Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**Отчёт**

**по заданию**

**Дисциплина: Управление информацией**

**Тема: «Разработка базы данных и информационной системы интернет-магазина»**

Работу выполнили А. Н. Мирошниченко, А. А. Сапунова, В. В. Чухиль

Направление подготовки 02.03.02 Фундаментальная информатика и

информационные технологии

Направленность (профиль) Математическое и программное обеспечение

компьютерных технологий

Преподаватель

канд. техн. наук, доц. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. В. Бессарабов

**Содержание**

[1 Спецификация бизнеса 3](#_Toc93441594)

[2 Диаграммы прецедентов 6](#_Toc93441595)

[3 Описание потока событий 13](#_Toc93441596)

[4 Концептуальная модель 14](#_Toc93441597)

[4.1 Пояснения к таблицам и их полям 15](#_Toc93441598)

[4.2 Пояснения к отношениям между таблицами 17](#_Toc93441599)

[5 Скрипт для создания и заполнения базы 19](#_Toc93441600)

[6 Процедурная часть приложения 28](#_Toc93441601)

[7 Пользовательский интерфейс 40](#_Toc93441602)

**1 Спецификация бизнеса**

Бизнес представляет собой книжный интернет-магазин. Покупатель книжного интернет-магазина может купить книги и канцелярские товары. Книги разделены по жанрам, а канцелярские товары по категориям. Среди жанров можно выделить следующие: приключения, фантастика, триллеры, психология, искусство и культура, компьютерная литература, школьные учебники и т.д. Среди категорий канцелярских товаров можно выделить следующие: бумага, линейка, ручка, степлер, тетрадь и т.д. При необходимости жанр или категорию можно добавить в интернет-магазин. У магазина есть склады. Товары на склады будут поставлять водители. Совершать покупки будут покупатели (клиенты). В магазине можно заказать товар (-ы) в количестве, ограниченном тем, что представлен (-ы) на центральном складе. Система складов является централизованной, иными словами, включает в себя один крупный центральный склад, где находится основная часть товаров, и филиальные склады.

**Доставка из центрального склада в филиальный склад.** Менеджер центрального склада получает книги и канцелярские товары от поставщиков при поступлении товаров на склад. Также он получает заказы от клиентов и передаёт заказы рабочим центрального склада на сборку. Стоит отметить, что заказы менеджер получает благодаря интернету, а каждый товар имеет артикул, благодаря чему рабочие понимают, из каких товаров им нужно собрать заказ. Рабочие центрального склада собирают заказы, сортируют заказы по адресам филиальных складов, помещают в машины и уведомляют менеджера о том, что заказы собраны. Менеджер фиксирует заказы, адреса филиальных складов, куда будет производиться доставка, и адреса пунктов выдачи заказов и составляет список из них. После того, как машина будет заполнена товарами, менеджер центрального склада присваивает всем упакованным заказам статус «в пути», а водитель центрального склада начинает развозить заказы по филиальным складам. Допустим, водитель центрального склада приехал на филиальный склад. Менеджер филиального склада принимает заказы, адрес доставки которых закреплён за этим филиальным складом, и передаёт их рабочим филиального склада.

**Доставка из филиального склада в пункты выдачи.** Менеджер филиального склада принимает заказы от водителя центрального склада и передаёт его рабочим филиального склада. Рабочие филиального склада сортируют заказы по адресам пунктов выдачи заказов и помещают их в машины. Менеджер филиального склада фиксирует заказы и адреса пунктов выдачи заказов, куда будет производиться доставка, и составляет список из них. После того, как машина будет заполнена товарами, водитель начинает развозить заказы в пункты выдачи заказов. С водителем всегда ездит грузчик. Допустим, водитель приехал в пункт выдачи заказов. Грузчик выгружает заказы в помещение пункта выдачи заказов.

**Личный кабинет клиента.** В личном кабинете клиента есть основные данные о клиенте (фамилия, имя, отчество, адрес, телефон, e-mail). Также в личном кабинете клиента есть личный счёт, куда будут возвращены деньги за заказ в случае невыполнения заказа. Заказ считается невыполненным, если клиент не забрал заказ из пункта выдачи заказов в течение 7 дней с того момента, как заказ доставлен в пункт выдачи заказов.

**Оформление заказа.** Для оформления заказа необходимо составить корзину товаров. Корзина товаров – это список товаров, которые клиент может заказать. Затем нужно заполнить информацию о заказе: способ оплаты (оплата при получении, банковская карта онлайн) и адрес пункта выдачи заказов. Основные данные о клиенте будут автоматически указаны в заказе из личного кабинета. Пункт выдачи заказов характеризуется адресом. Стоит отметить, что в тех городах, куда осуществляется доставка, есть хотя бы один пункт выдачи заказов. Если клиент не заберёт заказ из пункта выдачи заказов в течение 7 дней с того момента, как заказ будет доставлен в пункт выдачи заказов, заказ получит статус «не доставлен», водитель склада заберёт его из пункта выдачи заказов, увезёт на склад, а деньги за заказ поступят на счёт личного кабинета клиента. Если оплата будет производиться при получении, то после нажатия кнопки «оформить заказ» заказ примет статус «принят». Если оплата будет производиться онлайн, то непосредственно перед нажатием кнопки «оформить заказ» нужно указать данные банковской карты (номер, дата деактивации, CVC-код), с помощью которой будет производиться оплата, после чего заказ так же примет статус “принят”. Оплата онлайн банковской картой – единственный способ онлайн-оплаты в данном интернет-магазине. Стоит добавить, что исключены ситуации, когда заказ может потеряться в пути.

**Составление корзины товаров.** Для составления корзины товаров необходимо добавить каждый товар в корзину отдельно. Для того, чтобы добавить товар в корзину, клиент указывает количество товара, которое ему нужно, и нажимает кнопку «добавить в корзину».

**Товар.** Каждый товар характеризуется описанием (информацией о товаре), стоимостью в рублях и количеством товара в наличии на центральном складе. Также у товара могут быть отзывы.

**Отзыв.** Отзыв – это мнение отдельного покупателя, который приобрёл товар. Покупатель может оставить отзыв в интернет-магазине только для тех товаров, которые он приобрёл. В отзыве указываются фамилия, имя, отчество покупателя, дата покупки, оценка (от 1 до 5 звёзд), текстовый комментарий. При желании покупатель может приложить фотографии к отзыву.

**Заказы.** Заказы – список заказов, которые оформил клиент. Каждый заказ содержит следующую информацию: дата, время, статус (например, «принят», «передан в доставку», «в пути», «ожидает в пункте выдачи», «доставлен»), основные данные о клиенте из личного кабинета (фамилия, имя, отчество, адрес, телефон, e-mail), способ оплаты и данные банковской карты (номер, дата деактивации, CVC-код), если в качестве способа оплаты выбрана оплата онлайн. Если клиент указал ошибочные данные, то он может отредактировать их в течение часа с того момента, как заказ был оформлен.

**2 Диаграммы прецедентов**

В бизнесе представлены несколько ролей.

**Незарегистрированное в системе лицо.** Имеет доступ к описанию любого товара, представленного в интернет-магазине, а так же может просматривать отзывы к товарам.

**Клиент.** Зарегистрированное в системе лицо. Имеет такой же доступ, как и незарегистрированное лицо. Помимо этого, у клиента будет свой личный кабинет с информацией о себе, которую нужно заполнить при регистрации в системе. Клиент сможет добавлять товары в корзину, совершать покупки, просматривать свои заказы, оставлять отзывы к товарам.

**Администратор.** Зарегистрированное в системе лицо. Он может добавлять, редактировать и удалять товары. Актуальную информацию о товарах (например, появились новые товары или появились изменения в текущих) администратор получает от менеджера центрального склада. У него есть личный кабинет с функционалом администратора.

**Менеджер филиального склада.** Зарегистрированное в системе лицо. Он принимает заказы от водителя центрального склада и передаёт их рабочим филиального склада. На складе может взаимодействовать с любыми сотрудниками склада.

**Менеджер центрального склада.** Зарегистрированное в системе лицо. Может принимать заказ, редактировать статус заказа и отменять заказ. Также менеджер центрального склада передаёт новые заказы рабочим центрального склада, фиксирует заказы, адреса филиальных складов и адреса пунктов выдачи заказов. Вдобавок к этому, менеджер центрального склада принимает новые товары, которые привозят поставщики, фиксирует товары и уведомляет администратора об изменениях. На складе может взаимодействовать с любыми сотрудниками склада.

Стоит добавить, что любое зарегистрированное в системе лицо так же имеет функционал клиента.

**Рабочий склада.** Собирает заказы. На складе может взаимодействовать с любыми сотрудниками склада. Может быть как зарегистрированным в системе лицом, так и незарегистрированным в системе лицом.

**Водитель центрального склада.** Развозит заказы в филиальные склады. На складе может взаимодействовать с любыми сотрудниками склада. Может быть как зарегистрированным в системе лицом, так и незарегистрированным в системе лицом.

**Водитель филиального склада.** Развозит заказы в пункты выдачи заказов. На складе может взаимодействовать с любыми сотрудниками склада. Может быть как зарегистрированным в системе лицом, так и незарегистрированным в системе лицом.

**Грузчик филиального склада.** Работает вместе с водителем филиального склада. Выгружает заказы в помещение пункта выдачи заказов. На складе может взаимодействовать с любыми сотрудниками склада. Может быть как зарегистрированным в системе лицом, так и незарегистрированным в системе лицом.

**Оператор пункта выдачи заказов.** Зарегистрированное в системе лицо. Выдаёт заказ клиенту. Фиксирует полученные от грузчика филиального склада заказы. Устанавливает статус заказа «ожидает в пункте выдачи», когда заказ зафиксирован, и статус заказа «получен», когда заказ выдан клиенту.

На рисунках 1-5 приведены UML-диаграммы прецедентов, отражающие отношения между акторами и прецедентами.

