о проживающих в гостинице клиентах и о служащих, убирающихся в номерах. Количество номеров в гостинице известно, и имеются номера трех типов: одноместный, двухместный и трехместный, отличающиеся стоимостью проживания в сутки. В каждом номере есть телефон. О каждом проживающем должна храниться следующая информация: номер паспорта, фамилия, имя, отчество, город, из которого он прибыл, дата поселения в гостинице, выделенный гостиничный номер. О служащих гостиницы должна храниться информация следующего содержания: фамилия, имя, отчество, где (этаж) и когда (день недели) он убирает. Служащий гостиницы убирает все номера на одном этаже в определенные дни недели, при этом в разные дни он может убирать разные этажи.

Работа с системой предполагает получение следующей информации:

- о клиентах, проживавших в заданном номере, в заданный период времени;
- о количестве клиентов, прибывших из заданного города;
- кто из служащих убирал номер указанного клиента в заданный день недели;
- сколько в гостинице свободных номеров;
- список клиентов с указанием места жительства, которые проживали в те же дни, что и заданный клиент, в определенный период времени.

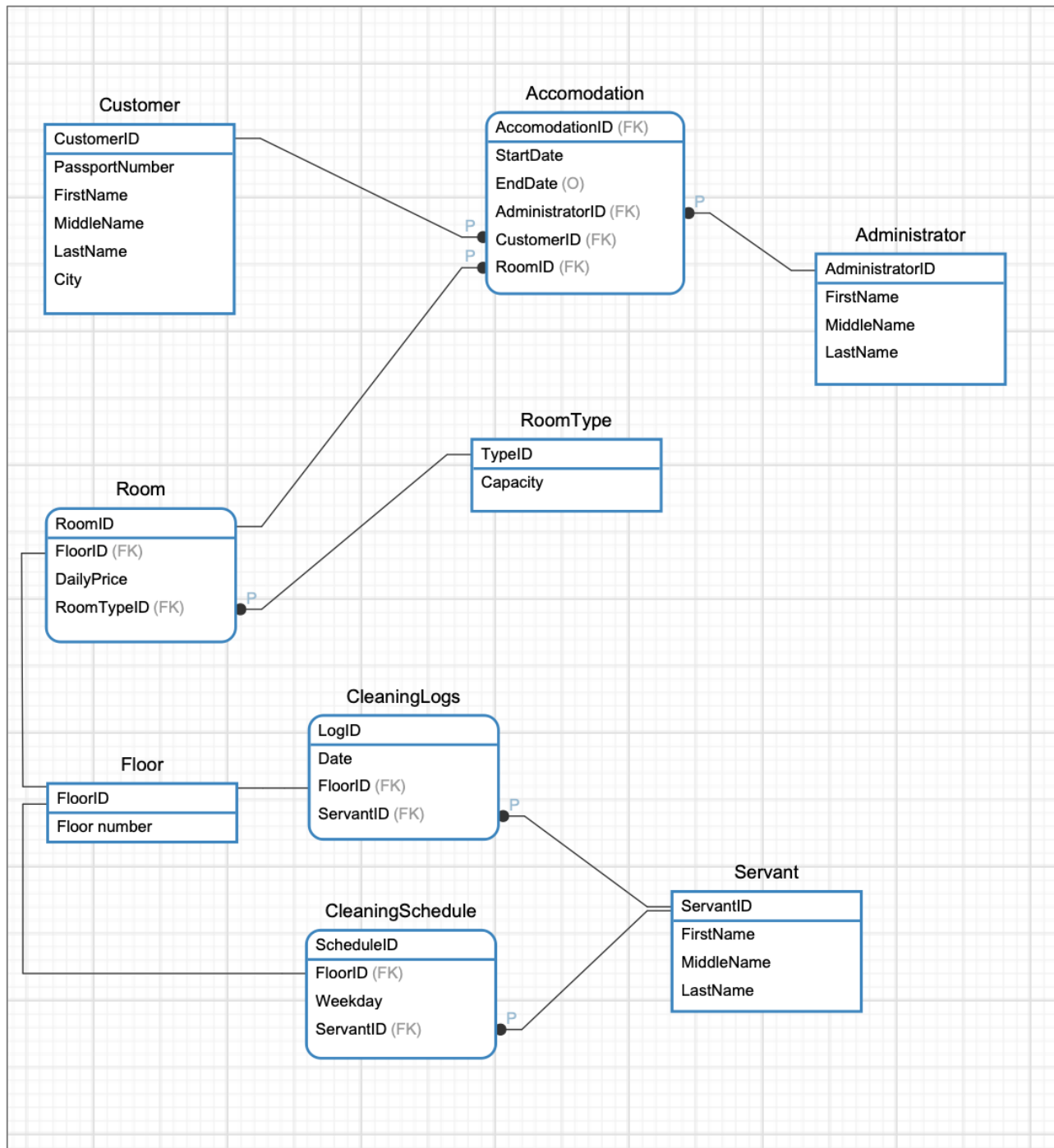
Администратор должен иметь возможность выполнить следующие операции:

- принять на работу или уволить служащего гостиницы;
- изменить расписание работы служащего;
- поселить или выселить клиента.

Необходимо предусмотреть также возможность автоматической выдачи отчета о работе гостиницы за указанный квартал текущего года. Такой отчет должен содержать следующие сведения:

- число клиентов за указанный период в каждом номере с указанием ФИО клиента, города, откуда он прибыл, количества дней проживания;
- общая сумма дохода за каждый номер;
- суммарный доход по всей гостинице.

