Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет Электроники и вычислительной техники\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направление (специальность) \_\_\_Информатика и вычислительная техника\_\_\_\_  
Кафедра \_\_\_\_\_\_\_\_\_САПР и ПК\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дисциплина \_\_\_\_\_\_\_Базы данных\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | Утверждаю  Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | «\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20 \_\_\_ г. |

**ЗАДАНИЕ**

**на курсовую работу (проект)**

Студент\_\_\_\_Куканов Александр Дмитриевич\_\_\_\_\_\_\_\_\_\_\_

(фамилия, имя, отчество)

Группа\_\_\_ИВТ-263\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Тема: \_Проектирование и разработка базы данных сети магазинов комиксов и приложения для взаимодействия с ней

Утверждена приказом от «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г. № \_\_\_\_\_\_\_\_\_

2. Срок представления работы (проекта) к защите «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г.

3. Содержание расчетно-пояснительной записки: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Перечень графического материала: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Дата выдачи задания «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20 \_\_\_ г.

Руководитель работы (проекта)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Соколов А.А.\_\_\_

подпись, дата инициалы и фамилия

Задание принял к исполнению  Куканов А.Д.

подпись, дата инициалы и фамилия

Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет Электроники и вычислительной техники\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_САПР и ПК\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе (проекту)**

по дисциплине \_\_\_\_\_\_\_Базы данных\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

на тему Проектирование и разработка базы данных сети магазинов комиксов и приложения для взаимодействия с ней

Студент\_\_\_ Куканов Александр Дмитриевич.

(фамилия, имя, отчество)

Группа\_\_\_ИВТ-263\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель работы (проекта) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Соколов А.А.\_

(подпись и дата подписания) (инициалы и фамилия)

Члены комиссии:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

Нормоконтролер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата подписания) (инициалы и фамилия)

Волгоград 2021 г.

Содержание

[Концептуальное проектирование базы данных 4](#_Toc74524237)

[1. Предметная область 4](#_Toc74524238)

[3. ER-диаграмма 8](#_Toc74524239)

[Логическое проектирование БД 9](#_Toc74524240)

[1. Реляционная модель: 9](#_Toc74524241)

[2. Набор данных 9](#_Toc74524242)

[Физическое проектирование БД 13](#_Toc74524243)

[1. Реляционная схема в MySQL 13](#_Toc74524244)

[2. Реляционная схема (из второй лабораторной работы) 14](#_Toc74524245)

[3. Таблицы 14](#_Toc74524247)

[Реализация SQL-запросов базы данных 18](#_Toc74524248)

[Индексы, хранимые процедуры, представления 24](#_Toc74524249)

[Взаимодействие с БД из приложения 38](#_Toc74524250)

[Ссылка на репозиторий: 42](#_Toc74524251)

[Список литературы 42](#_Toc74524252)

# Концептуальное проектирование базы данных

# Предметная область

Создание базы данных сети магазинов комиксов. БД будут использовать филиалы магазинов в городе. Данная БД поможет упростить работу за счет оптимизации хранения информации о товаре и времени обработки логистики, а также заказов и ассортимента.

1. **Описание предметной области**

Выбранная предметная область – магазин комиксов организация, специализирующаяся на продаже и отправке комиксов.

Точка зрения – администратор магазина.

Сотрудники – директор, администратор, продавец (физ. точка)

1. Функциональные и нефункциональные требования

##### Транзакционные (задачи учёта):

1. Оформить заказ с выбранными позициями
2. Изменить скидку на конкретный комикс
3. Добавить в описание издательства «Закрыто»
4. Добавить новый лэйбл
5. Изменить управляющего магазина

**Справочные (оперативные запросы):**

1. Показать текущие несобранные заказы для данного пользователя
2. Показать выходящие на этой неделе комиксы
3. Показать сотрудника, работающего сегодня
4. Показать наличие конкретного комикса в данном магазине
5. Показать собранные на сегодня заказы с выбранным типом доставки

**Справочные расчётные (аналитические запросы):**

1. Показать статистику (количество) купленных комиксы по данной тематике
2. Показать среднее количество проданных экземпляров данного издательства
3. Показать наиболее популярных авторов
4. Показать статистику (количество) по заказам в разные города
5. Показать статистику (количество поставок) за месяц по предпочитаемым транспортным компаниям для отправки

**Плановые (расчётные задачи):**

1. Составить расписание закупок на месяц
2. Спланировать скидки на ближайший месяц
3. Составить расписание отправок заказов

Типы сущностей

Комикс

* Название (Строка)
* Дэдлайн предзаказа (Дата дд.мм.гг)
* Дата выхода (Дата дд.мм.гг)
* Формат (Строка)
* ID (Строка)
* Описание (Строка)

Жанр

* Название (Строка)
* Описание (Строка)

Художник обложки

* ФИО (Строка)
* ID (Число)

Художник

* ФИО (Строка)
* ID (Число)

Сценарист

* ФИО (Строка)
* ID (Число)

Издательство

* Название (Строка)
* Описание (Строка)

Лэйбл

* Название (Строка)
* Описание (Строка)

Наличие

* Скидка (Число)
* Цена (Число)
* Количество (Число)

Магазин

* Адрес (Строка)
* Номер телефона (Строка +7ХХХХХХХХ)

Сотрудник

* ФИО (Строка)
* Должность (Строка)
* Дни работы (Строка)
* ID (Число)
* Покупатель (Строка)
* ФИО (Строка)
* ID (Строка)
* Адрес доставки (Строка)
* Эл. Почта (Строка)

Доставка

* Тип (Строка)
* Адрес (Строка)
* Стоимость(число)

Заказ

* Состояние (Строка)
* Номер (Строка)
* Дата оформления (Дата дд.мм.гг.)
* Комментарий (Строка)
* Тип («Купить сейчас»/ «Предзаказ»)

Обложка

* Наименование (Строка)
* URL (Строка)
* Комментарий (Строка)

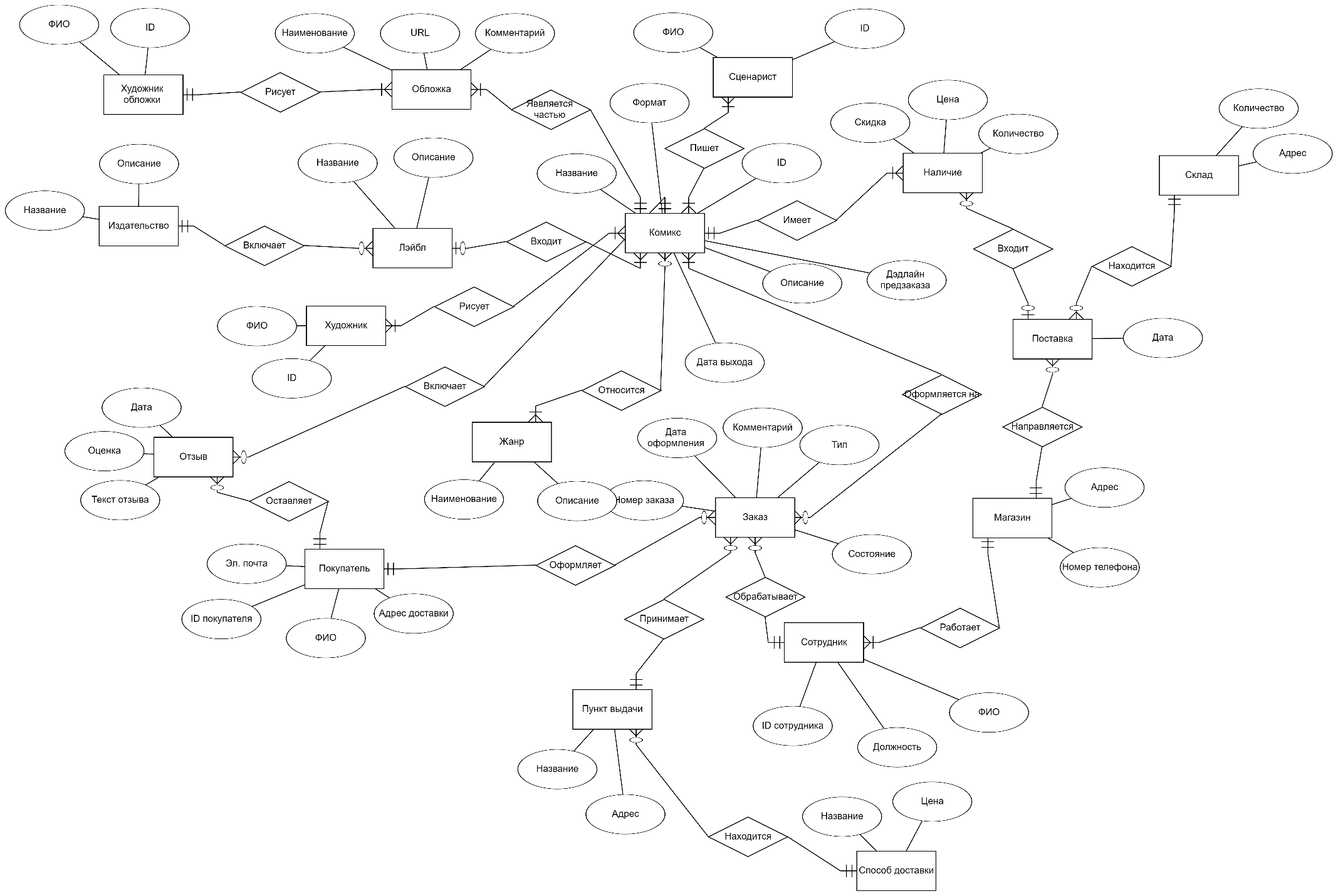
Поставка

* Дата (дата дд.мм.гг.)