Рисунок 1 – UML-диаграмма прецедентов, отражающая отношения между актором «Незарегистрированное лицо» и прецедентами, актором «Клиент» и прецедентами

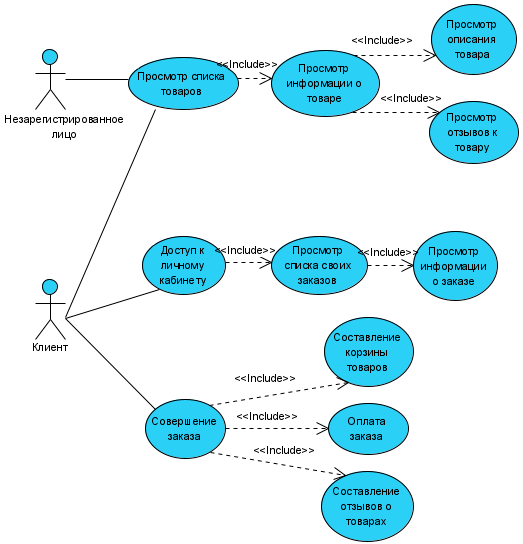


Рисунок 2 – UML-диаграмма прецедентов, отражающая отношения между актором «Администратор» и прецедентами

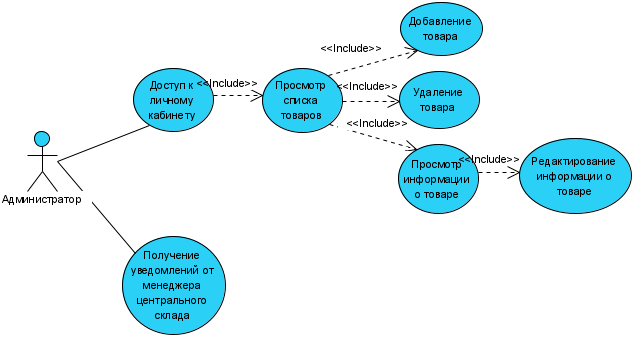


Рисунок 3 – UML-диаграмма прецедентов, отражающая отношения между актором «Менеджер филиального склада» и прецедентами, актором «Менеджер центрального склада» и прецедентами

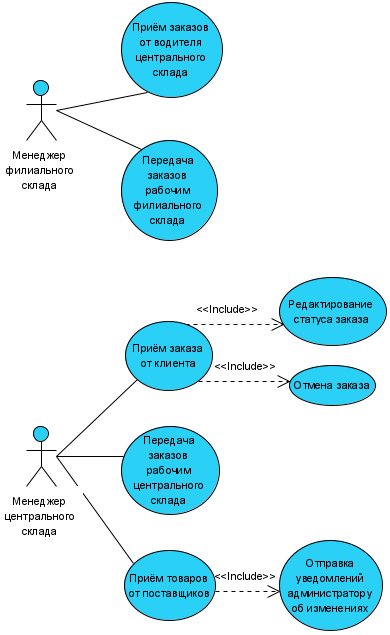


Рисунок 4 – UML-диаграмма прецедентов, отражающая отношения между актором «Рабочий склада» и прецедентами, актором «Водитель филиального склада» и прецедентами, актором «Водитель центрального склада» и прецедентами

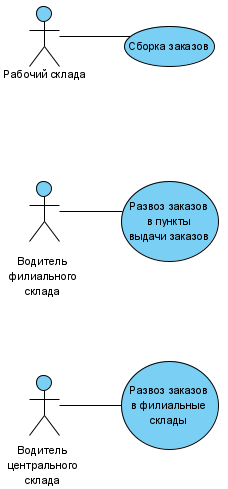
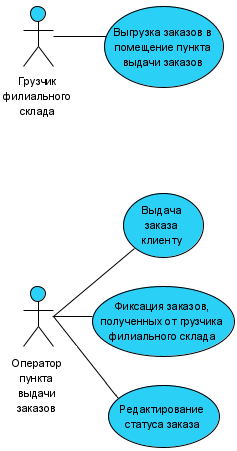


Рисунок 5 – UML-диаграмма прецедентов, отражающая отношения между актором «Грузчик филиального склада» и прецедентами, актором «Оператор пункта выдачи заказов» центрального склада и прецедентами



**3 Описание потока событий**

В процессе функционирования в системе происходят различные события: клиенты оформляют заказы, водители перевозят товары между складами, поступают новые товары на продажу. Все события происходят в некотором порядке. На рисунке 6 показана последовательность событий для поступления на склад товара и последующего заказа от клиента.

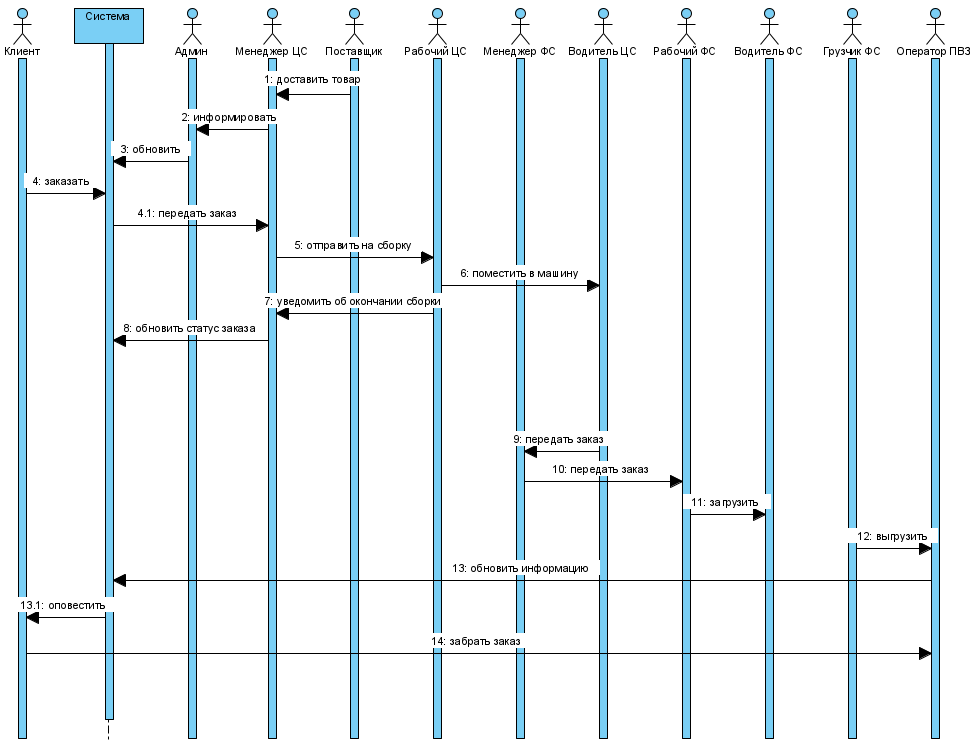


Рисунок 6 – Последовательность событий для поступления на склад товара и последующего заказа от клиента

**4 Концептуальная модель**

На рисунке 7 приведена структура базы данных в виде ER-диаграммы.

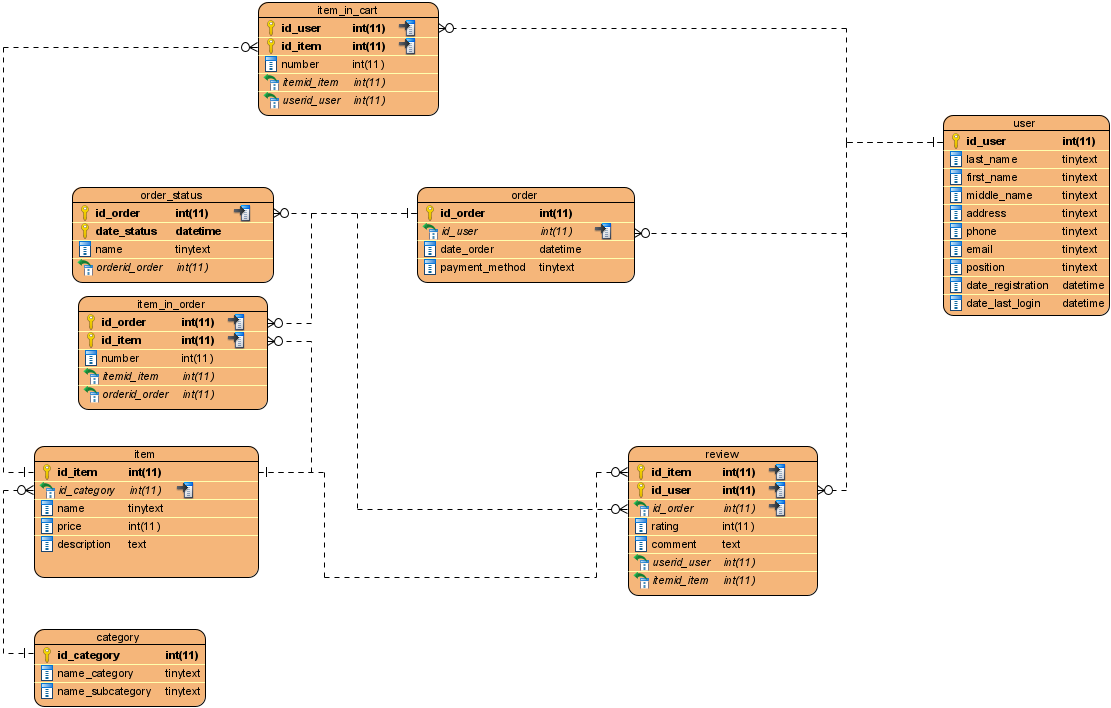
****

Рисунок 7 – Структура базы данных в виде ER-диаграммы

**4.1 Пояснения к таблицам и их полям**

**client – таблица с пользователями интернет-магазина.**

1. PK id\_user – уникальный идентификатор пользователя;
2. last\_name – фамилия пользователя;
3. first\_name – имя пользователя;
4. middle\_name – отчество пользователя;
5. address – адрес пользователя;
6. phone – номер телефона пользователя;
7. email – электронная почта пользователя;
8. position – роль пользователя в интернет-магазине;
9. date\_registration – дата регистрации пользователя в интернет‑магазине;
10. date\_last\_login – дата последнего входа пользователя в интернет‑магазин.

**category – таблица с категориями товаров, представленных в интернет-магазине.**

1. PK id\_category – уникальный идентификатор категории товаров;
2. name\_category – наименование категории товаров;
3. name\_subcategory – наименование подкатегории товаров (жанра для книги или категории для канцелярских товаров).

**item – таблица с товарами, представленными в интернет-магазине.**

1. PK id\_item – уникальный идентификатор товара;
2. FK id\_category – идентификатор категории товаров;
3. name – название товара;
4. price – цена за единицу товара;
5. description – описание товара.

**item\_in\_cart – таблица с товарами в корзине пользователя.**

1. PK, FK id\_user – уникальный идентификатор пользователя;
2. PK, FK id\_item – уникальный идентификатор товара;
3. number – количество товара в корзине.

