ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

*Московский институт электроники и математики им. А. Н. Тихонова*

*Департамент компьютерной инженерии*

**Домашняя работа**

**Тема: «Отель»**

**Базы данных.**

Подготовил:

Евдокимов Иван, БИВ216

# Сущности предметной области

Типы атрибутов: **идентифицирующие**, *многозначные*, составные.

Постояльцы

Атрибуты: ФИО, **паспортные данные**, дата рождения, пол, *телефоны,* почта

Сотрудники

Атрибуты: ФИО, **паспортные данные**, дата рождения, **СНИЛС** пол, *телефоны*, *почты*, **ИНН,** *данные об образовании*

Номера

Атрибуты: **Номер комнаты**, количество мест, цена, тип номера, вид из номера

Посещения

Атрибуты: дата прибытия, время пребывания

Услуги

Атрибуты: **номер услуги**, **название**, стоимость

# Особенности предметной области

Каждый постоялец может посещать отель несколько раз, каждое посещение относится к одному клиенту.

Каждое посещение постояльца связано с одним номером, каждый номер может быть занят в разных посещениях (но не одновременно)

В каждом посещении клиент может приобретать несколько услуг, каждая услуга может быть приобретена несколько раз.

Каждая услуга может оказываться разными сотрудниками (или не быть связана с сотрудником), каждый сотрудник может оказывать разные услуги (в конкретном случае услуга оказывается одним сотрудником или без сотрудника)

# ER-диаграмма

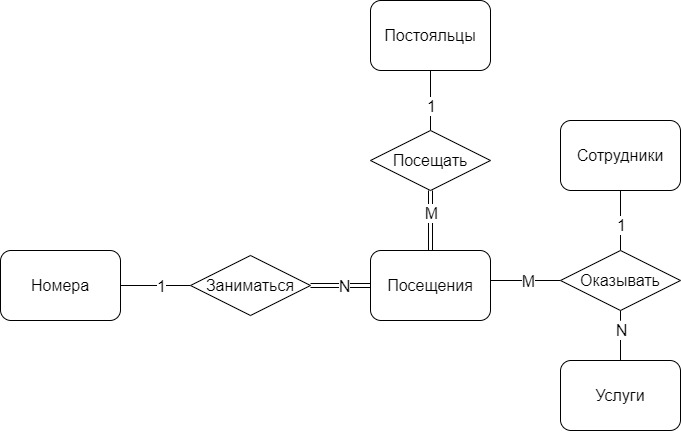


Рисунок 1 – ER-диаграмма

# Анализ информационных задач и круга пользователей системы

1. Руководитель:

* Добавление и изменение номеров.
* Добавление и изменение услуг.
* Просмотр всей бд.

1. Сотрудники отдела кадров:

* Прием новых сотрудников.
* Увольнение сотрудников.
* Внесение изменений в данные о сотрудниках.

1. Бухгалтеры:

* Подготовка ведомости для выплаты зарплаты сотрудникам.

1. Менеджеры:

* Добавление новых клиентов.
* Изменение данных о постояльцах.
* Редактирование данных об оказанных услугах.

1. Клиенты:

* Изменение своих данных.
* Просмотр списка доступных услуг.
* Просмотр списка доступных номеров.
* Просмотр своих посещений и услуг.

# Схема БД

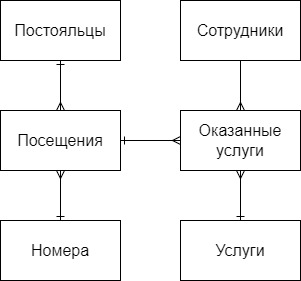


Рисунок 2 – Схема БД

# Исходная схема отношений

Таблица 1 – Схема отношения *ПОСТОЯЛЬЦЫ* (Guests)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор постояльца | GUEST\_ID | N(4) | Суррогатный первичный ключ |
| Фамилия | SURNAME | V(25) | обязательное поле |
| Имя, отчество | CNAME | V(45) | обязательное поле |
| Паспорт | PASSPORT | V(20) | обязательное поле, уникальное |
| Дата рождения | BORN | D | обязательное поле |
| пол | GENDER | C(1) | обязательное поле, in(‘м’, ’ж’) |
| телефоны | PHONE | V(30) | обязательное поле |
| Логин | LOGIN | V(50) |  |