Склад

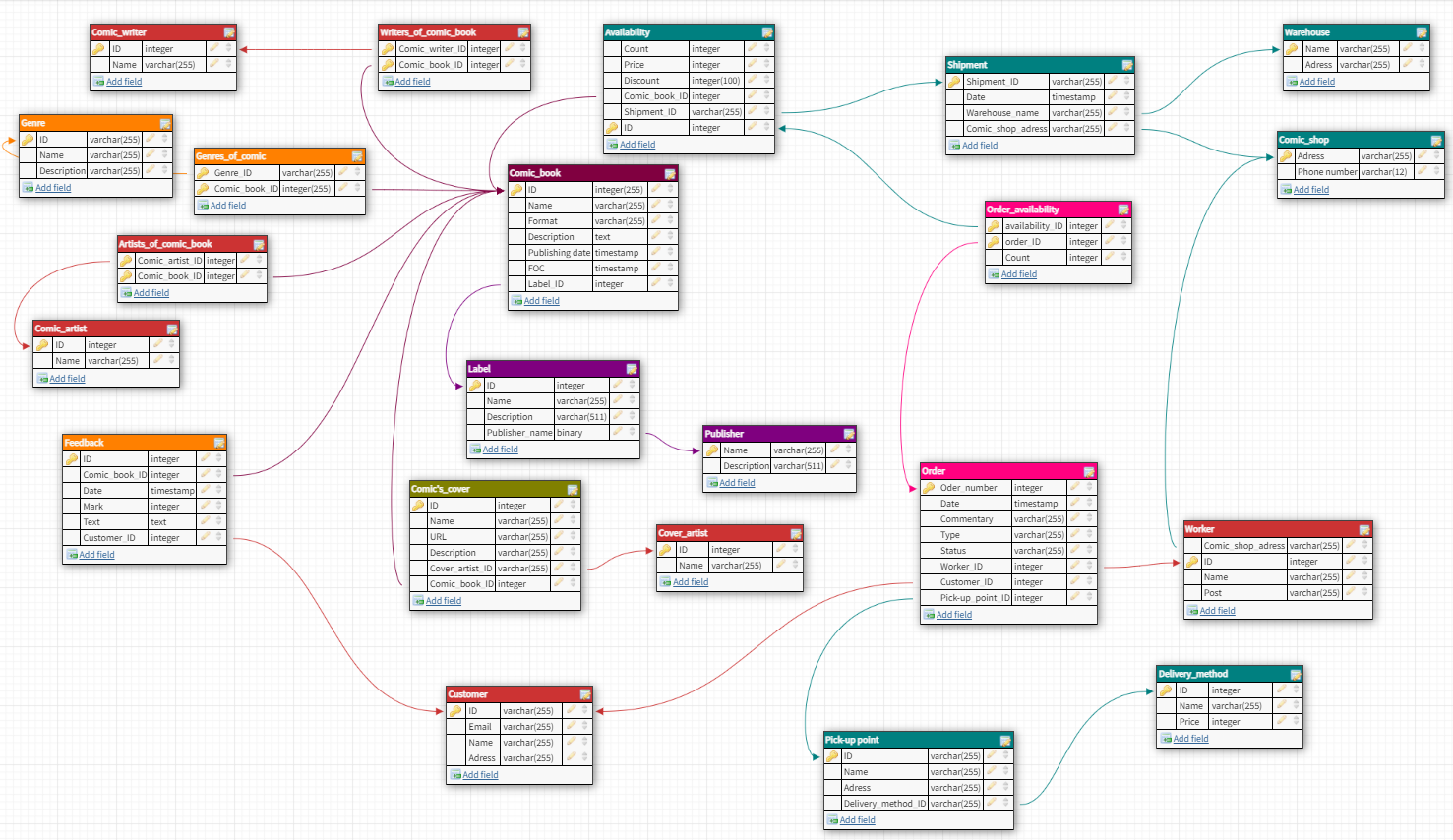
* Количество (Число)
* Адрес (Строка)

# ER-диаграмма



# Логическое проектирование БД

# Реляционная модель:



# Набор данных

Comic book

|  |  |  |
| --- | --- | --- |
| ID | 111 | 112 |
| Name | Criminal #12 | Blade runner 2019 |
| Format | Single issue | Hardcover |
| Description | Text | Text text |
| Publishing\_date | 24/06/2021 | 31/06/2021 |
| FOC | 20/06/2021 | 27/06/2021 |
| Label\_ID | 1234 | 1233 |

Comic’s cover

|  |  |  |
| --- | --- | --- |
| ID | 888 | 889 |
| Name | Incentive cover 1:20 | Blank cover 1:50 |
| URL | url.com/url.png | url.com/url2.png |
| Description | SDCC exclusive | Daredevil #21 hommage |
| Cover\_artist\_ID | 322 | 323 |
| Comic\_book\_ID | 111 | 112 |

Genres of comic book

|  |  |  |
| --- | --- | --- |
| Genre\_ID | 100 | 102 |
| Comic\_book\_ID | 111 | 112 |

Artists of comic book

|  |  |  |
| --- | --- | --- |
| Comic\_artist\_ID | 322 | 323 |
| Comic\_book\_ID | 111 | 112 |

Writers of comic book

|  |  |  |
| --- | --- | --- |
| Comic\_writer\_ID | 1111 | 1111 |
| Comic\_book\_ID | 111 | 112 |

Cover artist

|  |  |  |
| --- | --- | --- |
| ID | 322 | 323 |
| Name | Chip Kid | Chip Zdarsky |

Comic\_writer

|  |  |  |
| --- | --- | --- |
| ID | 1111 | 1111 |
| Name | Joe Hill | Tom King |

Comic\_artist

|  |  |  |
| --- | --- | --- |
| ID | 322 | 323 |
| Name | Chip Kid | Chip Zdarsky |

Genre

|  |  |  |
| --- | --- | --- |
| ID | 100 | 102 |
| Name | Pulp fiction | Science fiction |
| Description | Text | Double text |

Feedback

|  |  |  |
| --- | --- | --- |
| ID | 2345678 | 2345679 |
| Date | 22/07/2021 | 21/07/2021 |
| Mark | 4 | 4 |
| Text | Didn’t read lol | Didn’t read lol |
| Comic\_book\_ID | 111 | 112 |
| Customer\_ID | 66666 | 66667 |

Availability

|  |  |  |
| --- | --- | --- |
| ID | 992 | 991 |
| Count | 62 | 23 |
| Price | 150 | 4500 |
| Discount | 10 | 30 |
| Comic\_book\_ID | 111 | 112 |
| Shipment\_ID | 555 | 556 |

Shipment

|  |  |  |
| --- | --- | --- |
| Shipment\_ID | 555 | 556 |
| Date | 30/06/2021 | 2/07/2021 |
| Warehouse\_name | Original\_name | Not\_original\_name |
| Comic\_shop\_adress | Original\_adress | Not\_original\_adress |

Warehouse

|  |  |  |
| --- | --- | --- |
| Name | Original\_name | Not\_Original\_name |
| Adress | Original\_adress | Not\_original\_adress |

Comic\_shop

|  |  |  |
| --- | --- | --- |
| Adress | Original\_adress | Not\_original\_adress |
| Phone\_number | 85677892772 | 86785645454 |

Worker

|  |  |  |
| --- | --- | --- |
| ID | 777 | 778 |
| Comic\_shop\_adress | Original\_adress | Not\_original\_adress |
| Name | Даниил Данилов | Михаил Михайлов |
| Post | Admin | Manager |

Order

|  |  |  |
| --- | --- | --- |
| Order\_number | 111111 | 111112 |
| Date | 31/08/2021 | 31/09/2021 |
| Commentary | Free sheepment | Add magnet |
| Type | Buy now | Buy now |
| Status | Assembled | Shipped |
| Worker\_ID | 777 | 778 |
| Customer\_ID | 66666 | 66667 |
| Pick-up\_point\_name | CDEK office name | DPD office name |

