# Система продажи билетов (начальное описание предметной области)

## Исходные данные

# Необходимо разработать структуру БД для системы учёта продажи билетов транспортного предприятия.

# Система должна позволять учитывать маршрутную сеть предприятия, расписание движения транспортных средств, продажу билетов, текущий парк транспортных средств. База данных должна обеспечивать сохранение следующей информации:

# • о транспортных средствах; для каждой единицы подвижного состава должны учитываться её модель, серийный номер, индивидуальный регистрационный знак (регистрационный номер транспортного средства), год выпуска, срок службы и максимальный пробег до списания, список посадочных мест салона;

# • о маршрутной сети компании: населённый пункт старта, населённый пункт финиша, общая протяжённость маршрута, время в пути;

# • список запланированных рейсов компании (расписание): маршрут поездки, транспортное средство, едущее по маршруту, дата и время отправления, ФИО водителя, проданные на рейс билеты; в билете помимо рейса указываются ФИО пассажира и место.

# Этапы проектирования

1. Анализ предметной области и входных данных
2. Концептуальная модель
3. Логическая модель
4. Физическая часть

# Что надо определить для концептуальной (инфологической) модели?

* Сущности
* Атрибуты
* Ключи
* Связи

## Выделение сущностей

**Маршрут, Расписание, Билет, Пассажир, Водитель, Транспортное средство, Тип ТС.**

## Выделение связей

**Маршрут** «может присутствовать в 1 и многих» **Расписаниях**

**Расписание** «может содержать только один конкретный» **Маршрут**

**Расписание** «обслуживает только один» **Водитель**

**Водитель** «может работать на 0, 1 и многих» **Расписаниях**

**Билет** «может быть куплен только на одно» **Расписание**

**Расписание** «покупается 0, 1 и многими» **Билетами**

**Билет** «может быть куплен только одним» **Пассажиром**

**Пассажиром** «может купить 0,1 и много» **Билетов**

**Расписание** «обслуживается только одним» **Транспортным средством**

**Транспортное средство** «может обслуживать 0, 1 и много» **Расписаний**

**Транспортное средство** «может быть только одного» **Типа ТС**

**Тип ТС** «могут иметь 1 и много» **Транспортных средств**

Диаграмма сущностей-связей

Водитель

Маршрут

Билет

Расписание

Транспортное средство

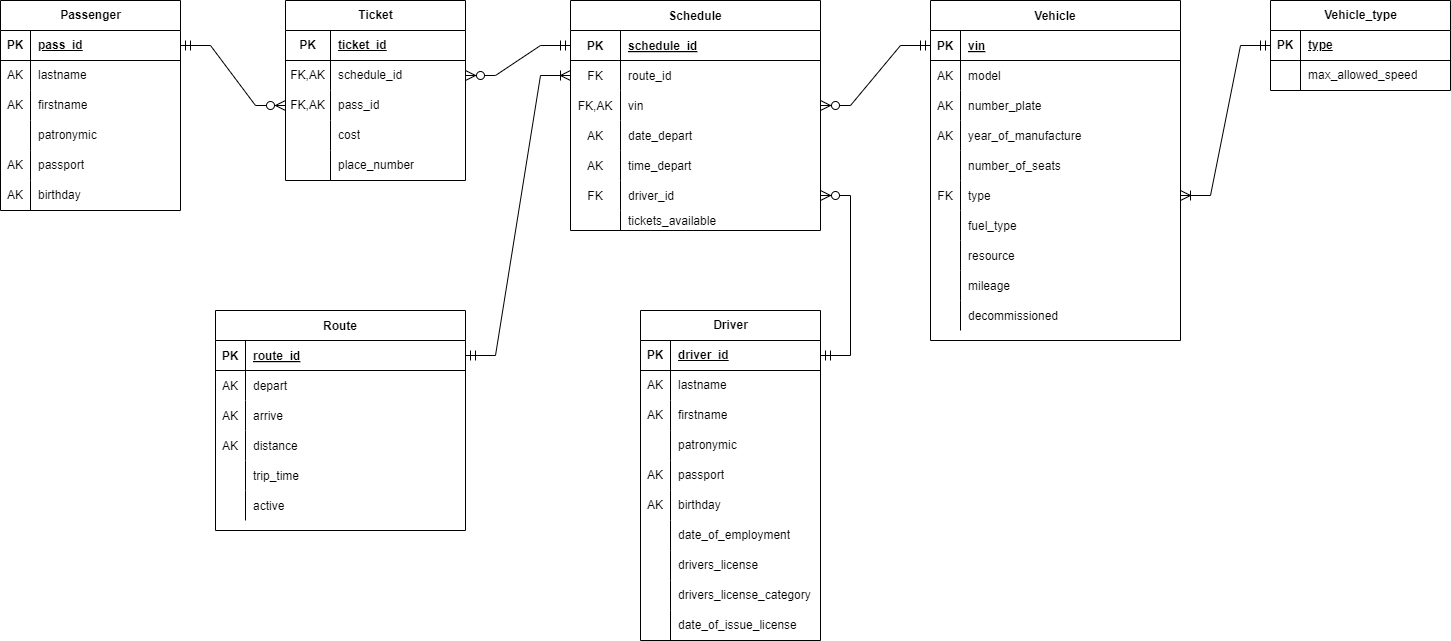
Пассажир

Тип ТС

## Выделение ключей, атрибутов, создание суррогатных ключей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Маршрут** | **Расписание** | **ТС** | **Тип ТС** | **Билет** | **Пассажир** | **Водитель** |
| 1 | route\_id | schedule\_id | vin | type | tickets\_id | pass\_id | driver\_id |
| 2 | Пункт отправления | Маршрут | Модель | Макс.разр. скорость | Рейс из расписания | Фамилия | Фамилия |
| 3 | Пункт назначения | ТС | Гос.Знак |  | Пассажир | Имя | Имя |
| 4 | Расстояние | Дата отправления | Год выпуска |  | Стоимость | Отчество | Отчество |
| 5 | Время в пути | Время отправления | Количество мест |  | Место | Номер паспорта | Номер паспорта |
| 6 | В работе | Водитель | Тип ТС |  |  | Дата рождения | Дата рождения |
| 7 |  | Доступно билетов | Вид топлива |  |  |  | Дата приема |
| 8 |  |  | Срок службы |  |  |  | Номер ВУ |
| 9 |  |  | Пробег |  |  |  | Категория ВУ |
| 10 |  |  | Списано |  |  |  | Дата выдачи ВУ |
|  |  |  |  |  |  |  |  |
|  |  | Первичный |  |  |  |  |  |
|  |  | Альтернативный |  |  |  |  |  |

# Создание ER-диаграммы



# Формирование логической модели

План действий

1. Каждая сущность превращается в таблицу. Имя сущности становится именем таблицы.
2. Каждый атрибут становится столбцом с тем же именем.
3. Компоненты уникального идентификатора сущности превращаются в первичный ключ таблицы.
4. Связи многие-к-одному (и один-к-одному) становятся внешними ключами.