Таблица – Схема отношения *СОТРУДНИКИ* (Employees)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор сотрудника | EMPLOYEE\_ID | N(4) | Суррогатный первичный ключ |
| Фамилия | SURNAME | V(25) | обязательное поле |
| Имя, отчество | ENAME | V(45) | обязательное поле |
| Паспорт | PASSPORT | V(20) | обязательное поле, уникальное |
| Дата рождения | BORN | D | обязательное поле |
| пол | GENDER | C(1) | обязательное поле |
| ИНН | INN | C(12) | обязательное уникальное поле |
| СНИЛС | SNILS | C(14) | обязательное уникальное поле |
| Должность | POST | V(30) | обязательное поле |
| Оклад | SAL | N(8,2) | Обязательное поле, 16000 |
| Образование | EDUCATION | V(30) | обязательное поле |
| телефоны | PHONE | V(30) | обязательное поле |
| почты | EMAILS | V(30) | обязательное поле |

Таблица – Схема отношения *УСЛУГИ* (Services)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Номер услуги | SNUMBER | N(6) | суррогатный первичный ключ |
| Название | SNAME | V(45) | обязательное уникальное поле |
| Стоимость | PRICE | N(8,2) | обязательное поле, , по умолчанию 0 |

Таблица – Схема отношения *ПОСЕЩЕНИЯ* (Visits)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания | |
| Идентификатор постояльца | GUEST\_ID | N(4) | Внешний ключ (к Guests) | Составной первичный ключ |
| Дата заселения | CHECK\_IN | D |  |
| Количество дней | STAY\_DAYS | integer | обязательное поле | |
| Номер | ROOM | N(4) | Внешний ключ (к Rooms) | |

Таблица – Схема отношения *ПРИОБРЕТЕННЫЕ УСЛУГИ* (AcqServices)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| ID постояльца | GUEST\_ID | N(4) | обязательное поле, составной внешний ключ (к Visits) |
| Дата заселения | CHECK\_IN | D |
| Услуга | SNUMBER | N(6) | обязательное поле, внешний ключ (к Services) |
| Дата приобретения | PURCHASE\_DATE | D | обязательное поле |

Таблица – Схема отношения *НОМЕРА* (Rooms)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Номер | NUMBER | C(6) | Перичный ключ |
| Тип номера | SNUMBER | N(6) | обязательное поле, внешний ключ (к Services) |
| Вид из номера | VIEW | C(10) | обязательное поле |
| Стоимость прживания | PRICE | N(6) | обязательное поле |

# Нормализация

Таблица – Схема отношения *ПОСТОЯЛЬЦЫ* (Guests)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор постояльца | GUEST\_ID | N(4) | Суррогатный первичный ключ |
| Фамилия | SURNAME | V(25) | обязательное поле |
| Имя, отчество | GNAME | V(45) | обязательное поле |
| Паспорт | PASSPORT | C(10) | обязательное поле, уникальное |
| Дата выдачи | P\_DATE | D | обязательное поле |
| Дата рождения | BORN | D | обязательное поле |
| Пол | GENDER | C(1) | обязательное поле, in(‘м’, ’ж’) |
| Логин | LOGIN | V(50) |  |

Таблица – Схема отношения *ТЕЛЕФОНЫ ПОСТОЯЛЬЦЕВ* (GuestsPhones)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор постояльца | GUEST\_ID | N(4) | Внешний ключ (к Guests) обязательное поле |
| Телефон | PHONE | V(20) | обязательное поле |

Таблица – Схема отношения *СОТРУДНИКИ* (Employees)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор сотрудника | EMPLOYEE\_ID | N(4) | Суррогатный первичный ключ |
| Фамилия | SURNAME | V(25) | обязательное поле |
| Имя, отчество | ENAME | V(45) | обязательное поле |
| Паспорт | PASSPORT | C(10) | обязательное поле, уникальное |
| Дата выдачи | P\_DATE | D | обязательное поле |
| Дата рождения | BORN | D | обязательное поле |
| Пол | GENDER | C(1) | обязательное поле, in(‘м’, ’ж’) |
| ИНН | INN | C(12) | обязательное уникальное поле |
| СНИЛС | SNILS | C(14) | обязательное уникальное поле |
| Должность | POST | V(30) | обязательное поле |

Таблица – Схема отношения *ТЕЛЕФОНЫ СОТРУДНИКОВ* (EmployeePhones)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор Сотрудника | EMPLOYEE\_ID | N(4) | Внешний ключ (к Employees) обязательное поле |
| Телефон | PHONE | V(20) | обязательное поле |