Order\_availability

|  |  |  |
| --- | --- | --- |
| Availability\_ID | 992 | 991 |
| Order\_ID | 111111 | 111112 |
| Count | 1 | 1 |

Pick-up\_point

|  |  |  |
| --- | --- | --- |
| ID | 98746512 | 654987132 |
| Name | CDEK office name | DPD office name |
| Adress | CDEK adress | DPD adress |
| Delivery\_method\_ID | CDEK | DPD |

Delivery\_method

|  |  |  |
| --- | --- | --- |
| ID | 4785 | 1452 |
| Name | CDEK | DPD |
| Price | 210 | 330 |

Customer

|  |  |  |
| --- | --- | --- |
| ID | 66666 | 66667 |
| Email | [customerEmail1@mail.com](mailto:customerEmail1@mail.com) | Customer2Email2@mail.com |
| Name | Ivanov Ivan | Customer Name2 |
| Adress | Customer1\_adress | Customer2\_adress |

Label

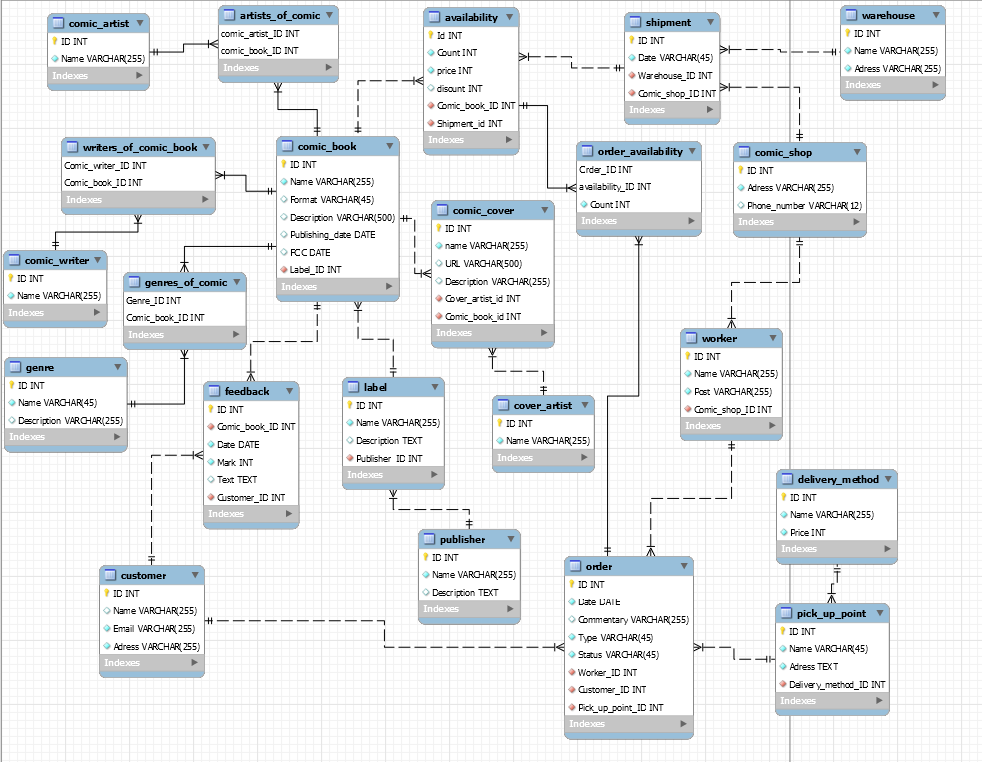
|  |  |  |
| --- | --- | --- |
| ID | 1234 | 1233 |
| Name | Label1 | Label2 |
| Description | Text1 | Text2 |
| Publisher\_name | DC comics | Marvel |

Publisher

|  |  |  |
| --- | --- | --- |
| Name | DC comics | Marvel |
| Description | Publisher1\_description | Publisher2\_description |

# Физическое проектирование БД

# Реляционная схема в MySQL

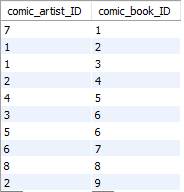


# Реляционная схема (из второй лабораторной работы)

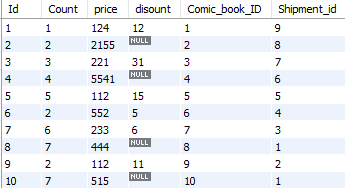
# 

# 3. Таблицы

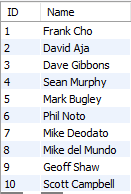
1) Artists of comic books (художники комиксов, многие ко многим)



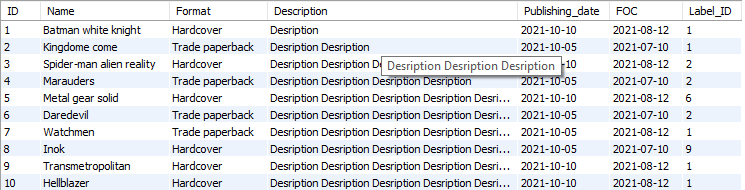
2) Availability



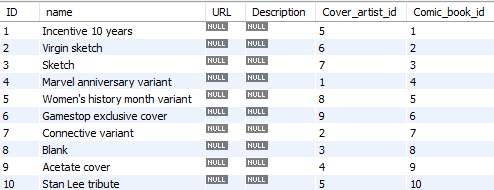
3) Comic artist



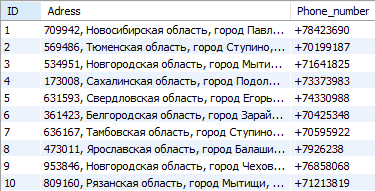
4) Comic book



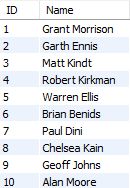
5) Comic cover



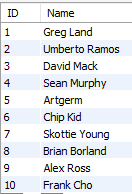
6) Comic shop



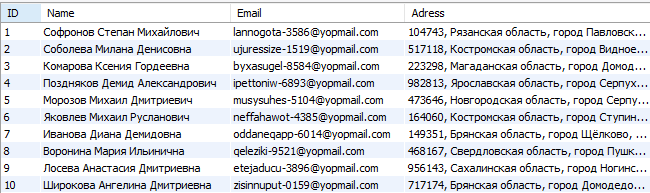
7) Comic writer



8) Cover artist



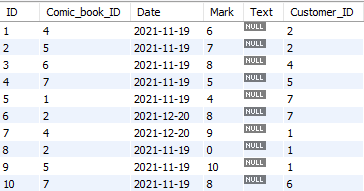
9) Customer



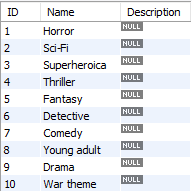
10) Delivery method



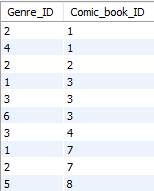
11) Feedback



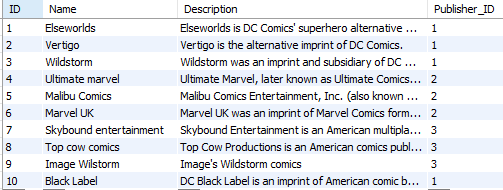
12) Genre



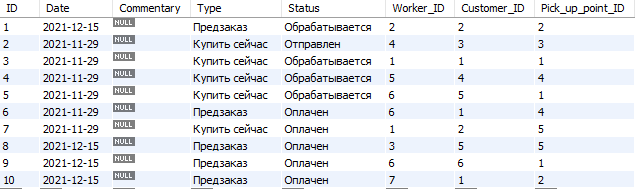
13) Genres of comics (Жанры комиксов, многие ко многим)



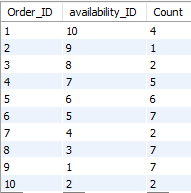
14) Label



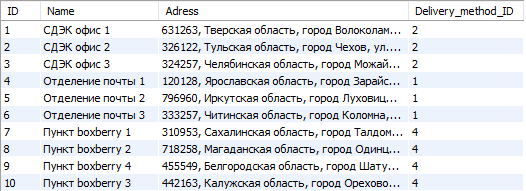
15) Order



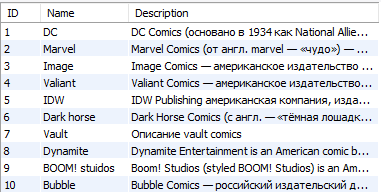
16) Order availability (Заказы и наличие, многие ко многим)