**order – таблица с заказами пользователя.**

1. PK id\_order – уникальный идентификатор заказа;
2. FK id\_user – идентификатор пользователя;
3. date\_order – дата заказа.

**item\_in\_order – таблица с товарами в заказе пользователя.**

1. PK, FK id\_order – уникальный идентификатор заказа;
2. PK, FK id\_item – уникальный идентификатор товара;
3. number – количество товара в заказе.

**review – таблица с отзывами на товары, представленные в интернет-магазине.**

1. PK, FK id\_item – уникальный идентификатор товара;
2. PK, FK id\_user – уникальный идентификатор пользователя;
3. FK id\_order – идентификатор заказа;
4. rating – оценка товару (от 1 до 5 звёзд);
5. comment – комментарий к товару.

**order\_status – таблица со статусами передвижения заказа, оформленного в интернет-магазине;**

1. PK, FK id\_order – уникальный идентификатор заказа;
2. PK date\_status – дата установки статуса заказа;
3. name – название статуса заказа.

**4.2 Пояснения к отношениям между таблицами**

**user – таблица с пользователями интернет-магазина.**

1. Между таблицами user и order отношение «один ко многим», потому что у пользователя может быть много заказов.
2. Между таблицами user и item отношение «многие ко многим», потому что любой клиент может добавить любой товар в корзину или оставить к нему отзыв (если клиент этот товар заказывал). Для добавления товара в корзину используется таблица item\_in\_cart для связи user и item (ключи id\_user и id\_item), а чтобы оставить отзыв, используется таблица Review так же для связи User и Item (ключи id\_user и id\_item).

**order – таблица с заказами пользователя.**

1. Между таблицами order и user отношение «многие к одному», потому что у пользователя может быть много заказов.
2. Между таблицами order и item отношение «многие ко многим», потому что любой заказ может содержать в себе разные товары. Для отдельного товара, который содержится в заказе, используется таблица item\_in\_order для связи order и item (ключи id\_order и id\_item).
3. Между таблицами order и review отношение «один ко многим», потому что в заказе может быть много товаров, а отзывы можно оставить к нескольким товарам из заказа.
4. Между таблицами order и order\_status отношение «один ко многим», потому что для одного заказа можно установить несколько статусов.

**order\_status – таблица со статусами передвижения заказа, оформленного в интернет-магазине.**

Между таблицами order\_status и order отношение «многие к одному», потому что для одного заказа можно установить несколько статусов.

**item – таблица с товарами, представленными в интернет-магазине.**

Между таблицами item и category отношение «многие к одному», потому что несколько товаров могут быть одной категории.

**category – таблица с категориями товаров, представленных в интернет-магазине.**

Между таблицами category и item отношение «один ко многим», потому что несколько товаров могут быть одной категории.

**5 Скрипт для создания и заполнения базы**

CREATE TABLE `category1` (

`id\_category` int(11) NOT NULL,

`name\_category` tinytext NOT NULL,

`name\_subcategory` tinytext NOT NULL

);

INSERT INTO `category1` (`id\_category`, `name\_category`, `name\_subcategory`) VALUES

(1, 'книги', 'психология'),

(2, 'книги', 'фантастика'),

(3, 'книги', 'триллеры'),

(4, 'книги', 'классическая'),

(5, 'книги', 'техническая'),

(6, 'книги', 'школьные учебники'),

(7, 'книги', 'драма'),

(8, 'книги', 'для детей'),

(9, 'книги', 'биографии'),

(10, 'канцелярские товары', 'бумага'),

(11, 'канцелярские товары', 'линейка'),

(12, 'канцелярские товары', 'ручка'),

(13, 'канцелярские товары', 'степлер'),

(14, 'канцелярские товары', 'тетрадь');

CREATE TABLE `item1` (

`id\_item` int(11) NOT NULL,

`id\_category` int(11) NOT NULL,

`name` tinytext NOT NULL,

`price` int(11) NOT NULL,

`description` text NOT NULL

);

INSERT INTO `item1` (`id\_item`, `id\_category`, `name`, `price`, `description`) VALUES

(1, 1, 'Мастер и Маргарита', 164, 'Описание товара 1.'),

(2, 2, 'Мы', 147, 'Описание товара 2.'),

(3, 3, 'Отцы и дети', 146, 'Описание товара 3.'),

(4, 4, 'Герой нашего времени', 147, 'Описание товара 4.'),

(5, 5, 'Обломов', 189, 'Описание товара 5.'),

(6, 6, 'Математика', 150, 'Описание товара 6.'),

(7, 7, 'Братья Карамазовы', 247, 'Описание товара 7.'),

(8, 8, 'Война и мир', 295, 'Описание товара 8.'),

(9, 9, 'Самый богатый человек в Вавилоне', 120, 'Описание товара 9.'),

(10, 10, 'Бумага офисная А4, 500 л.', 375, 'Описание товара 10.'),

(11, 10, 'Стикер, 100 л.', 43, 'Описание товара 11.'),

(12, 11, 'Линейка пластиковая 20 см, гибкая', 62, 'Описание товара 12.'),

(13, 11, 'Линейка деревянная 25 см', 13, 'Описание товара 13.'),

(14, 12, 'Ручка шариковая синяя, 0.7мм', 7, 'Описание товара 14.'),

(15, 12, 'Ручка гелевая красная, 0.5мм', 40, 'Описание товара 15.'),

(16, 13, 'Степлер, до 30 листов, чёрный', 334, 'Описание товара 16.'),

(17, 13, 'Степлер мощный, до 120 листов, чёрный', 3906, 'Описание товара 17.'),

(18, 14, 'Тетрадь 48 л., А5, клетка, скрепка', 41, 'Описание товара 18.'),

(19, 14, 'Тетрадь 80 л., А5, клетка, спираль, двойная обложка', 101, 'Описание товара 19.');

CREATE TABLE `item\_in\_cart1` (

`id\_user` int(11) NOT NULL,

`id\_item` int(11) NOT NULL,

`number` int(11) NOT NULL

);

INSERT INTO `item\_in\_cart1` (`id\_user`, `id\_item`, `number`) VALUES

(1, 1, 1),

(3, 1, 2),

(8, 1, 1),

(12, 1, 100),

(1, 2, 2),

(8, 2, 1),

(2, 3, 5),

(5, 4, 4),

(3, 5, 1),

(5, 5, 1),

(13, 8, 7),

(3, 10, 10),

(7, 11, 1),

(7, 12, 3),

(13, 12, 2),

(13, 13, 5),

(13, 15, 1),

(19, 17, 17),

(9, 18, 5),

(9, 19, 5);

CREATE TABLE `item\_in\_order1` (

`id\_order` int(11) NOT NULL,

`id\_item` int(11) NOT NULL,

`number` int(11) NOT NULL

);

INSERT INTO `item\_in\_order1` (`id\_order`, `id\_item`, `number`) VALUES

(1, 1, 1),

(1, 2, 2),

(2, 3, 5),

(3, 1, 2),

(3, 5, 1),

(3, 10, 10),

(5, 4, 4),

(5, 5, 1),

(7, 11, 1),

(7, 12, 3),

(8, 1, 1),

(8, 2, 1),

(9, 18, 5),

(9, 19, 5),

(12, 1, 100),

(13, 8, 7),

(13, 12, 2),

(13, 13, 5),

(13, 15, 1),

(19, 17, 17);

CREATE TABLE `order1` (

`id\_order` int(11) NOT NULL,

`id\_user` int(11) NOT NULL,

`date\_order` datetime NOT NULL

);

INSERT INTO `order1` (`id\_order`, `id\_user`, `date\_order`) VALUES

(1, 1, '2020-01-01 00:00:00'),

(2, 1, '2020-01-01 08:00:00'),

(3, 2, '2021-09-01 20:30:00'),

(4, 3, '2021-06-16 17:40:00'),

(5, 6, '2020-07-17 16:15:00'),

(6, 6, '2021-12-20 18:18:00'),

(7, 6, '2019-01-30 19:01:00'),

(8, 7, '2020-02-29 20:02:00'),

(9, 8, '2019-03-25 13:30:00'),

(10, 10, '2022-01-24 13:30:05'),

(11, 12, '2020-04-01 13:31:00'),

(12, 12, '2021-04-04 10:00:00'),

(13, 13, '2020-05-10 01:55:00'),

(14, 14, '2020-01-10 05:37:00'),

(15, 15, '2021-11-10 05:43:00'),

(16, 16, '2021-10-05 16:20:00'),

(17, 17, '2020-12-01 12:00:00'),

(18, 17, '2020-08-18 12:59:00'),

(19, 19, '2020-09-09 19:09:00'),

(20, 20, '2020-01-01 19:10:59');

CREATE TABLE `order\_status1` (

`id\_order` int(11) NOT NULL,

`date\_status` datetime NOT NULL,

`name` enum('принят','передан в доставку','в пути','ожидает в пункте выдачи','доставлен','не доставлен') NOT NULL

);

INSERT INTO `order\_status1` (`id\_order`, `date\_status`, `name`) VALUES

(1, '2020-01-01 00:00:00', 'принят'),

(1, '2020-01-01 00:30:00', 'передан в доставку'),

(1, '2020-01-01 00:35:00', 'в пути'),

(1, '2020-01-02 08:30:00', 'ожидает в пункте выдачи'),

(1, '2020-01-02 11:40:00', 'доставлен'),

(2, '2021-11-05 00:00:00', 'принят'),

(2, '2021-11-05 06:15:14', 'передан в доставку'),

(2, '2021-11-05 06:35:00', 'в пути'),

(2, '2021-12-01 00:00:00', 'ожидает в пункте выдачи'),

(2, '2021-12-08 00:00:00', 'не доставлен'),

(3, '2022-01-15 07:30:00', 'принят'),

(3, '2022-01-15 07:42:00', 'передан в доставку'),

(3, '2022-01-15 07:44:00', 'в пути'),

(3, '2022-01-18 16:50:00', 'ожидает в пункте выдачи'),

(4, '2022-02-23 17:02:00', 'принят'),

(4, '2022-02-23 17:50:00', 'передан в доставку'),

(4, '2022-02-23 18:00:01', 'в пути'),

(5, '2020-05-17 13:42:00', 'принят'),

(6, '2020-05-17 13:33:00', 'принят'),

(6, '2020-05-17 13:36:00', 'передан в доставку'),

(6, '2020-05-17 15:01:00', 'в пути'),

(6, '2020-06-17 12:22:00', 'ожидает в пункте выдачи'),

(6, '2020-06-23 12:58:00', 'доставлен');