Таблица – Схема отношения *ПОЧТЫ СОТРУДНИКОВ* (EmployeeEmails)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор сотрудника | EMPLOYEE\_ID | N(4) | Внешний ключ (к Employees) обязательное поле |
| Почта | EMAIL | V(30) | обязательное поле |

Таблица – Схема отношения *ОБРАЗОВАНИЯ СОТРУДНИКОВ* (EmployeeEducations)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Идентификатор сотрудника | EMPLOYEE\_ID | N(4) | Внешний ключ (к Employees) обязательное поле |
| Специальность | SPECIALIZATION | V(30) | Внешний ключ (к Specialization) обязательное поле |
| Уровень образования | EDUCATION\_LEVEL | V(30) | Внешний ключ (к EducationLevels) обязательное поле |

Таблица – Схема отношения *ОБРАЗОВАНИЯ* (Specialization)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Специальность | SPECIALIZATION | V(30) | Первичный ключ |

Таблица – Схема отношения *УРОВНИ ОБРАЗОВАНИЯ* (EducationLevels)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Уровень образования | EDUCATION\_LEVEL | V(30) | Первичный ключ |

Таблица – Схема отношения *ДОЛЖНОСТИ* (Posts)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Название должности | POST | V(30) | Первичный ключ |
| Оклад | SAL | N(8,2) | Обязательное поле, 16000 |

Таблица – Схема отношения *УСЛУГИ* (Services)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Номер услуги | SNUMBER | N(6) | суррогатный первичный ключ |
| Название | SNAME | V(45) | обязательное уникальное поле |
| Стоимость | PRICE | N(8,2) | обязательное поле, , по умолчанию 0 |

Таблица – Схема отношения *ПОСЕЩЕНИЯ* (Visits)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания | |
| Идентификатор постояльца | GUEST\_ID | N(4) | Внешний ключ (к Guests) | Составной первичный ключ |
| Дата заселения | CHECK\_IN | D |  |
| Количество дней | STAY\_DAYS | integer | обязательное поле | |
| Номер | ROOM | C(6) | Обязательное поле, внешний ключ (к Rooms) | |

Таблица – Схема отношения *ПРИОБРЕТЕННЫЕ УСЛУГИ* (AcqServices)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| ID постояльца | GUEST\_ID | N(4) | обязательное поле, составной внешний ключ (к Visits) |
| Дата заселения | CHECK\_IN | D |
| Услуга | SNUMBER | N(6) | обязательное поле, внешний ключ (к Services) |
| Дата приобретения | PURCHASE\_DATE | D | обязательное поле |

Таблица – Схема отношения *НОМЕРА* (Rooms)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Номер | NUMBER | C(6) | Перичный ключ |
| Тип номера | TYPE | V(15) | обязательное поле, внешний ключ (к RoomTypes) |

Таблица – Схема отношения *ТИПЫ НОМЕРОВ* (RoomTypes)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Тип номера | TYPE | V(15) | Первичный ключ |
| Количество мест | CAPACITY | N(2) | обязательное поле, |
| Вид из номера | VIEW | C(10) | обязательное поле, внешний ключ (к Views) |
| Стоимость проживания | PRICE | N(6) | обязательное поле |

Таблица – Схема отношения *ВИДЫ ИЗ НОМЕРОВ* (Views)

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип, длина | Примечания |
| Вид из номера | VIEW | C(10) | Первичный ключ |

# Итоговая схема БД

Рисунок 3 – Схема БД

# Определение дополнительных ограничений целостности.

В одном номере не может быть больше человек, чем CAPACITY.

Дата приобретения услуги лежит между датой вселения и выселения постояльца, который её приобрёл.

Посещения одного постояльца не могут пересекаться по времени.