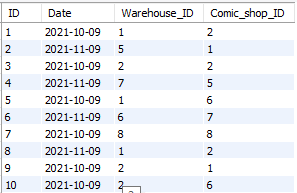
17) Pick up point



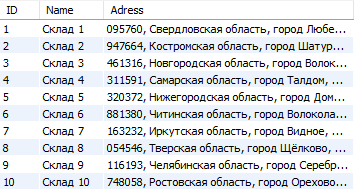
18) Publisher



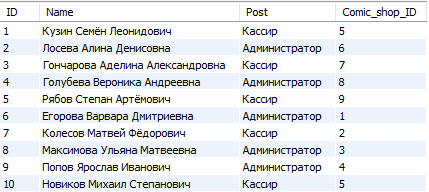
19) Shipment



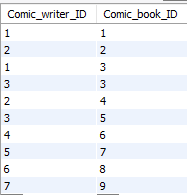
20) Warehouse



21) Worker



22) Writers of comic books (Писатели комиксов, многие ко многим)



# Реализация SQL-запросов базы данных

UPDATE:

UPDATE `comic\_book` SET `format` = 'tpb' WHERE `format` = 'Trade paperback';

UPDATE `availability` SET `discount` = '35' WHERE `price` > 1000;

UPDATE `order` SET `Status` = 'Отправлен' WHERE `type` = 'Предзаказ';

UPDATE `feedback` SET `mark` = 7 WHERE `mark` < 5;

UPDATE `worker` SET `post` = 'Уволен' WHERE `comic\_shop\_id` = 5;

UPDATE `order` SET `status` = 'Доставлен' WHERE `status` = 'Отправлен';

UPDATE `label` SET `description` = '' WHERE `publisher\_id` = 5;

DELETE:

DELETE FROM `worker` WHERE `Comic\_shop\_ID` = 4;

DELETE FROM `delivery\_method` WHERE `price` = 0;

DELETE FROM `feedback` WHERE `mark` < 8;

DELETE FROM `Comic\_shop` WHERE `adress` LIKE 'Вологодская область';

DELETE FROM `availability` WHERE `price` between 200 and 250;

DELETE FROM `worker` WHERE `post` = 'Уволен';

DELETE FROM `warehouse` WHERE `adress` LIKE 'Пермская область';

SELECT:

SELECT `id` FROM `comic\_book` WHERE `FOC`> '2021-08-08' ;

SELECT `id` FROM `order` WHERE `date` BETWEEN '2021-12-14' AND '2021-12-16';

SELECT `name` FROM `comic\_book` WHERE `FOC` BETWEEN '2021-08-09' AND '2021-08-13';

SELECT `comic\_book\_id`, `text` from `feedback` WHERE `mark` > 8;

SELECT `name` FROM `label` WHERE `publisher\_id` = 1;

SELECT `name` FROM `label` WHERE `publisher\_id` = 2;

SELECT `id`,`price`,`shipment\_id` FROM `availability` WHERE `price` > 1000;

SELECT `Comic\_Book\_Id`, `mark` FROM `feedback` WHERE `customer\_id` = 1;

SELECT `name` FROM `genre` WHERE `Description` IS NULL;

SELECT `name`, `FOC` FROM `comic\_book` WHERE `FOC` BETWEEN '2021-06-01' AND '2021-09-01' AND `label\_id` = 2;

SELECT `name` FROM `comic\_book` WHERE `Label\_ID` NOT IN (1,2);

SELECT `name`, `price` FROM `delivery\_method` WHERE `price` BETWEEN '500' AND '1000';

SELECT `order\_id`, `count` FROM `order\_availability` WHERE `count` > 5 AND `availability\_id` = 4;

SELECT `comic\_book\_id` FROM `genres\_of\_comic` WHERE `genre\_id` = 1;

SELECT `name`,`description` FROM `Comic\_book` WHERE `publishing\_date` > '2021-12-01' and `format` = 'hardcover';

SELECT `id`,`shipment\_id` FROM `availability` WHERE `price` < 1000 AND `discount` > '15';

SELECT `name` FROM `worker` WHERE `Comic\_shop\_ID` = 2 and `post` = 'Кассир';

SELECT `date` FROM `shipment` WHERE `Comic\_shop\_ID` = 2 AND `warehouse\_id` = 1;

SELECT `text` FROM `feedback` WHERE `comic\_book\_ID` = 6 AND `mark` < 6;

SELECT `name`, `URL` FROM `comic\_cover` WHERE `comic\_book\_id` = 1;

SELECT `comic\_book\_id` FROM `artists\_of\_comic` WHERE `comic\_artist\_id` = 1;

SELECT `name`,`price1` FROM `delivery\_method` WHERE `price` < 300;

SELECT CB.`name`, count FROM `availability` AV INNER JOIN `comic\_book` CB ON CB.`id` = AV.`comic\_book\_id` WHERE `count` > 5;

SELECT `id`,`date` FROM `order` WHERE `date` BETWEEN 2021-10-01 AND 2021-12-01;

SELECT O.`id`, `status` FROM `order` O INNER JOIN `customer` C ON C.`id` = O.`customer\_id` WHERE `customer\_id` = 1;

LIKE:

SELECT `name` FROM `comic\_book` WHERE `description` LIKE '%Batman%';

SELECT `name` FROM `warehouse` WHERE `adress` LIKE '%Ростовская область%';

SELECT `name` FROM `label` WHERE `Description` LIKE '%american%' or '%American%';

SELECT `name` FROM `customer` WHERE `adress` LIKE '%Костромская%';

SELECT `name` FROM `pick\_up\_point` WHERE `adress` LIKE '%Тульская%';

SELECT `phone\_number`,`adress` FROM `comic\_shop` WHERE `adress` LIKE '%Ступино%';

SELECT `name`, `description` FROM `label` WHERE `description` LIKE '%Image%';

create table `CustomerBackup` ( id int NOT NULL AUTO\_INCREMENT,

`Name` varchar(512),

`adress` varchar(512),

`email` varchar(255),

PRIMARY KEY(id)

);

INSERT SELECT:

INSERT INTO `customerbackup` SELECT \* FROM `Customer` WHERE `adress` LIKE '%Костромская%';

INSERT INTO `customerbackup` SELECT \* FROM `Customer` WHERE `adress` LIKE '%Ногинск%';

INSERT INTO `cover\_artist`(`name`) SELECT `name` FROM `comic\_artist`;

DELETE FROM `customerbackup`;

JOIN:

SELECT \* FROM `comic\_artist` CROSS JOIN `comic\_writer` USING (id);

SELECT A.`name` as `Comic book`, B.`name` as `label` FROM `comic\_book` A LEFT JOIN `label`B ON a.`label\_id` = b.`id`;

SELECT `order`.`date`, `worker`.`name` FROM `worker` JOIN `order` ON `worker.id` = `order`.`worker\_id` WHERE `order`.`date` > '2021-12-01';

SELECT `comic\_book`.`name`, `label`.`name`, `publisher`.`name` FROM `comic\_book` INNER JOIN `label` ON `label\_id` = `label`.`id` INNER JOIN `publisher` ON `publisher`.`id` = `publisher\_id`;

SELECT C.`name` as `Comic book`, CA.`name` as `Cover artist` FROM `Comic\_Book` C INNER JOIN `artists\_of\_comic` AC on C.`id` = AC.`comic\_book\_id` INNER JOIN `comic\_artist` CA ON AC.`comic\_artist\_id` = CA.`id`;

SELECT C.`name`, CW.`name` FROM `Comic\_Book` C INNER JOIN `writers\_of\_comic\_book` WC on C.`id` = WC.`comic\_book\_id` INNER JOIN `comic\_writer` CW ON WC.`comic\_writer\_id` = CW.`id`;

SELECT C.`name`as `comic book`,G.`name` as `genre` FROM `Comic\_book` C INNER JOIN `genres\_of\_comic` GC ON C.`id` = GC.`Comic\_book\_id` INNER JOIN `genre` G ON GC.`genre\_id` = G.`id`;

SELECT PUP.`name`,PUP.`adress`, DM.`price` FROM `pick\_up\_point` PUP LEFT JOIN `delivery\_method` DM ON PUP.`delivery\_method\_id` = DM.`id`;

SELECT S.`id`, S.`date`, S.`comic\_shop\_id`, W.`name` FROM `shipment` S NATURAL JOIN `warehouse` W;

SELECT C.`name` as `Customer`, CB.`name` as `Comic book`, F.`text` as `review`, F.`mark` FROM `customer` C INNER JOIN `feedback` F ON C.`id` = F.`customer\_id` INNER JOIN `comic\_book` CB ON F.`comic\_book\_id` = CB.`id`;

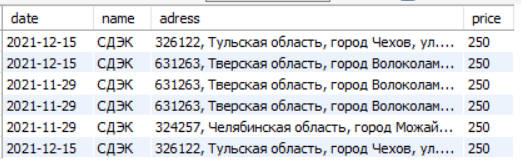


SELECT CB.`name` as `Comic book`, CC.`name` as `Cover name` FROM `comic\_book` CB INNER JOIN `comic\_cover` CC ON CB.`id` = CC.`comic\_book\_id` AND CC.`name` LIKE '%variant%';