Модель базы данных



Запросы к базе данных

1. Имена клиентов, проживавших в номере с RoomID = 1, начиная с 4 декабря 2019 года:

```
SELECT "Accommodation"."StartDate", "Accommodation"."EndDate",  
       "Customer"."FirstName"  
FROM "Accommodation"  
INNER JOIN "Customer"  
ON "Accommodation"."CustomerID" = "Customer"."CustomerID"  
WHERE "Accommodation"."StartDate" >= '2019-12-04'  
AND "Accommodation"."RoomID" = 1
```

	StartDate date	EndDate date	FirstName text
1	2020-04-01	2020-04-02	Назар
2	2019-12-05	2019-12-10	Александр

2. Количество клиентов, прибывших из Москвы:

```
SELECT COUNT(*) FROM "Customer"  
WHERE "Customer"."City" = 'Москва'
```

	count bigint
1	1

3. Кто из служащих убирал номер клиента с CustomerID = 1 в понедельник:

```
SELECT "Servant"."ServantID", "Servant"."FirstName"  
FROM "Accommodation"  
INNER JOIN "Room" ON "Accommodation"."RoomID" = "Room"."RoomID"  
INNER JOIN "CleaningSchedule" ON "Room"."FloorID" =  
"CleaningSchedule"."FloorID"  
INNER JOIN "Servant" ON "CleaningSchedule"."ServantID" =  
"Servant"."ServantID"  
WHERE "CustomerID" = 1  
AND "Weekday" = 1
```

	ServantID [PK] bigint	FirstName text
1	1	Анфиса

4. Количество свободных номеров в гостинице:

```
SELECT COUNT("RoomID") FROM "Room"
WHERE "RoomID" = ANY(
    SELECT "RoomID" FROM "Room"
    INTERSECT
    SELECT "RoomID" FROM "Accommodation"
    WHERE "EndDate" IS NULL
)
```

	count bigint	
1	2	

5. Клиенты с указанием их городов, которые жили в то же время, что и клиент с CustomerID = 1:

```
SELECT "Customer"."CustomerID", "Customer"."FirstName", "Customer"."City"
FROM "Accommodation"
INNER JOIN "Customer" ON "Accommodation"."CustomerID" =
"Customer"."CustomerID"
WHERE ("StartDate" BETWEEN (SELECT "StartDate" FROM "Accommodation" WHERE
"CustomerID" = 1) AND (SELECT "EndDate" FROM "Accommodation" WHERE
"CustomerID" = 1))
OR ("EndDate" BETWEEN (SELECT "StartDate" FROM "Accommodation" WHERE
"CustomerID" = 1) AND (SELECT "EndDate" FROM "Accommodation" WHERE
"CustomerID" = 1))
EXCEPT
SELECT "Customer"."CustomerID", "Customer"."FirstName", "Customer"."City"
FROM "Customer" WHERE "CustomerID" = 1
```

	CustomerID bigint	FirstName text	City text
1	2	Андрей	Санкт-Петербург

6. Число клиентов за период с декабря 2019 по февраль 2020 в каждом номере:

```
SELECT "RoomID", COUNT("CustomerID") FROM "Accommodation"
WHERE "StartDate" BETWEEN '2019-12-1' AND '2020-02-29'
GROUP BY "RoomID"
```

RoomID bigint	count bigint
1	1
2	1

7. Клиенты, имена которых начинаются на «А», в алфавитном порядке:

```
SELECT "CustomerID", "FirstName", "MiddleName", "LastName", "City"
FROM "Customer"
WHERE "FirstName" LIKE 'A%'
```

	CustomerID [PK] bigint	FirstName text	MiddleName text	LastName text	City text
1	2	Андрей	Михайлович	Пингвинин	Санкт-Петербург
2	4	Алишер	Тахирович	Мегилян	Черкесск
3	5	Анна	Михайловна	Полякова	Армавир
4	1	Александр	Васильевич	Прохоров	Москва

8. Дни недели, в которые производится уборка больше чем двумя сотрудниками:

```
SELECT "Weekday", COUNT("ServantID")
FROM "CleaningSchedule"
GROUP BY "Weekday"
HAVING COUNT("ServantID") > 2
```

	Weekday integer	count bigint
1	1	4

9. Весь обслуживающий персонал, включая как администраторов, так и уборщиков:

```
SELECT "FirstName", "MiddleName" FROM "Servant"
UNION
SELECT "FirstName", "MiddleName" FROM "Administrator"
```

	FirstName text	MiddleName text
1	Семен	Васильевич
2	Дмитрий	Андреевич
3	Дмитрий	Сергеевич
4	Ирина	Юрьевна
5	Василий	Михайлович
6	Андрей	Викторович
7	Андрей	Александрович
8	Михаил	Анатолевич
9	Светлана	Михайловна
10	Анфиса	Михайловна

10. Клиенты в порядке их выселения (от недавних к старым), при этом, если EndDate = NULL (еще не выселился), то будем сравнивать дату заселения:

```
SELECT "CustomerID", "StartDate", "EndDate"  
FROM "Accommodation"  
ORDER BY  
(CASE  
    WHEN "EndDate" IS NULL THEN "StartDate"  
    ELSE "EndDate"  
END) DESC
```

	CustomerID bigint	StartDate date	EndDate date
1	5	2020-04-20	[null]
2	3	2020-04-01	2020-04-02
3	4	2020-04-01	[null]
4	2	2019-12-07	2019-12-29
5	1	2019-12-05	2019-12-10

Вывод

В результате выполнения лабораторной работы были получены навыки выполнения SQL запросов.