# Описание групп пользователей и прав доступа к ним.

Опишем для каждой группы пользователей права доступа к каждой таблице. Права доступа должны быть распределены так, чтобы для каждого объекта БД был хотя бы один пользователь, который имеет право добавлять и удалять данные из объекта. Используются следующие сокращения:

Таблица – Обозначения прав доступа

|  |  |
| --- | --- |
| Обозначение | Расшифровка |
| I (insert) | Добавление данных |
| S (select) | Чтение данных |
| U (update) | Модификация данных |
| D (delete) | Удаление данных |

Таблица – Права доступа к таблицам для групп пользователей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблицы | Группы пользователей | | | | |
| Руководитель орг. | Сотрудники отд. кадров | Бухгалтеры | Менеджеры | Клиенты |
| Постояльцы (Guests) | S |  |  | SUID | SU\* |
| Телефоны клиентов (GuestsPhones) | S |  |  | SUID | SUID\* |
| Сотрудники (Employees) | S | SUID | S | S |  |
| Телефоны сотрудников (EmployeePhones) | S | SUID |  | S |  |
| Почты сотрудников (EmployeeEmails) | S | SUID |  | S |  |
| Должности (Posts) | S | SUID | S | S |  |
| Образования сотрудников (EmployeeEducations) | S | SUID | S | S |  |
| Образования (Specialization) | S | SUID |  | S |  |
| Уровни образования (EducationLevels) | S | SUID |  | S |  |
| Услуги (Services) | SUID |  |  | S | S |
| Номера (Rooms) | SUID |  |  | S | S |
| Типы номеров (RoomTypes) | SUID |  |  | S | S |
| Виды из номеров (Views) | SUID |  |  | S | S |
| Приобретенные услуги (AcqServices) | S |  | S | SUID |  |
| Посещения (Visits) | S |  | S | SUID |  |

\* - только свои

# Создание отношений.

CREATE TABLE Guests (

GUEST\_ID NUMERIC(4) PRIMARY KEY,

SURNAME VARCHAR(25) NOT NULL,

GNAME VARCHAR(45) NOT NULL,

PASSPORT CHAR(10) NOT NULL UNIQUE,

P\_DATE DATE NOT NULL,

BORN DATE NOT NULL,

GENDER CHAR(1) NOT NULL CHECK (GENDER IN ('м', 'ж')),

LOGIN VARCHAR(50)

);

CREATE TABLE GuestsPhones (

GUEST\_ID NUMERIC(4) REFERENCES Guests (GUEST\_ID) NOT NULL,

PHONE VARCHAR(20) NOT NULL

);

CREATE TABLE Employees (

EMPLOYEE\_ID NUMERIC(4) PRIMARY KEY,

SURNAME VARCHAR(25) NOT NULL,

ENAME VARCHAR(45) NOT NULL,

PASSPORT CHAR(10) NOT NULL UNIQUE,

P\_DATE DATE NOT NULL,

BORN DATE NOT NULL,

GENDER CHAR(1) NOT NULL CHECK (GENDER IN ('м', 'ж')),

INN CHAR(12) NOT NULL UNIQUE,

SNILS CHAR(14) NOT NULL UNIQUE,

POST VARCHAR(30) NOT NULL

);

CREATE TABLE EmployeeEmails (

EMPLOYEE\_ID NUMERIC(4) REFERENCES Employees (EMPLOYEE\_ID) NOT NULL,

EMAIL VARCHAR(30) NOT NULL

);

CREATE TABLE EmployeePhones (

EMPLOYEE\_ID NUMERIC(4) REFERENCES Employees (EMPLOYEE\_ID) NOT NULL,

PHONE VARCHAR(20) NOT NULL

);

CREATE TABLE Specialization (

SPECIALIZATION VARCHAR(30) PRIMARY KEY

);

CREATE TABLE EducationLevels (

EDUCATION\_LEVEL VARCHAR(30) PRIMARY KEY

);

CREATE TABLE EmployeeEducations (

EMPLOYEE\_ID NUMERIC(4) REFERENCES Employees (EMPLOYEE\_ID) NOT NULL,

SPECIALIZATION VARCHAR(30) REFERENCES Specialization (SPECIALIZATION) NOT NULL,

EDUCATION\_LEVEL VARCHAR(30) REFERENCES EducationLevels (EDUCATION\_LEVEL) NOT NULL

);

CREATE TABLE Posts (

POST VARCHAR(30) PRIMARY KEY,

SAL NUMERIC(8, 2) NOT NULL CHECK (SAL >= 16000)

);

CREATE TABLE Services (

SNUMBER NUMERIC(6) PRIMARY KEY,

SNAME VARCHAR(45) NOT NULL UNIQUE,

PRICE NUMERIC(8, 2) NOT NULL DEFAULT 0 CHECK (PRICE >= 0)

);

CREATE TABLE Views (

VIEW CHAR(10) PRIMARY KEY

);