SELECT CB.`name`, AV.`count` ,AV.`price` FROM `comic\_book` CB INNER JOIN `availability` AV ON CB.`id` = AV.`comic\_book\_id`;

SELECT CCv.`name` as `Comic cover`, CVA.`name` as `Cover artist` FROM `comic\_cover` CCv INNER JOIN `cover\_artist` CVA ON CCv.`cover\_artist\_id` = CVA.`id`;

SELECT O.`date`, DM.`name`, PUP.`adress`, DM.`price` FROM `order` O INNER JOIN `pick\_up\_point` PUP ON O.`picK\_up\_point\_id` = PUP.`id` INNER JOIN `delivery\_method` DM ON PUP.`delivery\_method\_id`= DM.`id` AND price < 500;



SELECT OA.`order\_id`,CB.`name`, AV.`price`, OA.`count` FROM `order\_availability` OA INNER JOIN `availability` AV ON OA.`availability\_id` = AV.`id` AND oa.`count` > 2 INNER JOIN `Comic\_book` CB ON AV.`comic\_book\_id` = CB.`id`;

SELECT `name` FROM `cover\_artist` NATURAL JOIN `comic\_artist`;

SELECT CS.`adress`, CS.`phone\_number`, W.`name` FROM `comic\_shop` CS INNER JOIN `worker` W ON CS.`id` = W.`comic\_shop\_id`;

GROUP BY:

SELECT `comic\_book\_id`, AVG(mark) from `feedback` GROUP BY `comic\_book\_id`;

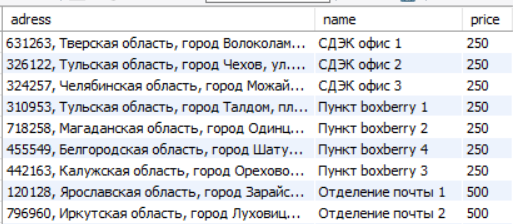
SELECT `shipment\_id`, SUM(price) FROM `availability` GROUP BY `shipment\_id`;

SELECT `comic\_book\_id`, MIN(mark) from `feedback` GROUP BY `comic\_book\_id` order by MIN(mark);

SELECT O.`date`, SUM(A.price \* OA.count) FROM `availability` A INNER JOIN `order\_availability` OA ON A.`id` = OA.`availability\_ID` INNER JOIN `order` O ON OA.`order\_id` = O.`id` group by date;

SELECT `id`, price \* count FROM `availability` order by price \* count;

SELECT `adress`, pick\_up\_point.`name` , `price` FROM `delivery\_method` INNER JOIN `pick\_up\_point` ON pick\_up\_point.delivery\_method\_id = delivery\_method.id ORDER BY `price`;



SELECT `pick\_up\_point`.`name`, MIN(`price`) FROM `delivery\_method` INNER JOIN `pick\_up\_point` ON `pick\_up\_point`.`delivery\_method\_id` = `pick\_up\_point`.`id` GROUP BY `delivery\_method\_id`;

SELECT `comic\_book\_id`, max(mark) from `feedback` GROUP BY `comic\_book\_id` order by MIN(mark);

SELECT `name`, COUNT(id) FROM `pick\_up\_point` GROUP BY `delivery\_method\_id` ORDER BY COUNT(id) DESC;

SELECT `publisher`.`name`, COUNT(`label`.id) as `количество` FROM `label` INNER JOIN `publisher` ON `publisher`.`id` = `label`.`publisher\_id` GROUP BY `publisher\_id` ORDER BY COUNT(ID);

SELECT `post`, count(id) FROM `worker` GROUP BY `post` ORDER BY count(id) DESC;

SELECT CB.`name` as `comic book`, count(`comic\_book\_id`) FROM `feedback` FB INNER JOIN `comic\_book` CB ON FB.`comic\_Book\_id` = CB.`id` GROUP BY `comic\_Book\_id`;

SELECT `label`.`name`, count(`comic\_book`.`id`) as `Кол-во` FROM `comic\_book` INNER JOIN `label` ON `label`.`id` = `comic\_book`.`label\_id` GROUP BY (label\_id);

SELECT `comic\_book`.`name`, count(comic\_book\_id) as `кол-во` FROM `availability` INNER JOIN `comic\_book` ON `availability`.`comic\_Book\_id` = `comic\_book`.`id` GROUP BY (comic\_book\_id);

SELECT `type`, count(id) as `кол-во` FROM `order` GROUP BY `type`;

SELECT `format`, count(id) as `кол-во` FROM `comic\_book` GROUP BY `format` ORDER BY count(id) DESC;

SELECT PUP.`name`, avg(price) as `price` FROM `pick\_up\_point` PUP INNER JOIN `delivery\_method` DM ON PUP.`delivery\_method\_id` = DM.`id` GROUP BY `delivery\_method\_id`;

UNION:

SELECT \* FROM `customer` union SELECT \* FROM `customerbackup`;

SELECT `name` FROM `comic\_writer` UNION SELECT `name` FROM `comic\_artist`;

SELECT `adress` FROM `comic\_shop` WHERE `adress` LIKE '%Тамбовская область%' UNION SELECT `adress` FROM `WAREHOUSE` WHERE `adress` LIKE '%Тамбовская область%';

SELECT `name` FROM `comic\_artist` UNION SELECT `name` FROM `cover\_artist` ORDER BY `name`;

SELECT ANY, ALL

SELECT `comic\_book`.`name` as `Comic book`, `feedback`.`text` FROM FEEDBACK INNER JOIN `comic\_book` ON `comic\_book`.`id` = `feedback`.`comic\_book\_id` WHERE `text` = ANY(SELECT `text` FROM `feedback`);

SELECT `name`, `price` FROM `delivery\_method` WHERE `price` > ANY ( SELECT `price` FROM `delivery\_method`);

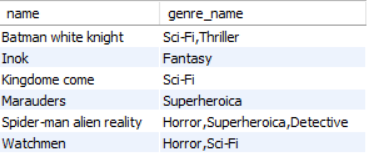
SELECT `name`,`publishing\_date` from `Comic\_Book` WHERE `FOC` > ANY (SELECT `date` FROM `Order`);

GROUP\_CONCAT:

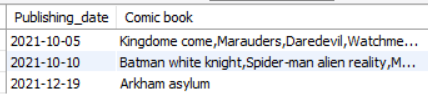
SELECT `comic\_book\_id`, GROUP\_CONCAT(`comic\_artist\_id`) as `comic\_artists` FROM `artists\_of\_comic` GROUP BY `comic\_book\_id`;

SELECT DM.`name`,DM.`price`, group\_concat(PUP.`name`) as `pick\_up\_points` FROM `delivery\_method` DM INNER JOIN `pick\_up\_point` PUP ON DM.`id` = PUP.`delivery\_method\_id` GROUP BY DM.`name`;

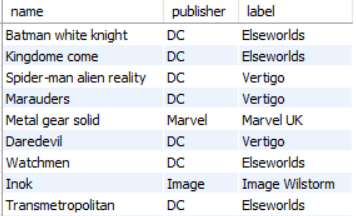
SELECT CB.`name`, group\_concat(G.`name`) as `genre\_name` FROM `comic\_book` CB INNER JOIN `genres\_of\_comic` GC ON GC.`comic\_book\_id` = CB.`id` INNER JOIN `genre` G ON G.`id` = GC.`genre\_id` GROUP BY CB.`name`;



SELECT `Publishing\_date`, Group\_concat(`name`) as `Comic book` FROM `comic\_book` GROUP BY `publishing\_date`;



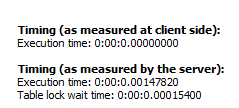
SELECT CB.`name`, P.`name` as `publisher`, L.`name` as `label` FROM `comic\_book` CB INNER JOIN `label` L ON L.`id` = CB.`label\_id` INNER JOIN `publisher` P ON P.`id` = L.`publisher\_id`;

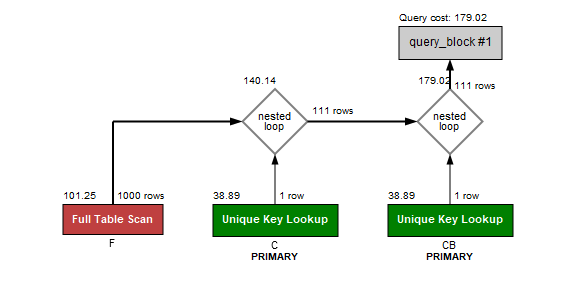


# Индексы, хранимые процедуры, представления

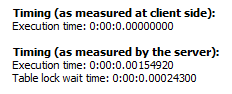
1. SELECT C.`name` as `Customer`, CB.`name` as `Comic book`, F.`text` as `review`, F.`mark` FROM `customer` C   
INNER JOIN `feedback` F ON C.`id` = F.`customer\_id`   
INNER JOIN `comic\_book` CB ON F.`comic\_book\_id` = CB.`id`  
WHERE F.`date` between '2021-06-06' AND '2021-07-07';

