

2. Базы данных - ER

В базе данных есть таблица заказов - orders. В ней есть поля: id (идентификатор заказа), name (название заказа), town (адрес доставки заказа), price (цена заказа), customer_id (идентификатор покупателя). Также есть таблицы: towns (справочник адресов), items (товары), customers (покупатели). Известно, что между orders и items предполагается связь многие-ко-многим.

Что нужно сделать:

Спроектируйте ER-диаграмму с учетом этих вводных. Состав полей таблиц укажите на свое усмотрение с учетом условия, но для таблицы orders учтите те поля, что указаны.

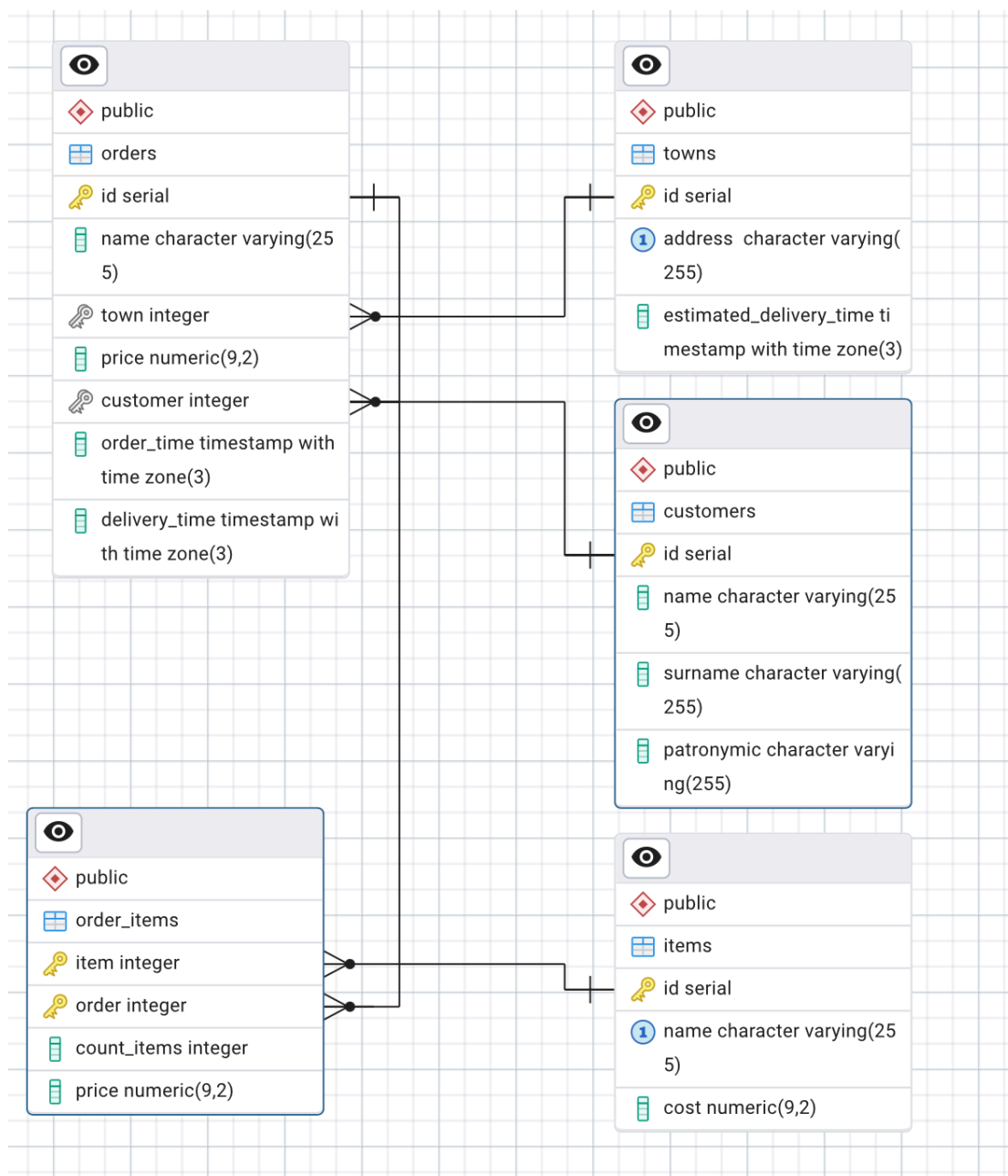


Диаграмма выполнена в нотации crow's foot

Всего было реализовано 5 сущностей:

1) **Orders** - таблица заказов

Имеет следующие поля:

Name	Data type	Constraints
id	Serial	Primary key
Name	String	NOT NULL
Town	integer	NOT NULL FOREIGN KEY(Town -> towns.id)
Price	numeric(9,2)	NOT NULL Check(price>=0)
customer_id	integer	NOT NULL FOREIGN KEY (customer_id -> customers.id)
order_time	Timestamp with time zone	NOT NULL
delivery_time	Timestamp with time zone	

2) **items_orders**- таблица осуществляющая отношения многим ко многим между items и orders, так как в реляционный БД не могут храниться массивы : по 1НФ элементы таблицы должны быть атомарны

Name	Data type	Constraints
id_order	integer	PK1.1 FOREIGN KEY(id_order -> order.id)
id_items	integer	PK1.2 FOREIGN KEY(id_items -> items.id)
count	integer	NOT NULL Check(count>0)
price	numeric(9,2)	NOT NULL Check(price>0)

3) **items** - таблица продуктов

Name	Data type	Constaraints
Id	serial	PK
Name	string	NOT NULL,UNIQUE
Cost	numeric(9,2)	NOT NULL CHECK(Cost>0)

4) **customers**- таблица клиентов

Name	Data type	Constraints
Id	serial	PK
Name	string	NOT NULL
Patronymic	string	
Surname	string	NOT NULL

5) **towns** - таблица городов

Name	Data type	Constraints
Id	Serial	PK
Address	string	NOT NULL,UNIQUE
Estimated_delivery_time	Timestamp with time zone	NOT NULL