CREATE TABLE RoomTypes (

TYPE VARCHAR(15) PRIMARY KEY,

CAPACITY NUMERIC(2) NOT NULL CHECK (CAPACITY >= 1),

VIEW CHAR(10) REFERENCES Views (VIEW) NOT NULL,

PRICE NUMERIC(6) NOT NULL

);

CREATE TABLE Rooms (

NUMBER CHAR(6) PRIMARY KEY,

TYPE VARCHAR(15) REFERENCES RoomTypes (TYPE) NOT NULL

);

CREATE TABLE Visits (

GUEST\_ID NUMERIC(4) REFERENCES Guests (GUEST\_ID),

CHECK\_IN DATE NOT NULL,

STAY\_DAYS integer NOT NULL,

ROOM CHAR(6) REFERENCES Rooms (NUMBER) NOT NULL,

PRIMARY KEY (GUEST\_ID, CHECK\_IN)

);

CREATE TABLE AcqServices (

GUEST\_ID NUMERIC(4) NOT NULL,

CHECK\_IN DATE NOT NULL,

SNUMBER NUMERIC(6) NOT NULL REFERENCES Services (SNUMBER),

PURCHASE\_DATE DATE NOT NULL CHECK (PURCHASE\_DATE >= CHECK\_IN),

PRIMARY KEY (GUEST\_ID, CHECK\_IN)

);

# Заполнение справочных таблиц.

INSERT INTO Specialization (SPECIALIZATION) VALUES

(‘маркетинг’),

(‘менеджмент’),

(‘гостиничное дело’),

(‘экономика’),

(‘юриспруденция’);

INSERT INTO EducationLevel (EDUCATION\_LEVEL) VALUES

(‘основное общее’),

(‘среднее общее’),

(‘среднее профессиональное’),

(‘высшее бакалавриат’),

(‘высшее специалитет’),

(‘высшее магистратура’),

(‘высшее аспирантура’),

(‘высшее ординатура’);

INSERT INTO Posts (POST, SAL) VALUES

(‘менеджер’, 70000),

(‘главный менеджер’, 120000),

(‘консьерж’, 30000),

(‘горничная’, 35000),

(‘повар’, 50000),

(‘официант’, 40000),

(‘охранник’, 25000);

INSERT INTO Services (SNUMBER, SNAME, PRICE) VALUES

(1, ‘сейф’, 1000),

(2, ‘стирка’, 500),

(3, ‘массаж’, 1200),

(4, ‘завтрак в номер’, 400),

(5, ‘химчистка’, 600),

(6, ‘трансфер на рынок’, 700);

INSERT INTO Rooms (NUMBER, TYPE) VALUES

(‘A1’, ’стандарт-гд’),

(‘A2’, ’стандарт-гд’),

(‘A3’, ’стандарт-г’),

(‘A4’, ’стандарт-м’),

(‘A5’, ’стандарт-м’),

(‘B1’, ’ полулюкс-г’),

(‘B2’, ’ полулюкс-г’),

(‘B3’, ’ полулюкс-м’),

(‘B4’, ’ полулюкс-м’),

(‘C1’, ’ люкс-м’),

(‘C2’, ’ люкс-м’),

(‘C3’, ’ люкс-м’),

(‘C4’, ’ люкс-г’),

(‘C5’, ’ люкс-г’);

INSERT INTO RoomTypes (TYPE, CAPACITY, VIEW, PRICE) VALUES

(‘люкс-м’, 6, ‘море’, 30000),

(‘люкс-г’, 6, ‘горы, 29000),

(‘полулюкс-м’, 4, ‘море’, 28000),

(‘полулюкс-г’, 4, ‘горы, 27000),

(‘полулюкс-гд’, 4, ‘город’, 25000),

(‘стандарт-м’, 2, ‘море’, 18000),

(‘стандарт-г’, 2, ‘горы’, 17000),

(‘стандарт-гд’, 1, ‘город’, 15000);

INSERT INTO Views (VIEW) VALUES

(‘море’),

(‘горы),

(‘город),

(‘другое);

# Создание представлений.

1) Публичная информация о сотрудниках

Create or Replace View EmployeesInfo as

Select SURNAME, ENAME, GENDER, POST from Employees

Order by SURNAME;

2) Приобретенные услуги

Create or Replace View MyServives as

SELECT s.\*, Acs.PURCHASE\_DATE

FROM Services s

JOIN AcqServices Acs ON s.SNUMBER=Acs.SNUMBER

JOIN Guests g ON g.GUEST\_ID=Acs.GUEST\_ID

WHERE g.LOGIN = user

ORDER BY s.SNAME;