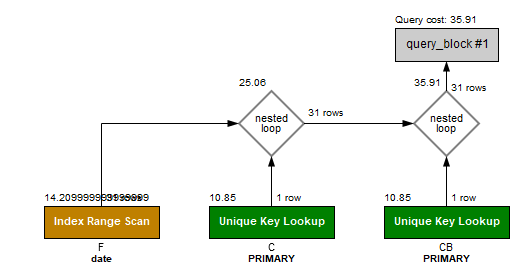
Без индексов:





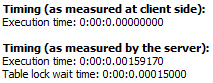
С индексом:

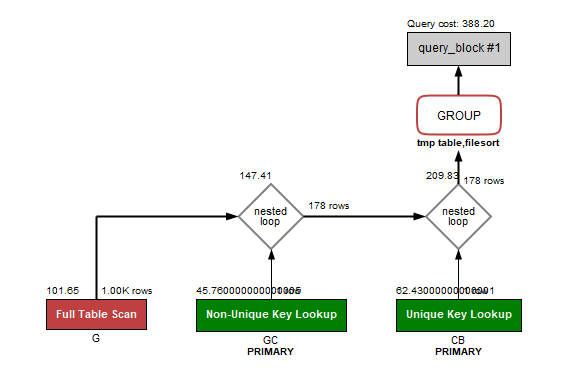




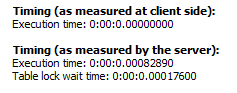
2. SELECT CB.`name`, group\_concat(G.`name`) as `genre\_name` FROM `comic\_book` CB INNER JOIN `genres\_of\_comic` GC ON GC.`comic\_book\_id` = CB.`id` INNER JOIN `genre` G ON G.`id` = GC.`genre\_id` WHERE G.`name` LIKE ('thriller%') GROUP BY CB.`name` ;

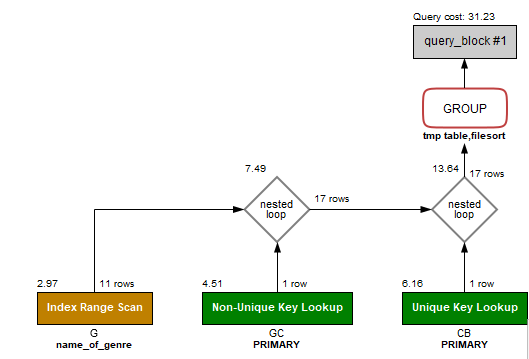
Без индексов:





С индексом:





3. SELECT CB.`name`, CB.`Format`, CB.`description`, CB.`Publishing\_date`, CB.`FOC`, AVG (Mark) as `average mark`,GROUP\_CONCAT( CA.`name`) as `Artist`, GROUP\_CONCAT(CVA.`name`) as `cover artist` , GROUP\_CONCAT(CW.`name`) as `Writer`, L.`name` as `label`, PB.`name` as `publisher` FROM

`feedback` FB JOIN `comic\_book` CB ON FB.`comic\_book\_id` = CB.`id`

JOIN `artists\_of\_comic` AOC ON AOC.`comic\_book\_id` = CB.`id`

JOIN `comic\_artist` CA ON CA.`id` = AOC.`comic\_artist\_id`

JOIN `writers\_of\_comic\_book` WOC ON WOC.`comic\_book\_id` = CB.`id`

JOIN `comic\_writer` CW ON CW.`id` = WOC.`comic\_writer\_id`

JOIN `comic\_cover` CC ON CC.`comic\_book\_id` = CB.`id`

JOIN `cover\_artist` CVA ON CVA.`id` = CC.`cover\_artist\_id`

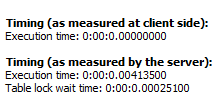
JOIN `label` L ON L.`id` = CB.`label\_id`

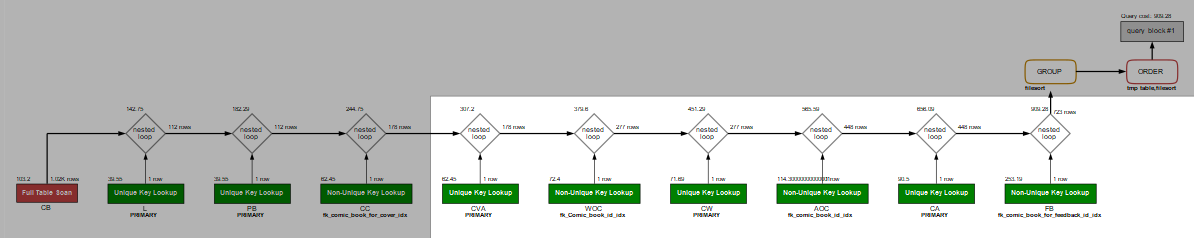
JOIN `publisher` PB ON PB.`id` = L.`publisher\_id`

WHERE CB.`publishing\_date` BETWEEN '2021-01-06' AND '2021-11-07'

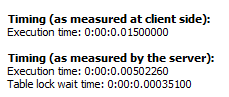
GROUP BY CB.`name` ORDER BY AVG (Mark) DESC LIMIT 5;

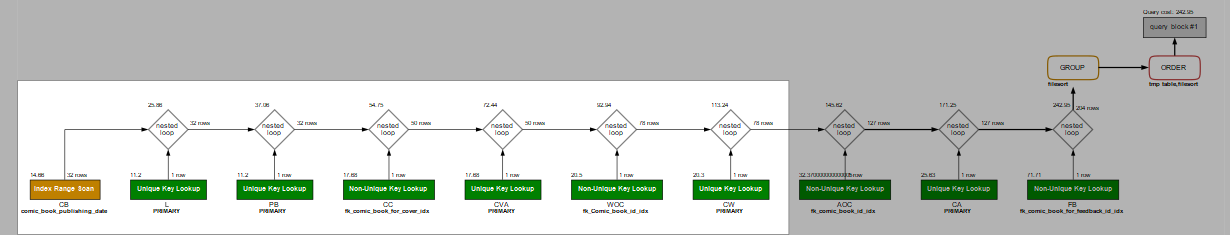
Без индексов:





С индексом:





4. SELECT CS.`adress`, CB.`name`, CB.`Format`, CB.`description`, CB.`Publishing\_date`, CB.`FOC`, L.`name` as `Label`, AV.`count`, AV.`price`, AV.`discount` FROM `comic\_shop` CS

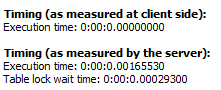
JOIN `shipment` SH ON SH.`comic\_shop\_id` = CS.`id`

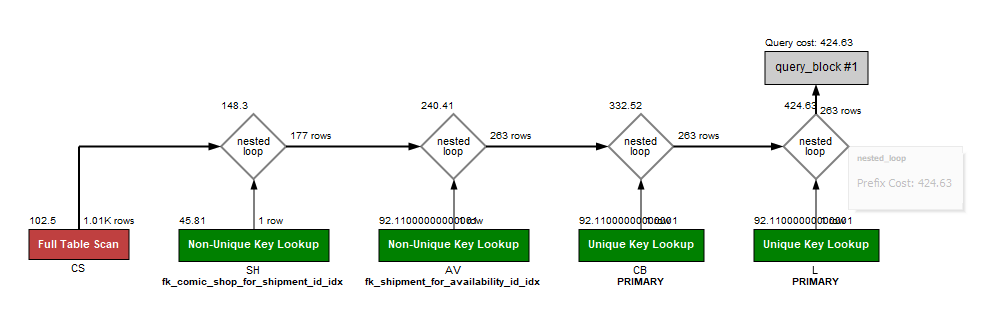
JOIN `availability` AV ON AV.`shipment\_id` = SH.`id`

JOIN `comic\_book` CB ON AV.`comic\_book\_id` = CB.`id`

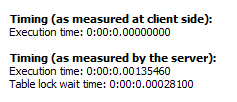
JOIN `label` L ON CB.`label\_id` = L.`id` WHERE CS.`adress` LIKE '709%';