CREATE TABLE `review1` (

`id\_item` int(11) NOT NULL,

`id\_user` int(11) NOT NULL,

`id\_order` int(11) NOT NULL,

`rating` enum('1','2','3','4','5') NOT NULL,

`comment` text NOT NULL

);

INSERT INTO `review1` (`id\_item`, `id\_user`, `id\_order`, `rating`, `comment`) VALUES

(1, 1, 1, '5', 'мне очень понравилась книга'),

(1, 2, 3, '2', 'мне очень не понравилось, слишком много лишнего'),

(1, 7, 8, '4', 'круто'),

(1, 12, 12, '5', 'книженция зашла, норм, ещё возьму'),

(2, 1, 1, '3', 'могло быть лучше'),

(2, 7, 8, '1', 'ну такое'),

(3, 1, 2, '4', 'интересная книга, рекомендую к чтению'),

(4, 6, 5, '1', 'не понимаю, как это вообще можно читать, ужасно скучно'),

(5, 2, 3, '5', 'Обломов очень классное произведение, прочитал уже 3 раза'),

(5, 6, 5, '4', 'я сижу у реки и читаю, легко читается, класс, Спасибо!!! но не хватает изюминки'),

(8, 13, 13, '5', 'хоть и довольно длинное произведение, в школе такое не читал, но сейчас, после прочтения, я впечатлён :)'),

(10, 2, 3, '5', 'бумага как бумага'),

(11, 6, 7, '4', 'обычные стикеры, моему ребёнку в школе пригодились'),

(12, 6, 7, '4', 'норм линейка'),

(12, 13, 13, '1', 'класс, линейка гнётся, но я её сломал('),

(13, 13, 13, '2', 'слабая линейка, под моим весом сломалась и сгорела'),

(15, 13, 13, '5', 'работы учеников проверять теперь одно удовольствие, ручка безумно удобная!!!'),

(17, 19, 19, '3', 'степлер реально мощный, пробил себе палец, в больнице сейчас, поэтому ставлю 3 из-за плохой безопасности'),

(18, 8, 9, '3', 'так себе тетрадь, промокает под дождём'),

(19, 8, 9, '5', 'крутая тетрадь, все мои лекции вместились!!1');

CREATE TABLE `user1` (

`id\_user` int(11) NOT NULL,

`last\_name` tinytext NOT NULL,

`first\_name` tinytext NOT NULL,

`middle\_name` tinytext NOT NULL,

`address` tinytext NOT NULL,

`phone` tinytext NOT NULL,

`email` tinytext NOT NULL,

`position` enum('клиент','менеджер','администратор','оператор') NOT NULL,

`date\_registration` datetime NOT NULL,

`date\_last\_login` datetime NOT NULL

);

INSERT INTO `user1` (`id\_user`, `last\_name`, `first\_name`, `middle\_name`, `address`, `phone`, `email`, `position`, `date\_registration`, `date\_last\_login`) VALUES

(1, 'Малеж', 'Елизавета', 'Юрьевна', 'Краснодар, Российская 42', '+7-999-333-45-45', 'malezh@mail.ru', 'клиент', '2020-09-09 00:00:00', '2022-01-16 00:00:00'),

(2, 'Кондратенко', 'Данна', 'Александровна', 'Краснодар, Ставропольская 11', '+7-909-343-54-00', 'kondratenko@gmail.com', 'клиент', '2021-01-26 00:00:00', '2021-03-16 00:00:00'),

(3, 'Антонюк', 'Виктория', 'Дмитриевна', 'Краснодар, Ленина 10', '+7-900-300-00-02', 'antonyuk@mail.er', 'администратор', '2018-01-01 12:00:00', '2022-01-16 00:00:00'),

(4, 'Гречка', 'Екатерина', 'Владимировна', 'Краснодар, Димитрова 105', '+7-809-222-09-70', 'kasha@u.uwu', 'менеджер', '2018-01-01 06:00:00', '2022-01-15 00:00:00'),

(5, 'Мирошниченко', 'Артём', 'Николаевич', 'Краснодар, Ставропольская 149', '+7-800-555-35-35', 'artemgenius@gmail.com', 'оператор', '2018-01-01 00:00:00', '2022-01-10 00:00:00'),

(6, 'Жданов', 'Вадим', 'Геннадьевич', 'Майкоп, Пушкина 202', '+7-963-963-93-69', 'zhdanov@inside.site', 'клиент', '2021-09-20 00:00:00', '2021-09-20 00:00:00'),

(7, 'Ерёменко', 'Кирилл', 'Егорович', 'Севастополь, Севастопольская 1', '+7-803-888-66-66', 'eremenko@ma.du', 'клиент', '2021-10-10 10:28:05', '2022-03-16 06:30:00'),

(8, 'Саенко', 'Илья', 'Ильич', 'Москва, Краснодарская 666', '+7-666-666-77-01', 'saenko@moscow.com', 'клиент', '2020-06-06 20:20:20', '2021-03-17 20:37:04'),

(9, 'Панченко', 'Андрей', 'Артёмович', 'Санкт-Петербург, Кушкина 444', '+7-442-222-12-86', 'panchenko@mail.mail', 'клиент', '2019-08-18 00:20:30', '2022-09-08 23:01:00'),

(10, 'Соколов', 'Даниэль', 'Елисеевич', 'Анапа, Панамская 9', '+7-990-099-00-00', 'sokolov@brand.russia', 'клиент', '2020-12-20 10:42:00', '2021-12-22 18:37:00'),

(11, 'Готлиб', 'Артём', 'Артёмович', 'Новосибирск, Новосибирская 110', '+7-383-566-30-50', 'gotlib@gmail.com', 'клиент', '2020-01-02 01:10:00', '2021-01-10 00:00:00'),

(12, 'Булаткин', 'Егор', 'Николаевич', 'Пенза, Пензенская 220', '+7-841-294-49-84', 'bulatkin@gmail.com', 'клиент', '2020-03-04 02:20:00', '2021-02-11 00:00:00'),

(13, 'Моргенштерн', 'Алишер', 'Тагирович', 'Уфа, Уфимская 330', '+7-347-863-98-12', 'morgenshtern@gmail.com', 'клиент', '2020-05-06 03:30:00', '2021-03-12 00:00:00'),

(14, 'Алексеев', 'Иван', 'Александрович', 'Ярцево, Легендарная 16', '+7-481-480-99-71', 'kondratenko@gmail.com', 'клиент', '2020-07-08 04:40:00', '2021-04-13 00:00:00'),

(15, 'Фёдоров', 'Мирон', 'Янович', 'Мурино, Амурская 440', '+7-812-626-18-49', 'fedorov@gmail.com', 'клиент', '2020-09-10 05:50:00', '2021-05-14 00:00:00'),

(16, 'Мингазов', 'Марат', 'Фирдусович', 'Йошкар-Ола, Крымская 550', '+7-836-077-04-64', 'mingazov@gmail.com', 'клиент', '2020-11-22 06:10:00', '2021-06-15 00:00:00'),

(17, 'Самедов', 'Тимур', 'Тофикович', 'Тобольск, Тобольская 600', '+7-495-450-72-88', 'samedov@gmail.com', 'клиент', '2021-01-05 07:20:00', '2021-07-16 00:00:00'),

(18, 'Жалелов', 'Адиль', 'Оралбекович', 'Майкоп, Майкопская 775', '+7-877-693-54-14', 'zhzhzh@gg.creative', 'клиент', '2021-02-10 08:30:00', '2021-08-17 00:00:00'),

(19, 'Савченко', 'Олег', 'Вадимович', 'Краснодар, Лермонтова 837', '+7-861-039-20-35', 'savva@cringe.net', 'клиент', '2021-03-15 09:40:00', '2021-09-18 00:00:00'),

(20, 'Бумага', 'Владислав', 'Андреевич', 'Минск, Минеральная 999', '+7-375-441-13-43', 'paperdog@google.com', 'клиент', '2021-04-20 10:50:00', '2021-10-25 00:00:00');

**// Индексы**

ALTER TABLE `category1`

ADD PRIMARY KEY (`id\_category`);

ALTER TABLE `item1`

ADD PRIMARY KEY (`id\_item`),

ADD KEY `id\_category` (`id\_category`);

ALTER TABLE `item\_in\_cart1`

ADD PRIMARY KEY (`id\_item`,`id\_user`) USING BTREE,

ADD KEY `id\_user` (`id\_user`),

ADD KEY `id\_item` (`id\_item`);

ALTER TABLE `item\_in\_order1`

ADD PRIMARY KEY (`id\_order`,`id\_item`) USING BTREE,

ADD KEY `id\_order` (`id\_order`,`id\_item`),

ADD KEY `id\_item` (`id\_item`);

ALTER TABLE `order1`

ADD PRIMARY KEY (`id\_order`),

ADD KEY `id\_user` (`id\_user`);

ALTER TABLE `order\_status1`

ADD PRIMARY KEY (`id\_order`,`date\_status`) USING BTREE,

ADD KEY `id\_order` (`id\_order`);

ALTER TABLE `review1`

ADD PRIMARY KEY (`id\_item`,`id\_user`) USING BTREE,

ADD KEY `id\_item` (`id\_item`),

ADD KEY `id\_user` (`id\_user`,`id\_order`),

ADD KEY `id\_order` (`id\_order`);

ALTER TABLE `user1`

ADD PRIMARY KEY (`id\_user`);

**// Ограничения ключей**

ALTER TABLE `item1`

ADD CONSTRAINT `item1\_ibfk\_1` FOREIGN KEY (`id\_category`) REFERENCES `category1` (`id\_category`);

ALTER TABLE `item\_in\_cart1`

ADD CONSTRAINT `item\_in\_cart1\_ibfk\_1` FOREIGN KEY (`id\_user`) REFERENCES `user1` (`id\_user`),

ADD CONSTRAINT `item\_in\_cart1\_ibfk\_2` FOREIGN KEY (`id\_item`) REFERENCES `item1` (`id\_item`);

ALTER TABLE `item\_in\_order1`

ADD CONSTRAINT `item\_in\_order1\_ibfk\_1` FOREIGN KEY (`id\_item`) REFERENCES `item1` (`id\_item`),

ADD CONSTRAINT `item\_in\_order1\_ibfk\_2` FOREIGN KEY (`id\_order`) REFERENCES `order1` (`id\_order`);