3) Информация о себе

Create or Replace View GuestInfo as

SELECT \* FROM Guests g

WHERE g.LOGIN=user

WITH CHECK OPTION;

4) Свои телефоны

Create or Replace View MyPhones as

SELECT \* FROM GuestsPhones gp

WHERE EXISTS (SELECT 1 FROM Guests g

WHERE g.GUEST\_ID=gp.GUEST\_ID

AND g.LOGIN=user)

WITH CHECK OPTION;

5) Свободные номера

Create or Replace View FreeRooms as

SELECT r.NUMBER, rt.\* FROM Rooms r

JOIN RoomTypes rt ON rt.TYPE=r.TYPE

WHERE NOT EXISTS (SELECT 1 FROM Visits v

WHERE v.ROOM=r.NUMBER

AND current\_date BETWEEN v.CHECK\_IN AND

v.CHECK\_IN+v.STAY\_DAYS)

ORDER BY r.NUMBER;

6) Счёт постояльцев (за номер и оказанные услуги)

CREATE OR REPLACE VIEW RoomPaymentInvoice AS

SELECT g.\*,

v.CHECK\_IN,

(

SELECT SUM(s.PRICE)

FROM Services s

JOIN AcqServices Acs ON Acs.SNUMBER = s.SNUMBER

WHERE Acs.GUEST\_ID = g.GUEST\_ID

AND Acs.CHECK\_IN = v.CHECK\_IN

) + rt.PRICE \* v.STAY\_DAYS AS invoice

FROM Guests g

JOIN Visits v ON v.GUEST\_ID = g.GUEST\_ID

JOIN Rooms r ON v.Room = r.NUMBER

JOIN RoomTypes rt ON rt.Type = r.Type

ORDER BY r.NUMBER;

Для работы с этими представлениями соответствующим пользователям нужно определить права доступа к представлениям

Таблица 14 – Права доступа к представлениям для групп пользователей

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Представления | Группы пользователей | | | | |
| Руководитель орг. | Сотрудник отд. кадров | Бухгалтер | Менеджер | Клиент |
| 1 | EmployeesInfo |  |  |  |  | S |
| 2 | MyServives |  |  |  |  | S |
| 3 | GuestInfo |  |  |  |  | SU |
| 4 | MyPhones |  |  |  |  | SUID |
| 5 | FreeRooms |  |  |  | S | S |
| 6 | room\_payment\_inmvoice | S |  | S | S |  |

# Назначение прав доступа.

CREATE ROLE director;

CREATE ROLE hr;

CREATE ROLE accountant;

CREATE ROLE manager;

CREATE ROLE guest;

GRANT SELECT ON Guests, GuestsPhones, Employees, EmployeePhones, EmployeeEmails, Posts, EmployeeEducations, Specialization, EducationLevels, AcqServices, Visits, RoomPaymentInvoice TO director;

GRANT SELECT, UPDATE, INSERT, DELETE ON Services, Rooms, RoomTypes, Views TO director;

GRANT SELECT, UPDATE, INSERT, DELETE ON Employees, EmployeePhones, EmployeeEmails, Posts, EmployeeEducations, Specialization, EducationLevels TO hr;

GRANT SELECT ON Employees, Posts, EmployeeEducations, AcqServices, Visits, RoomPaymentInvoice to accountant;

GRANT SELECT ON Employees, EmployeePhones, EmployeeEmails, Posts, EmployeeEducations, Specialization, EducationLevels, Services, Rooms, RoomTypes, Views, FreeRooms, RoomPaymentInvoice TO manager;

GRANT SELECT, UPDATE, INSERT, DELETE ON Guests, GuestsPhones, AcqServices, Visits TO manager;

GRANT SELECT ON Services, Rooms, RoomTypes, Views, EmployeesInfo, MyServives, FreeRooms TO guest;

GRANT SELECT, UPDATE ON GuestInfo TO guest;

GRANT SELECT, UPDATE, INSERT, DELETE ON MyPhones TO guest;

# Создание триггеров.

1. В одном номере не может быть больше человек, чем CAPACITY.