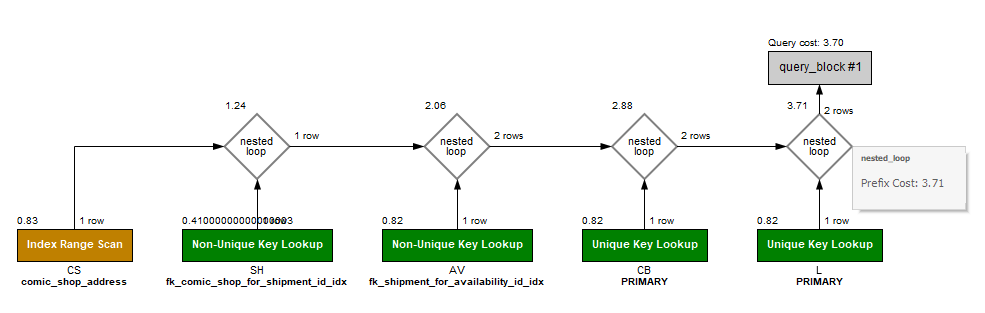
Без индекса:



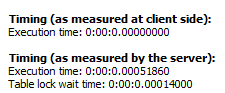


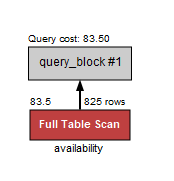
С индексом:



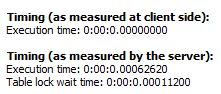


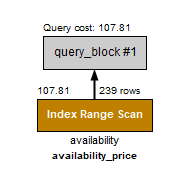
5. SELECT `price`,`shipment\_id` FROM `availability` WHERE `price` > 3500;





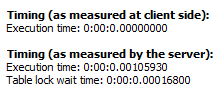
С индексом:

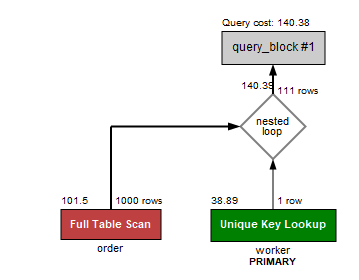




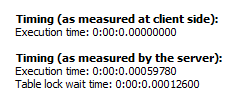
6. SELECT `order`.`date`, `worker`.`name` FROM `worker` JOIN `order` ON `worker`.`id` = `order`.`worker\_id` WHERE `order`.`date` BETWEEN '2021-12-01' AND '2022-02-02';

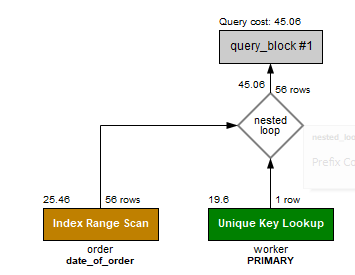
Без индексов:





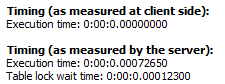
С индексом:

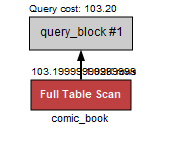




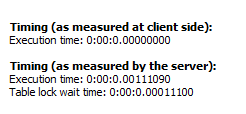
7. SELECT `id` FROM `comic\_book` WHERE `FOC`> '2021-08-08' ;

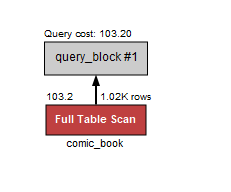
Без индексов:





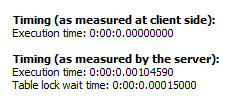
С индексами:

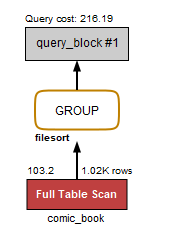




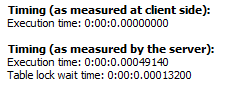
8. SELECT `Publishing\_date`, Group\_concat(`name`) as `Comic book` FROM `comic\_book` GROUP BY `publishing\_date`;

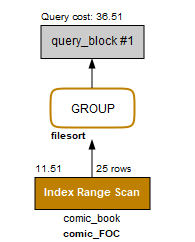
Без индекса:





С индексом:





Добавление индексов:

1 CREATE INDEX `feedback\_date` ON feedback(date);

2 CREATE INDEX `genre\_name` ON genre(name);

3 CREATE INDEX `comic\_book\_publishing\_date` ON comic\_book(publishing\_date);

4 CREATE INDEX `comic\_shop\_address` ON comic\_shop(adress);

5 CREATE INDEX `availability\_price` ON `availability`(price);

6 CREATE INDEX `date\_of\_order` ON `order`(date);

7 CREATE INDEX `comic\_book\_FOC` ON `comic\_book`(`FOC`);

8 CREATE INDEX `comic\_FOC` ON `comic\_book`(`FOC`);

**Процедуры**:

1. Нечеткая оценка комикса по отзывам:

DELIMITER $$

CREATE PROCEDURE `Comics\_mark`()

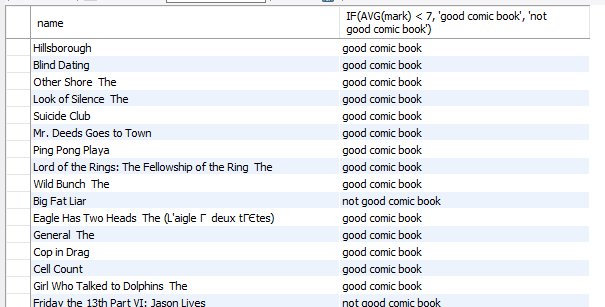
BEGIN

SELECT CB.`name`, IF(AVG(mark) < 7, 'good comic book', 'not good comic book') FROM `feedback` F

INNER JOIN `comic\_book` CB ON F.`comic\_book\_id` = CB.`id` GROUP BY CB.`name`;

END$$

CALL Comics\_mark();



1. Нечеткая оценка стоимости заказов

PROCEDURE `Shipment\_cost`()

BEGIN

SELECT O.`id` as `order\_id`,

CASE

WHEN price\*AV.count = 3500 THEN 'Free shipment'

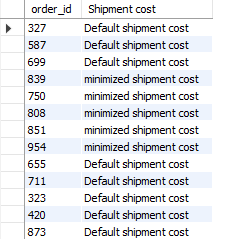
WHEN price\*AV.count < 3500 THEN 'Default shipment cost'

WHEN price\*AV.count > 3500 THEN 'minimized shipment cost'

END AS `Shipment cost`

FROM `order` O INNER JOIN `order\_availability` OA ON O.`id` = OA.`order\_id` INNER JOIN `availability` AV ON OA.`availability\_id` = AV.`id` GROUP BY O.`id`;

END



1. Нечёткая оценка поставок в магазин

PROCEDURE `Shipment\_mark`()

BEGIN

SELECT CS.`adress`,

CASE

WHEN SUM(count) > 500 THEN 'Крупная поставка'

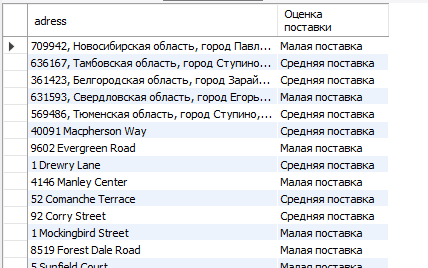
WHEN SUM(count) < 100 THEN 'Малая поставка'

ELSE 'Средняя поставка'

END AS `Оценка поставки`

FROM comic\_shop CS INNER JOIN SHIPMENT SH ON CS.`id` = SH.`comic\_shop\_id` INNER JOIN availability AV ON SH.`id` = AV.`shipment\_id` GROUP BY CS.`adress`;

END



**Функции**:

1. Стоимость комиксов в наличии

FUNCTION `Shipment\_cost`(count int, price int) RETURNS int

DETERMINISTIC

BEGIN

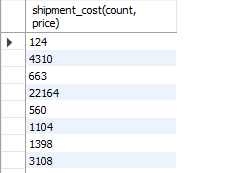
DECLARE scost INT;

SET scost = 0;

SELECT count\*price into scost FROM availability LIMIT 1;

RETURN scost;

END



2)

Нечеткая оценка стоимости методов доставки

FUNCTION `delivery\_high\_price`(`price` int) RETURNS int

DETERMINISTIC

BEGIN

DECLARE count INTEGER;

SET `count` = 0;

IF `price` > 200 THEN

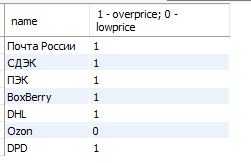
SET `count` = `count` + 1;

END IF;

RETURN `count`;

END

select name, delivery\_high\_price(price) as `1 - overprice; 0 - lowprice` from delivery\_method;



3) Подсчёт крупных поставок

FUNCTION `huge\_shipments`(`price` int, `count` int) RETURNS int

DETERMINISTIC

BEGIN

DECLARE sum\_price INTEGER;

if `count` > 15 THEN

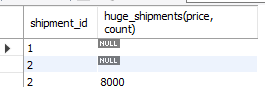
set sum\_price = `price` \* `count`;

END IF;

return sum\_price;

END

select shipment\_id,huge\_shipments(price, count) from availability;