ALTER TABLE `order1`

ADD CONSTRAINT `order1\_ibfk\_1` FOREIGN KEY (`id\_user`) REFERENCES `user1` (`id\_user`);

ALTER TABLE `order\_status1`

ADD CONSTRAINT `order\_status1\_ibfk\_1` FOREIGN KEY (`id\_order`) REFERENCES `order1` (`id\_order`);

ALTER TABLE `review1`

ADD CONSTRAINT `review1\_ibfk\_1` FOREIGN KEY (`id\_item`) REFERENCES `item1` (`id\_item`),

ADD CONSTRAINT `review1\_ibfk\_2` FOREIGN KEY (`id\_order`) REFERENCES `order1` (`id\_order`),

ADD CONSTRAINT `review1\_ibfk\_3` FOREIGN KEY (`id\_user`) REFERENCES `user1` (`id\_user`);

**6 Процедурная часть приложения**

**// Транзакции**

Set autocommit=0;

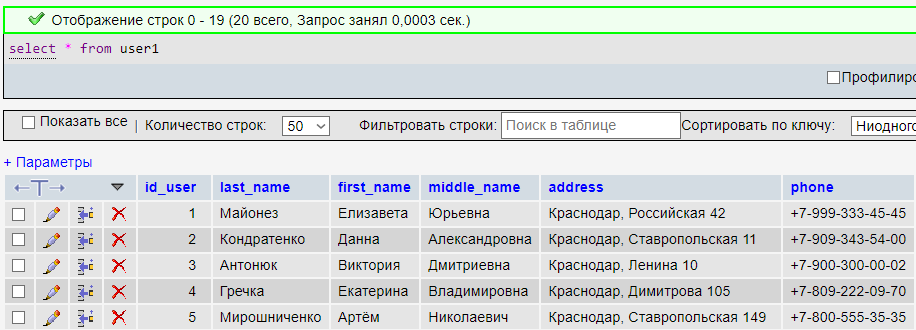
Start transaction;

DELETE FROM order\_status1 WHERE id\_order=5;



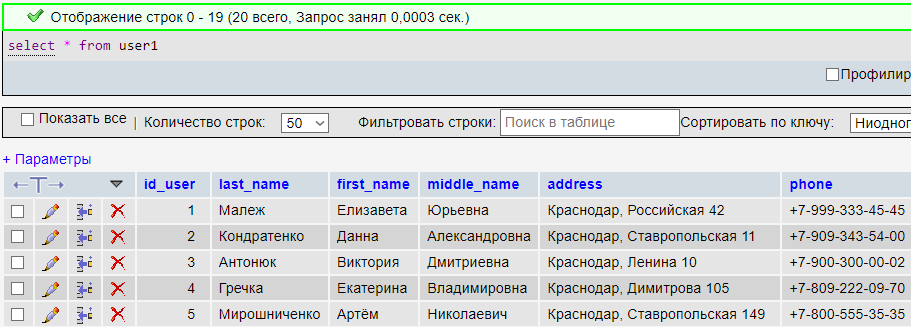
UPDATE user1 SET last\_name=’Майонез’ WHERE id\_user=1;

Select \* from user1;



ROLLBACK;

Select \* from user1;



Commit;

**// Вывести номера заказов, товары из этих заказов, цены на товары, даты заказов и имена клиентов**

select order1.id\_order 'Order', item1.name 'Item', item1.price 'Price', order1.date\_order 'Date', user1.last\_name 'Surname'

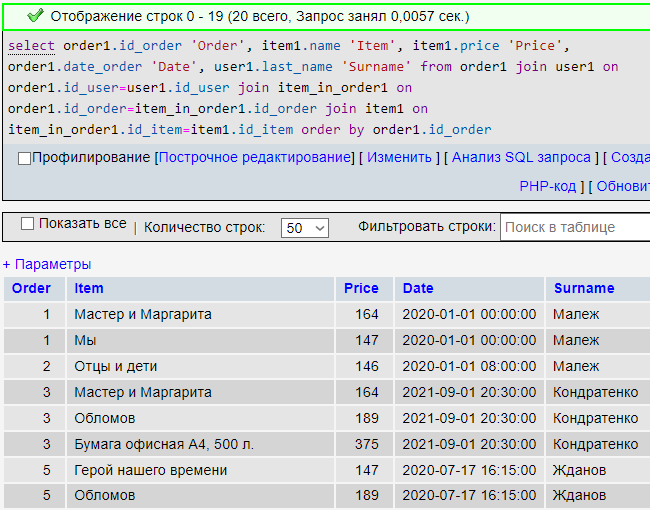
from order1

join user1 on order1.id\_user=user1.id\_user

join item\_in\_order1 on order1.id\_order=item\_in\_order1.id\_order

join item1 on item\_in\_order1.id\_item=item1.id\_item

order by order1.id\_order;



**// Вывести все товары (названия, цену, категорию и подкатегорию), имеющие категорию “книги”**

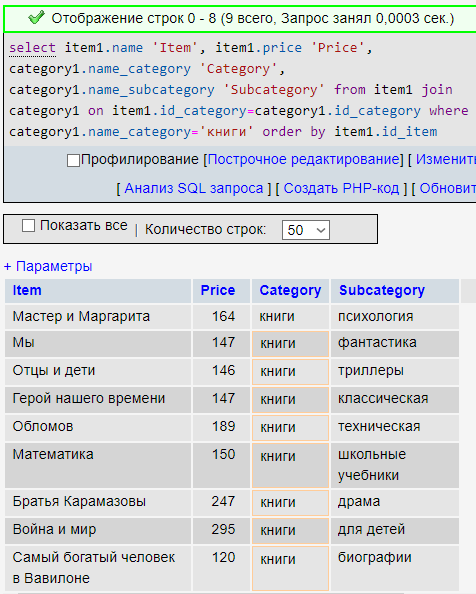
select item1.name 'Item', item1.price 'Price', category1.name\_category 'Category', category1.name\_subcategory 'Subcategory'

from item1

join category1 on item1.id\_category=category1.id\_category

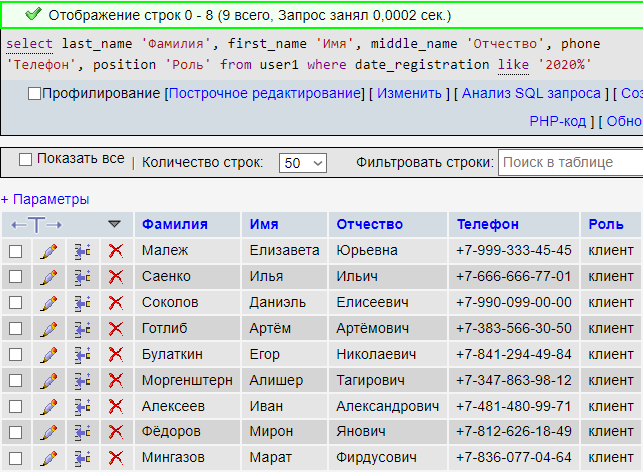
where category1.name\_category='книги'

order by item1.id\_item;



**// Вывести ФИО, номер телефона и роль пользователей, которые зарегистрировались в 2020 году**

select last\_name ‘Фамилия’, first\_name ‘Имя’, middle\_name ‘Отчество’, phone ‘Телефон’, position ‘Роль’ from user1 where date\_registration like ‘2020%’

****

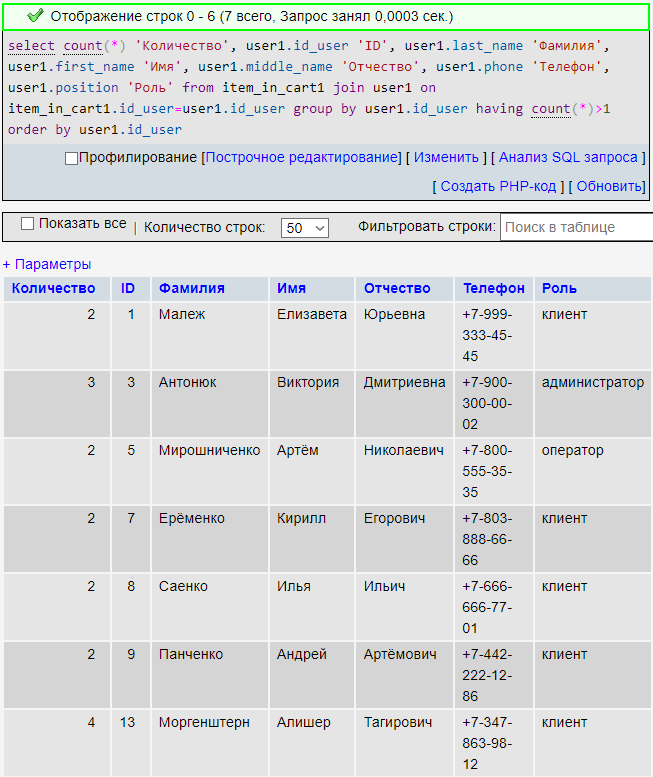
**// Вывести ФИО, номер телефона и роль пользователей, у которых в корзине больше 1 товара**

select count(\*) 'Количество', user1.id\_user 'ID', user1.last\_name 'Фамилия', user1.first\_name 'Имя', user1.middle\_name 'Отчество', user1.phone 'Телефон', user1.position 'Роль' from item\_in\_cart1 join user1 on item\_in\_cart1.id\_user=user1.id\_user

group by user1.id\_user

having count(\*)>1

order by user1.id\_user



**// Вывести фамилию пользователя, название товара и оценку из отзывов, оценка из которых больше 3**

select user1.last\_name 'Фамилия', item1.name 'Товар', review1.rating 'Оценка'

