Part 1: Выбор сценария

Для данной работы выбран сценарий: **Система бронирования в ресторане.** Эта система будет управлять бронированиями клиентов, столиками, персоналом и пунктами меню.

Part 2: Проектирование Базы Данных и Документация

**Идентификация Сущностей и Атрибутов:**

1. Клиенты (Customers);
2. Столы (Tables);
3. Бронирования (Reservations);
4. Гости (Guests);

**Проектирование Таблиц:**

1. **Table Name: Customers**

* **Description:** Содержит информацию о клиентах
* **Attributes:**
* *CustomerID*: INTEGER, PK, NOT NULL, UNIQUE
* *FirstName*: VARCHAR(50), NOT NULL
* *LastName*: VARCHAR(50), NOT NULL
* *Phone:* VARCHAR(16), NOT NULL
* **Constraints:**
* PK\_Customers: PRIMARY KEY (CustomerID)

1. **Table Name: Tables**

* **Description:** Содержит информацию о столиках
* **Attributes:**
* *TableID:* INTEGER, PK, NOT NULL, UNIQUE
* *IsActive: BOOLEAN*
* *Capacity: INTEGER, NOT NULL*
* **Constraints:**
* PK\_TABLES: PRIMARY KEY (TableID)
* DFLT\_ACTIVITY: DEFAULT (IsActive = FALSE)
* CHK\_CAPACITY: CHECK (Capacity > 0)

1. **Table Name: Reservations**

* **Description:** Содержит информацию обронированиях столиков
* **Attributes:**
* *ReservationID:* INTEGER, PK, NOT NULL, UNIQUE
* *CustomerID:* INTEGER, FK (REFERENSES Customers), NOT NULL
* *ReservationDate:* DATE, NOT NULL
* *ReservationTime:* TIME, NOT NULL
* *GuestsCount:* INTEGER, NOT NULL
* ***Constraints:***
* PK\_Reservation: PRIMARY KEY (ReservationID)
* FK\_Customer: FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
* CHK\_GuestCount: CHECK (GuestsCount > 0)

1. **Table Name: Guests**

* **Description:** Содержит информацию о гостях
* **Attributes:**
* *GuestID:* INTEGER, PK, NOT NULL, UNIQUE
* *GuestName:* VARCHAR(50), NOT NULL
* *ReservationID:* INTEGER, FK (REFERENCES Reservations), NOT NULL
* **Constraints:**
* PK\_Guest: PRIMARY KEY (GuestID)
* FK\_Reservation: FOREIGN KEY (ReservationID) REFERENCES Reservations(ReservationID)

1. **Table Name: Reservation\_Tables**

* **Description:** Является связующей таблицей между Reservations и Tables для осуществления связи Многие-ко-Многим
* **Attributes:**
* ReservationID: INTEGER, (REFERENCES Reservations), NOT NULL
* TableID: INTEGER, FK (REFERENCES Tables), NOT NULL
* **Constraints:**
* PK\_Reservation\_Table: PRIMARY KEY (ReservationID, TableID)
* FK\_Reservation: FOREIGN KEY (ReservationID) REFERENCES Reservations(ReservationID)
* FK\_Table: FOREIGN KEY (TableID) REFERENCES Tables(TableID)

**Взаимосвязи:**

* **Customers и Reservations (Один-ко-Многим):** Один клиент может забронировать несколько столиков, но один столик может быть забронирован лишь одним человеком.
* Reservations.CustomerID является внешним ключом, ссылающимся на Customers.CustomerID.
* **Reservations и Tables (Многие-ко-Многим):** Одно бронирование может включать несколько столов (например, для большой компании). Один стол может быть зарезервирован в разных бронированиях (в разное время).
* Реализовано с помощью связующей таблицы Reservation\_Tables с составным первичным ключем Reservation\_Tables.(ReservationID, TableID).
* **Reservations и Guests (Один-ко-Многим):** Одно бронирование может включать нескольких гостей, но один гость может сидеть лишь на забронированном для него столике.
* Guests.ReservationID, является внешним ключем, ссылающимся на Reservations.ReservationID.

Part 3: ER-Диаграмма