CREATE OR REPLACE FUNCTION check\_rooms()

RETURNS TRIGGER AS $$

DECLARE

Checker INTEGER;

BEGIN

SELECT COUNT(\*) INTO Checker FROM Rooms r

JOIN RoomTypes rt ON rt.TYPE=r.TYPE

WHERE EXISTS

(SELECT 1 FROM Visits v

WHERE v.ROOM=r.NUMBER

AND (SELECT COUNT(\*) FROM Visits vv

WHERE vv.ROOM=r.NUMBER

AND v.CHECK\_IN BETWEEN vv.CHECK\_IN AND

vv.CHECK\_IN+vv.STAY\_DAYS

OR vv.CHECK\_IN BETWEEN v.CHECK\_IN AND

v.CHECK\_IN+v.STAY\_DAYS)

> rt.CAPACITY);

IF Checker > 0 THEN

RAISE EXCEPTION 'Room is full';

END IF;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER check\_rooms\_trigger

AFTER INSERT OR UPDATE ON Visits

FOR EACH STATEMENT

EXECUTE FUNCTION check\_rooms();

1. Дата приобретения услуги до выселения постояльца, который её приобрёл.

CREATE OR REPLACE FUNCTION check\_service()

RETURNS TRIGGER AS $$

DECLARE

checker INTEGER;

BEGIN

IF (SELECT v.STAY\_DAYS+v.CHECK\_IN FROM Visits v

where v.GUEST\_ID=NEW.GUEST\_ID

AND v.CHECK\_IN=NEW.CHECK\_IN) < NEW.PURCHASE\_DATE THEN

RAISE EXCEPTION 'The service can’t be purchased after check-out.';

END IF;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER check\_service\_trigger

AFTER INSERT OR UPDATE ON AcqServices

FOR EACH ROW

EXECUTE FUNCTION check\_service();

1. Посещения одного постояльца не могут пересекаться по времени.

CREATE OR REPLACE FUNCTION check\_visits()

RETURNS TRIGGER AS $$

DECLARE

Checker INTEGER;

BEGIN

SELECT COUNT(\*) INTO Checker FROM Visits v

WHERE EXISTS (SELECT 1 FROM Visits vv

WHERE v.GUEST\_ID=vv.GUEST\_ID

AND v.CHECK\_IN BETWEEN vv.CHECK\_IN AND

vv.CHECK\_IN+vv.STAY\_DAYS

OR vv.CHECK\_IN BETWEEN v.CHECK\_IN AND

v.CHECK\_IN+v.STAY\_DAYS);

IF Checker > 0 THEN

RAISE EXCEPTION 'Guests can’t visit hotel multiple times at the same time.';

END IF;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER check\_visits\_trigger

AFTER INSERT OR UPDATE ON Visits

FOR EACH STATEMENT

EXECUTE FUNCTION check\_visits();

# Создание индексов.

Необходимо создать индексы для полей, по которым часто идёт поиск. В первую очередь, это внешние ключи.

CREATE INDEX idx\_GuestsPhones\_GUEST\_ID ON GuestsPhones (GUEST\_ID);

CREATE INDEX idx\_EmployeeEmails\_EMPLOYEE\_ID ON EmployeeEmails (EMPLOYEE\_ID);

CREATE INDEX idx\_EmployeePhones\_EMPLOYEE\_ID ON EmployeePhones (EMPLOYEE\_ID);

CREATE INDEX idx\_EmployeeEducations\_EMPLOYEE\_ID ON EmployeeEducations (EMPLOYEE\_ID);

CREATE INDEX idx\_EmployeeEducations\_SPECIALIZATION ON EmployeeEducations (SPECIALIZATION);

CREATE INDEX idx\_EmployeeEducations\_EDUCATION\_LEVEL ON EmployeeEducations (EDUCATION\_LEVEL);

CREATE INDEX idx\_RoomTypes\_VIEW ON RoomTypes (VIEW);

CREATE INDEX idx\_Visits\_ROOM ON Visits (ROOM);

CREATE INDEX idx\_AcqServices\_SNUMBER ON AcqServices (SNUMBER);

# Разработка стратегии резервного копирования.

Стратегия резервного копирования для базы данных гостиницы:

Интенсивность обновления разработанной базы данных невысокая (заселение постояльцев раз в день до 12 дня и запись услуг), для обеспечения сохранности достаточно проводить полное резервное копирование БД дважды в день (после заселения новых постояльцев днём и перед окончанием рабочего дня для фиксирования приобретённых услуг). Для разработанной БД нет необходимости держать сервер включенным круглосуточно, поэтому можно создать соответствующее задание операционной системы, которое будет автоматически запускаться перед выключением сервера.