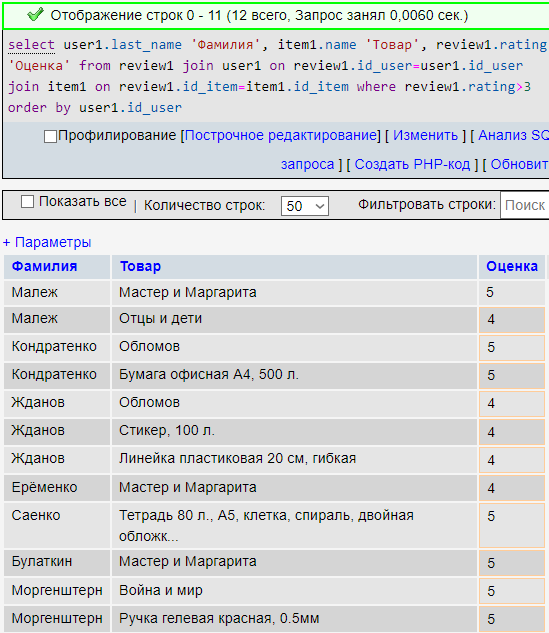
from review1

join user1 on review1.id\_user=user1.id\_user

join item1 on review1.id\_item=item1.id\_item

where review1.rating>3

order by user1.id\_user

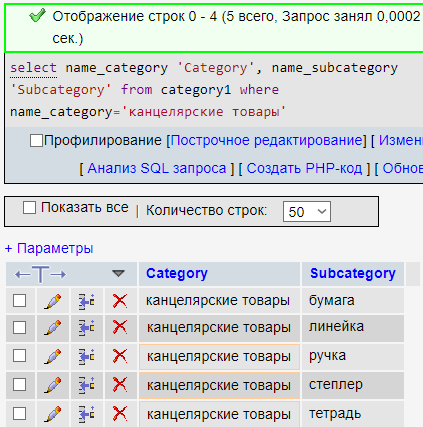


**// Вывести все подкатегории категории “канцелярские товары”**

select name\_category 'Category', name\_subcategory 'Subcategory'

from category1

where name\_category='канцелярские товары'

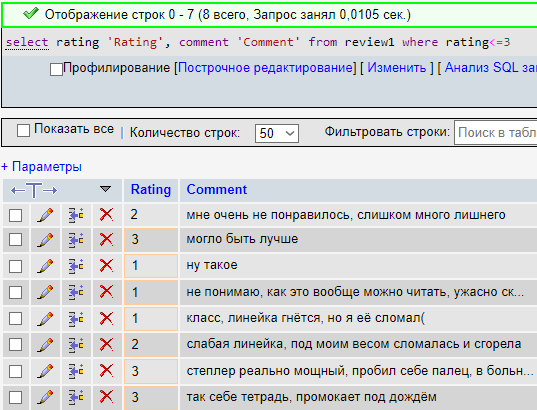


**// Вывести все отзывы с оценкой 3 и ниже**

select rating 'Rating', comment 'Comment'

from review1

where rating<=3



**// Вывести сумму всех заказов клиента с id\_user=1**

select user1.last\_name 'Фамилия', sum(item1.price \* item\_in\_order1.number) 'Сумма'

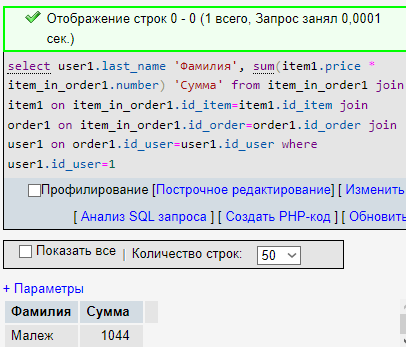
from item\_in\_order1

join item1 on item\_in\_order1.id\_item=item1.id\_item

join order1 on item\_in\_order1.id\_order=order1.id\_order

join user1 on order1.id\_user=user1.id\_user

where user1.id\_user=1



**// Вывести фамилию, телефон и почту клиентов, которые совершили хотя бы 1 заказ**

select user1.last\_name 'Фамилия', user1.phone 'Телефон', user1.email 'Почта'

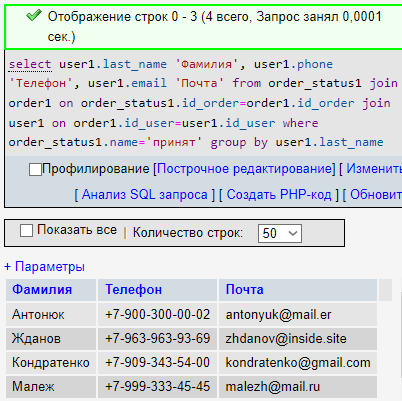
from order\_status1

join order1 on order\_status1.id\_order=order1.id\_order

join user1 on order1.id\_user=user1.id\_user

where order\_status1.name='принят'

group by user1.last\_name



**// Отсортировать книги, представленные в магазине, по цене, в порядке убывания**

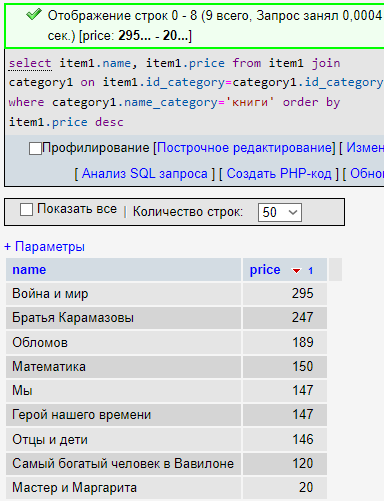
select item1.name, item1.price

from item1

join category1 on item1.id\_category=category1.id\_category

where category1.name\_category='книги'

order by item1.price desc



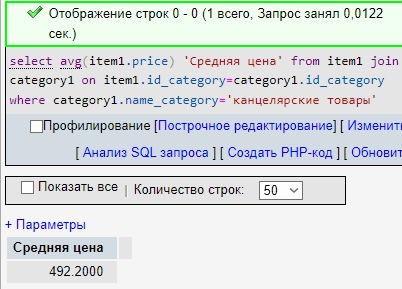
**// Посчитать среднюю цену на канцелярский товар в магазине**

select avg(item1.price) 'Средняя цена'

from item1

join category1 on item1.id\_category=category1.id\_category

where category1.name\_category='канцелярские товары'



**// Триггер удаляет строку из таблицы после её добавления, если цена книги будет меньше или равной 0**

DELIMITER $$

CREATE TRIGGER trig1

AFTER INSERT

ON item1 FOR EACH ROW

BEGIN

IF NEW.price <= 0 THEN

DELETE item1

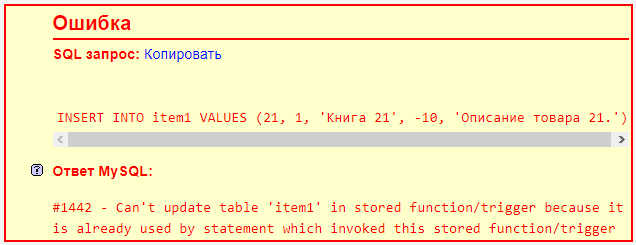
FROM item1

END IF;

END$$

DELIMITER ;

INSERT INTO item1 VALUES (21, 1, ‘Книга 21’, -10, ‘Описание товара 21.’);



INSERT INTO item1 VALUES (21, 1, ‘Книга 21’, 10, ‘Описание товара 21.’);



DELETE FROM item1 WHERE price<12 AND name LIKE ‘К%’



**// Триггер оставляет старые значения строки, если после обновления цена книги будет меньше или равной 0**

DELIMITER $$

CREATE TRIGGER trig11

AFTER UPDATE

ON item1 FOR EACH ROW

BEGIN

IF NEW.price <= 0 THEN

UPDATE item1

SET id\_item = OLD.id\_item,

id\_category = OLD.id\_category,

name = OLD.name,

price = OLD.price,

description = OLD.description;

END IF;

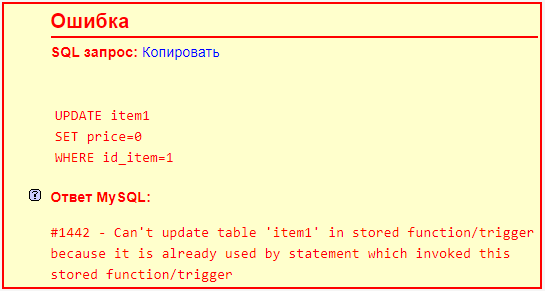
END$$

DELIMITER ;

UPDATE item1

SET price=0

WHERE id\_item=1;



UPDATE item1

SET price=20

WHERE id\_item=1;



**7 Пользовательский интерфейс**

На рисунках 8-15 приведены образцы пользовательского интерфейса.

Рисунок 8 – Раздел «Книги»