## Описание сущностей-атрибутов

|  |  |  |  |
| --- | --- | --- | --- |
| **Название атрибута** | **Attribute** **name** | **Тип атрибута** | **Ограничения** |
| **Пассажир (Passenger)** | | | |
| Уникальный идентификатор пассажира | pass\_id | Число, целое положительное (integer) | **PK,**  generated always as identity (start with 1 increment by 1) primary key |
| Фамилия | lastname | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  **CHECK(**Lastname != ’’) |
| Имя | firstname | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  **CHECK(**Firstname != ’’) |
| Отчество | patronymic | Текст, до 30 символов (varchar(30)) | Не обязательный (**NULL**) |
| Номер паспорта | passport | Текст, точно 12 символов (char(12)) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону check(passport ~ '(?:^\d{2}\s\d{2}\s)(?:\d{6})') |
| Дата рождения | birthday | Дата (date) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону ‘ГГГГ-ММ-ДД’ |
| **Билет (Ticket)** | | | |
| Уникальный идентификатор билета | ticket\_id | Число, целое положительное (integer) | **PK,**  generated always as identity (start with 1 increment by 1) primary key |
| Рейс в расписании | schedule\_id | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  **АК**,  **FK** для связи с **Расписание**  on update cascade  on delete no action |
| Пассажир | pass\_id | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  **АК**,  **FK** для связи с **Пассажир**  on update cascade  on delete no action |
| Стоимость билета | cost | Число, дробное, с 2 знаками после запятой (money) | Обязательный (**NOT NULL**),  больше или равно 0,  check (cost >= 0::money)  default 0.00::money |
| Номер места | place\_number | Число, целое положительное (integer) | Не обязательный (**NULL**),  CHECK(place\_number >0 AND place\_number <= 100) |
| **Расписание (Schedule)** | | | |
| Уникальный идентификатор расписания | schedule\_id | Число, целое положительное (serial) | **PK,**  generated always as identity (start with 1 increment by 1) primary key |
| Маршрут | route\_id | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  **FK** для связи с **Маршрут**  on update cascade  on delete no action |
| Транспортное средство | vin | Текст, точно 17 символов (char(17)) | Обязательный (**NOT NULL**),  **АК**,  **FK** для связи с **ТС**  check(vin != ''),  on update cascade  on delete no action |
| Дата отправления | date\_depart | Дата (Date) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону ‘ГГГГ-ММ-ДД |
| Время отправления | time\_depart | Время (Time) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону ‘ЧЧ:ММ:СС’ |
| Водитель | driver\_id | Число, целое положительное (integer) | Не обязательный (**NULL**),  **FK** для связи с **Водитель**  on update cascade  on delete set null |
| Доступно билетов | tickets\_available | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  check(tickets\_avaliable >= 0 AND tickets\_avaliable <= 100) |
| **Транспортное средство (Vehicle)** | | | |
| Серийный номер | vin | Текст, точно 17 символов (char(17)) | **PK**  check(vin != '') |
| Модель | model | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК** |
| Гос.номерной знак | number\_plate | Текст, до 12 символов (varchar(12)) | Обязательный (**NOT NULL**),  **АК,**  Должен удовлетворять шаблону 'Х000ХХ[00,000]RUS или ХХ000[00,000]RUS’  check(number\_plate ~ '(?:^[АВЕКМНОРСТУХ]\d{3}(?<!000)[АВЕКМНОРСТУХ]{2}\d{2,3}RUS$)|(?:^[АВЕКМНОРСТУХ]{2}\d{3}(?<!000)\d{2,3}RUS$)') |