Рисунок 9 – Список книг и их жанры

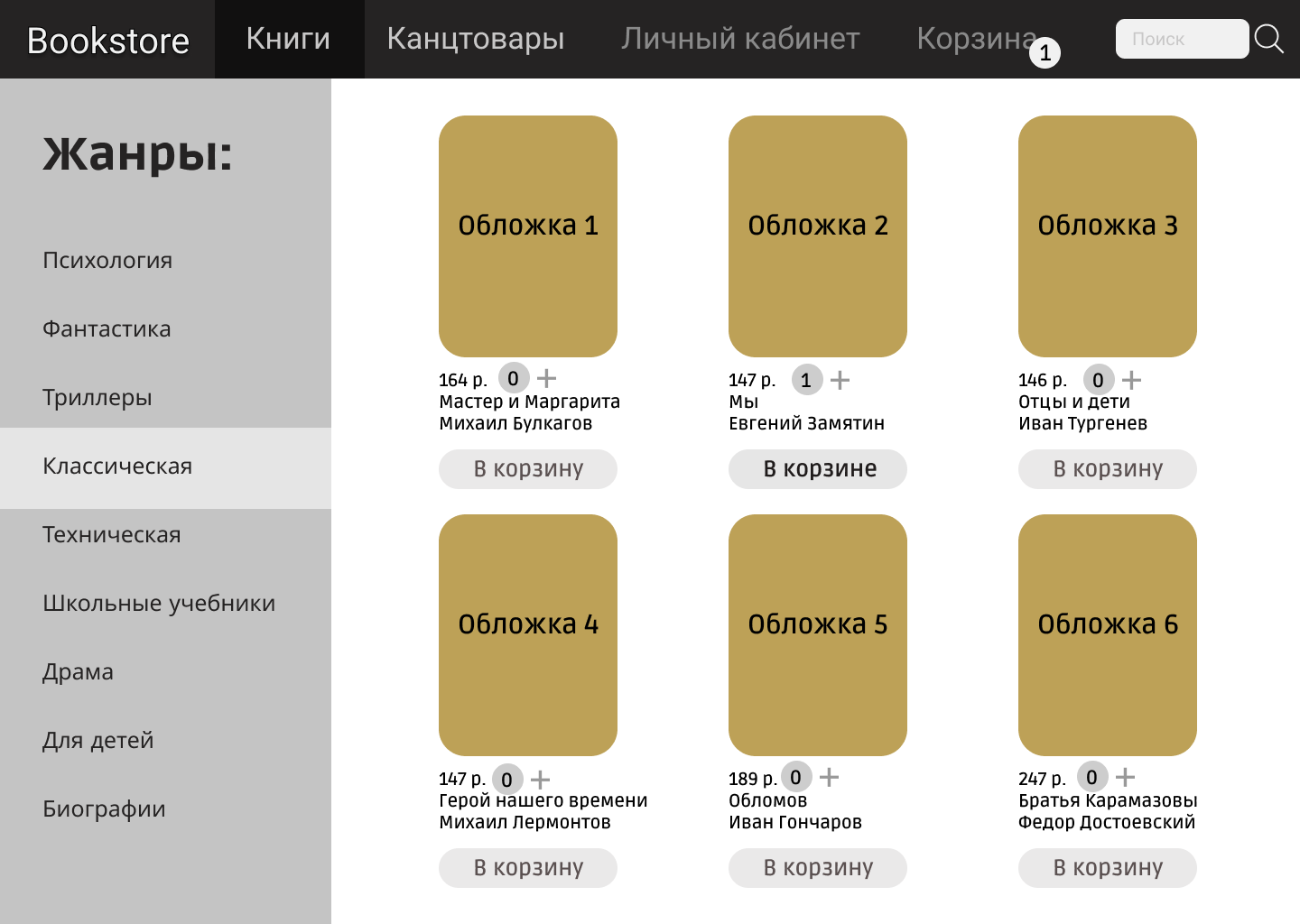


Рисунок 10 – Личный кабинет пользователя

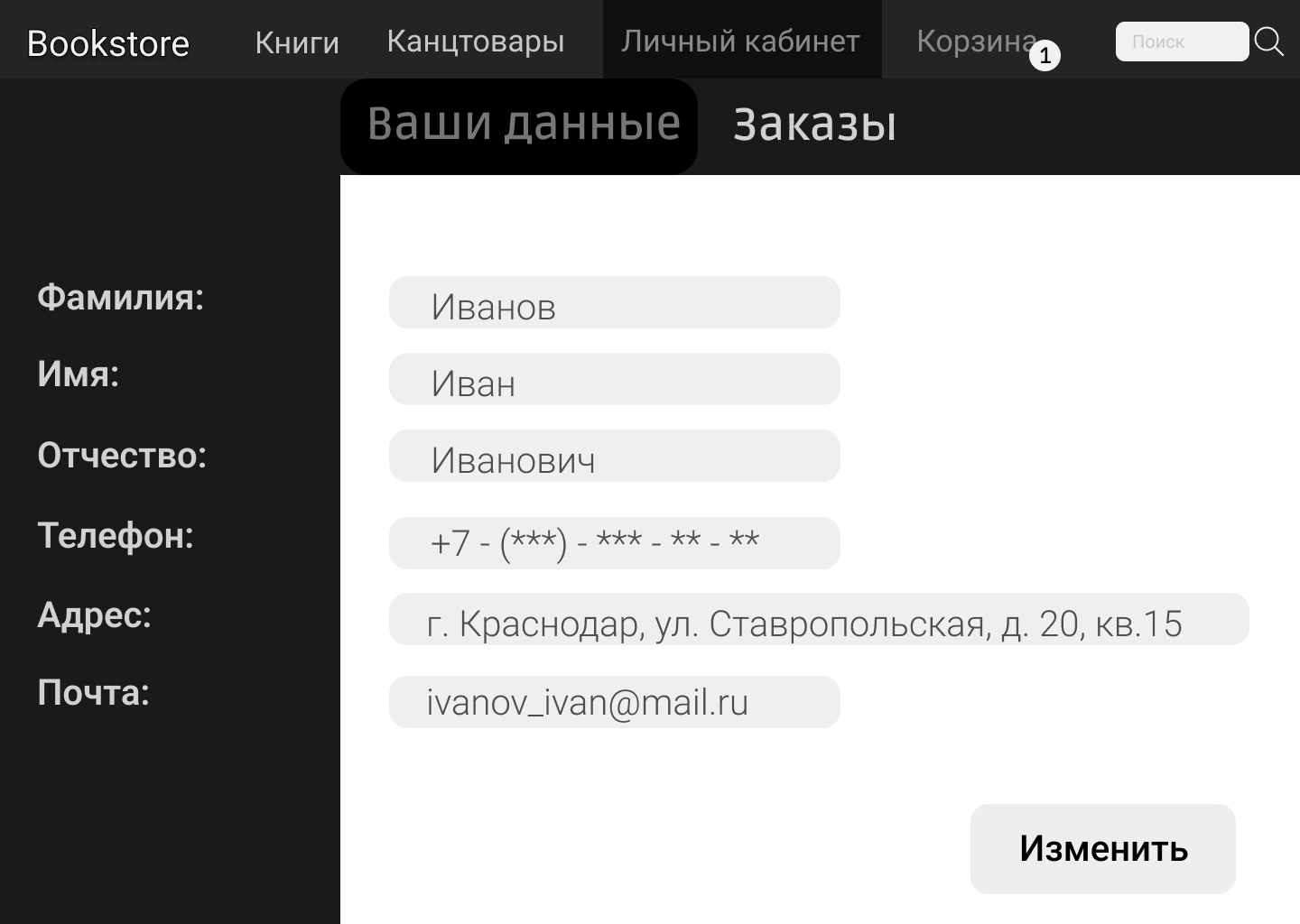


Рисунок 11 – Корзина товаров

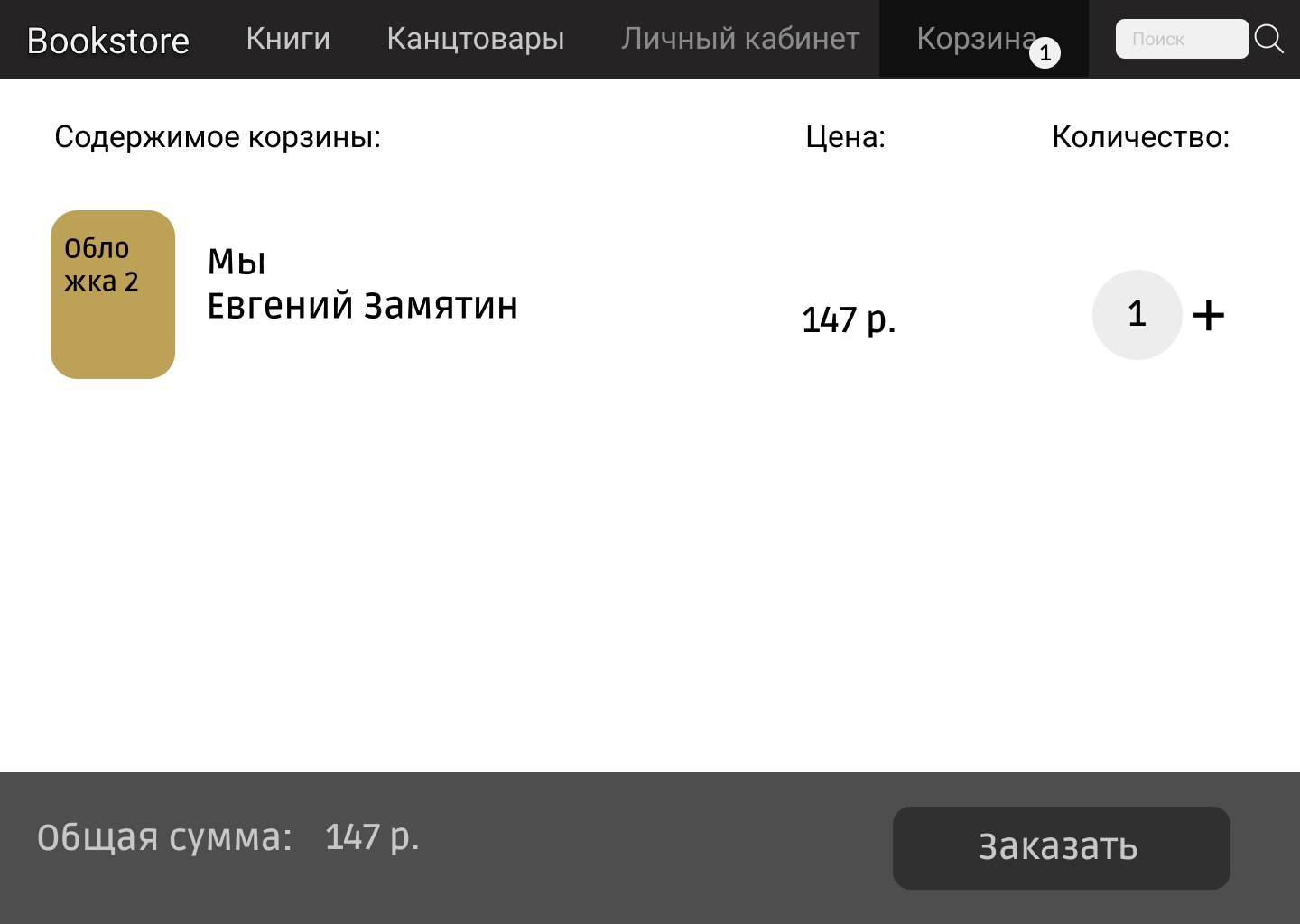