| Год выпуска | year\_of\_manufacture | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  **АК,**  Меньше или равно текущей дате:  CHECK(year\_of\_manufacture <= date\_part('year', now())::int4) |
| Количество мест | number\_of\_seats | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  CHECK(number\_of\_seats >= 0 AND number\_of\_seats <= 100) |
| Тип ТС | type | Текст, до 30 символов (varchar(30)) | Не обязательный (**NULL**),  **FK** для связи с **Тип ТС,**  default 'not set' ::text,  on update cascade  on delete set default |
| Вид топлива | fuel\_type | Текст, до 30 символов (varchar(30)) | Не обязательный (**NULL**) |
| Срок службы | resource | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  CHECK(resource >= 0 AND resource < 20) |
| Пробег | mileage | Число, целое положительное (integer) | Обязательный (**NOT NULL**) |
| Списано | decommissioned | (Boolean) | Не обязательный (**NULL**),  Должен удовлетворять шаблону ‘yes или no’,  default 'no'::boolean |
| **Тип ТС (Vehicle\_type)** | | | |
| Тип ТС | type | Текст, до 30 символов (varchar(30)) | **PK, check(type != '')** |
| Макс.разр. скорость для типа ТС | max\_allowed\_speed | Число, целое положительное (integer) | Не обязательный (**NULL**),  CHECK(max\_allowed\_speed >= 40 AND max\_allowed\_speed <= 130),  default 70 |
| **Маршрут (Route)** | | | |
| Уникальный идентификатор маршрута | route\_id | Число, целое положительное (serial) | **PK** |
| Пункт отправления | depart | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  **check(depart != '')** |
| Пункт назначения | arrive | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  check(arrive != '') |
| Расстояние | distance | Число, целое положительное (integer) | Обязательный (**NOT NULL**),  **АК,**  check(distance > 0) |
| Время в пути | trip\_time | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  Должен удовлетворять шаблону CHECK(trip\_time ~ ' (?:\d\d?\s(?:days?)\s)?\b[0-2]?[0-9]:[0-5][0-9]\b') |
| В работе | active | (Boolean) | Обязательный (**NOT NULL**),  Должен удовлетворять шаблону ‘yes или no’  default 'yes'::boolean |
| **Водитель (Driver)** | | | |
| Уникальный идентификатор водителя | driver\_id | Число, целое положительное (serial) | **PK** |
| Фамилия | lastname | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  **CHECK(**lastname != ’’) |
| Имя | firstname | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  **АК,**  **CHECK(**firstname != ’’) |
| Отчество | patronymic | Текст, до 30 символов (varchar(30)) | Не обязательный (**NULL**) |
| Номер паспорта | passport | Текст, точно 12 символов (char(12)) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону CHECK(passport ~ ' (?:^\d{2}\s\d{2}\s)(?:\d{6})') |
| Дата рождения | birthday | Дата (date) | Обязательный (**NOT NULL**),  **АК**,  Должен удовлетворять шаблону ‘ГГГГ-ММ-ДД’ |
| Дата приема на работу | date\_of\_employment | Дата (date) | Обязательный (**NOT NULL**), Должен удовлетворять шаблону ‘ГГГГ-ММ-ДД’ |
| Номер ВУ | drivers\_license | Текст, точно 12 символов (char(12)) | Обязательный (**NOT NULL**), Должен удовлетворять шаблону CHECK(drivers\_license ~ ' (?:^\d{2}\s\d{2}\s)(?:\d{6})') |
| Категория ВУ | drivers\_license\_category | Текст, до 30 символов (varchar(30)) | Обязательный (**NOT NULL**),  Категории записываются через запятую  check(drivers\_license\_category ~ '^(?:A|B|C|D|E|M|BE|CE|DE|C1|C1E|D1|D1E|Tm|Tb|A1|B1)(?:\s\*,\s\*(?:A|B|C|D|E|M|BE|CE|DE|C1|C1E|D1|D1E|Tm|Tb|A1|B1))\*$') |
| Дата выдачи ВУ | date\_of\_issue\_license | Дата (date) | Обязательный (**NOT NULL**), Должен удовлетворять шаблону ‘ГГГГ-ММ-ДД’ |