Рисунок 12 – Оформление заказа

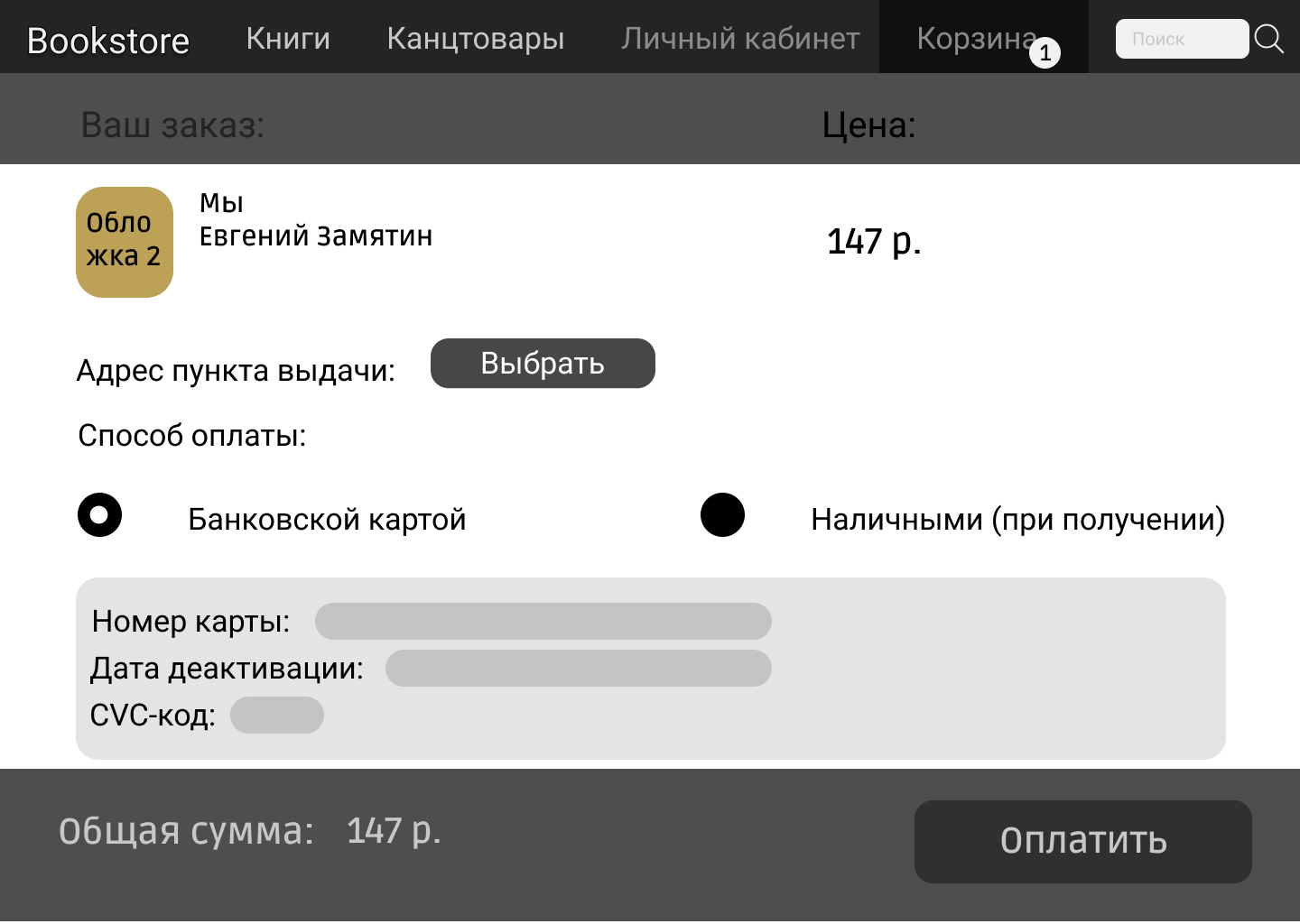


Рисунок 13 – Список заказов

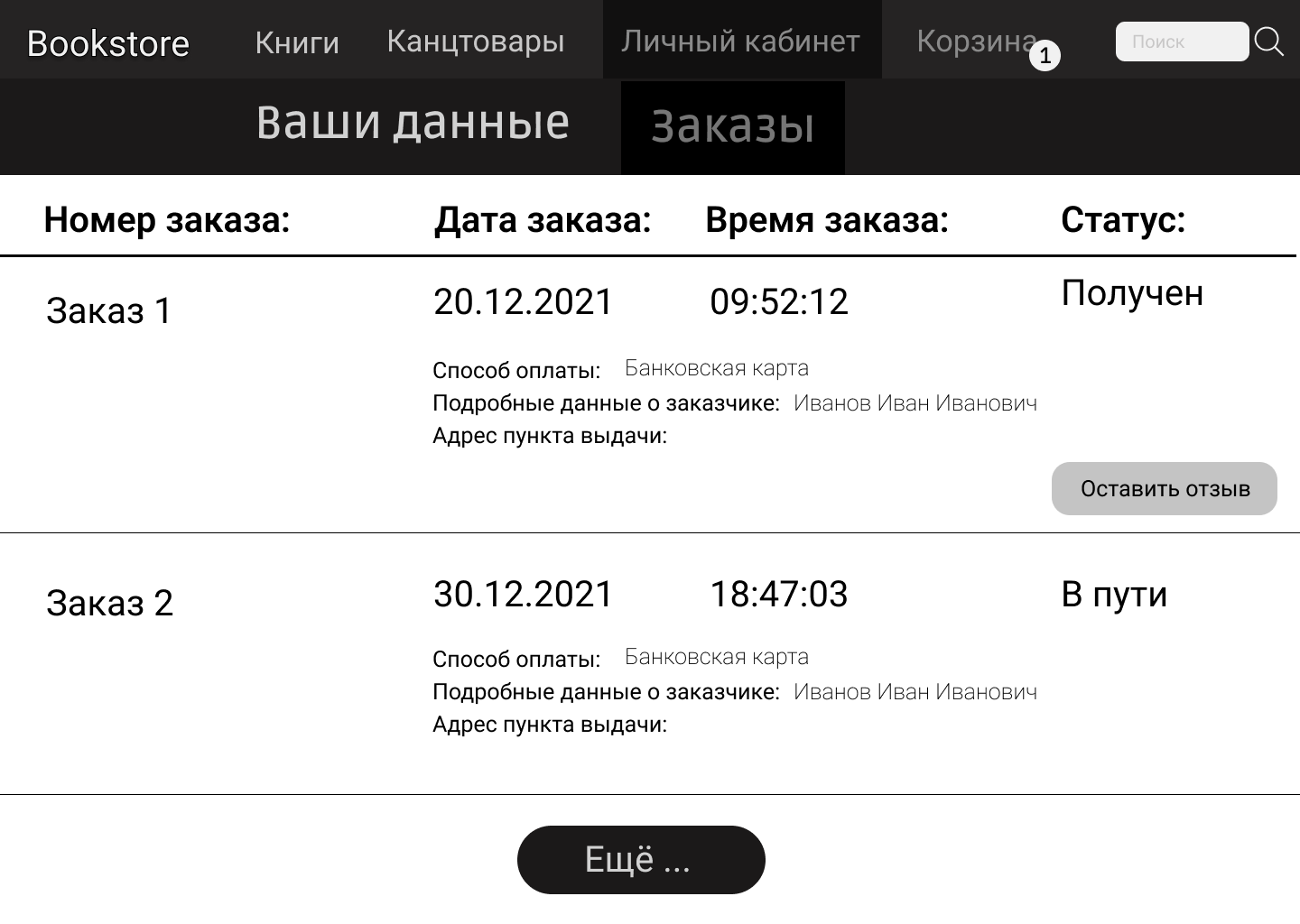


Рисунок 14 – Информация о заказе

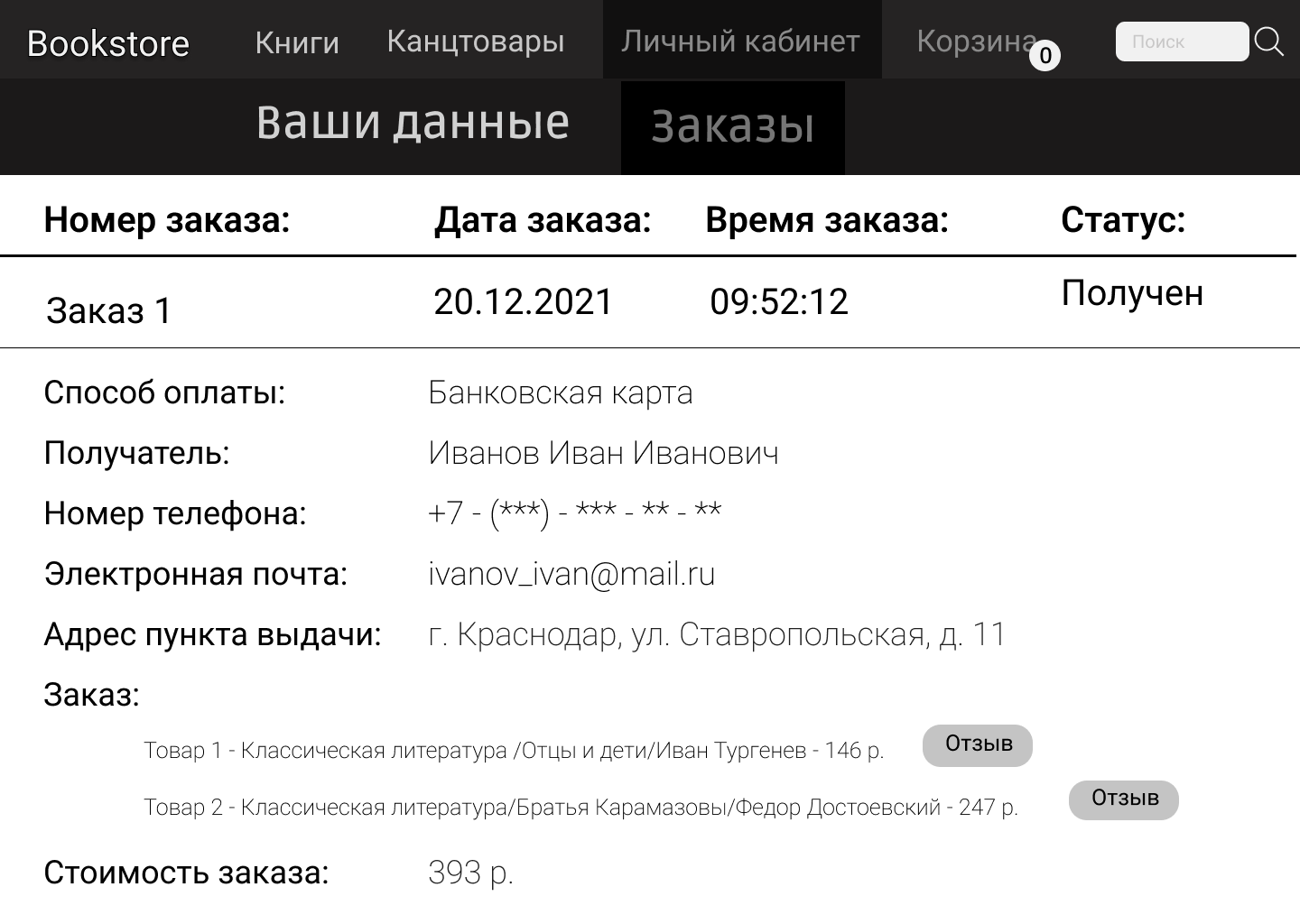


Рисунок 15 – Отзыв к товару