**Представления**

1. Персонал магазина в аризоне

VIEW `comic\_shop\_web`.`arizona\_shop\_staff` AS

SELECT

`w`.`Name` AS `name`, `w`.`Post` AS `post`

FROM

(`comic\_shop\_web`.`worker` `w`

JOIN `comic\_shop\_web`.`comic\_shop` `cs` ON ((`w`.`Comic\_shop\_ID` = `cs`.`ID`)))

WHERE

(`cs`.`Adress` LIKE '%Arizona%')

select \* from arizona\_shop\_staff;

1. Перечень пунктов выдачи

VIEW `comic\_shop\_web`.`points` AS

SELECT

`dm`.`Name` AS `name`, `pup`.`Adress` AS `adress`

FROM

(`comic\_shop\_web`.`pick\_up\_point` `pup`

JOIN `comic\_shop\_web`.`delivery\_method` `dm` ON ((`pup`.`Delivery\_method\_ID` = `dm`.`ID`)))

select\*from points;

1. Комиксы и их издатели

VIEW `comic\_shop\_web`.`comic\_books` AS

SELECT

`cb`.`Name` AS `name`,

`cb`.`Format` AS `format`,

`cb`.`Publishing\_date` AS `publishing\_date`,

`cb`.`FOC` AS `FOC`,

`l`.`Name` AS `label`,

`p`.`Name` AS `publisher`

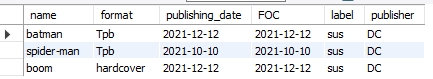
FROM

((`comic\_shop\_web`.`comic\_book` `cb`

JOIN `comic\_shop\_web`.`label` `l` ON ((`cb`.`Label\_ID` = `l`.`ID`)))

JOIN `comic\_shop\_web`.`publisher` `p` ON ((`l`.`Publisher\_ID` = `p`.`ID`)))

select\*from comic\_books;



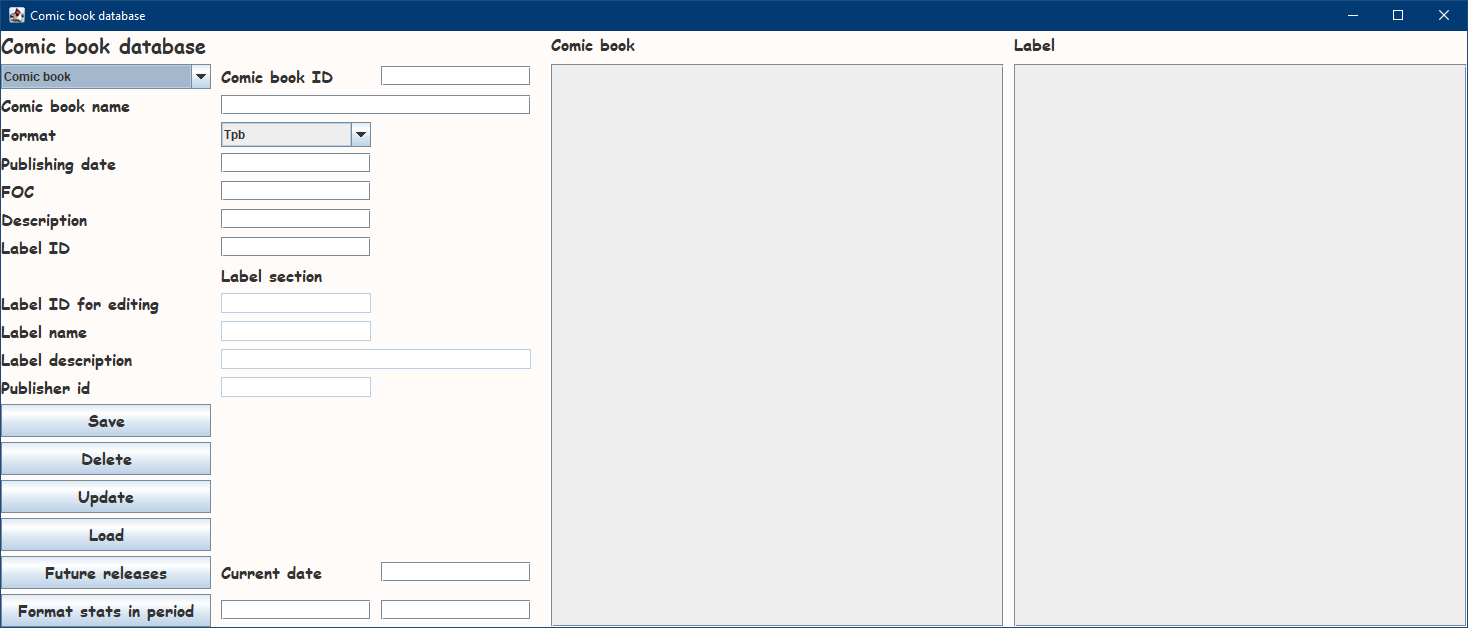
# Взаимодействие с БД из приложения

1. Для реализации лабораторной работы, а также курсовой работы был выбран язык программирования Java.
2. Были изучены библиотеки для работы с выбранной СУБД и на их основе было реализовано простейшее приложение с графическим интерфейсом для подключения к собственной БД в СУБД на выбранном языке программирования и с помощью выбранных библиотек для подключения:

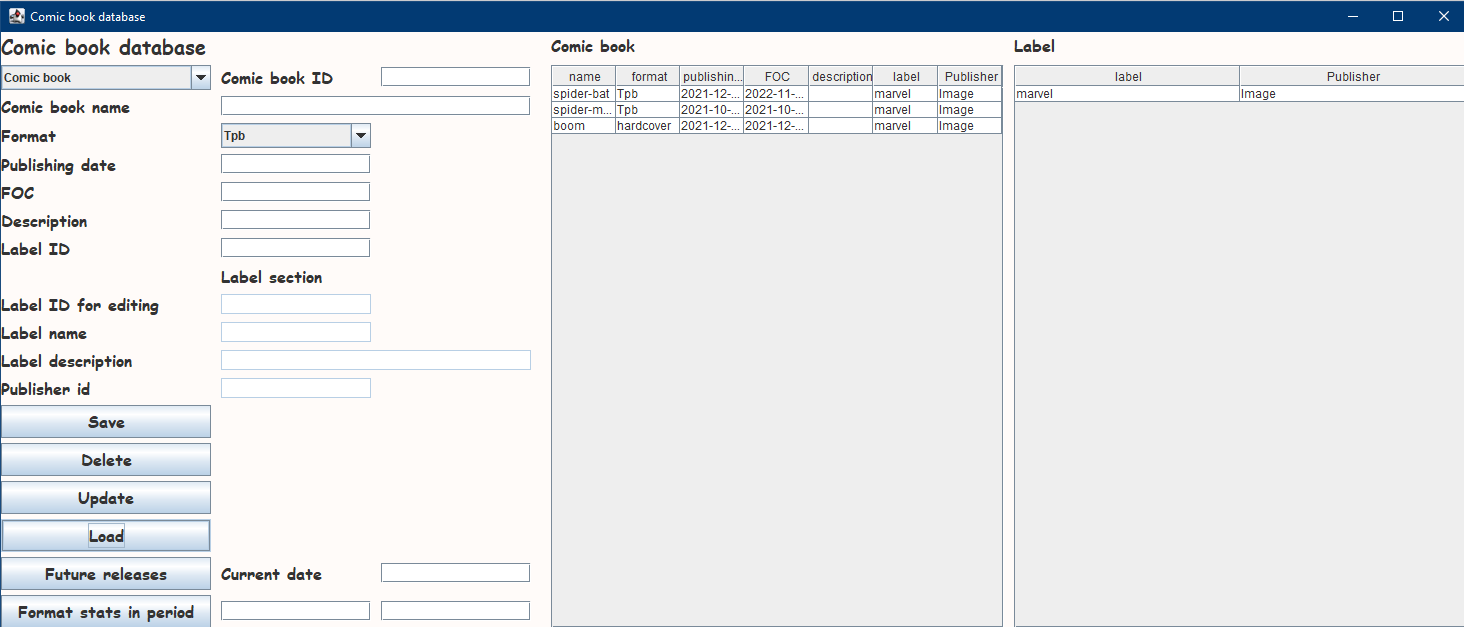
import net.proteanit.sql.DbUtils;  
import javax.swing.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.sql.DriverManager;  
import java.sql.SQLException;  
import java.sql.\*;

1. В качестве интерфейса взаимодействия с пользователем на данной лабораторной работе было реализовано оконное приложение (Windows forms).
2. Исходный код загружен в репозиторий, ссылка на который находится в протоколе.
3. Была реализована CRUD (получение, добавление, редактирование и удаление данных) для 2-х таблиц базы данных, одна из которых связана внешним ключом с другой.
4. Была протестирована логика взаимодействия с БД, с помощью проверки данных в БД можно убедиться, что все функции приложения выполняются корректно. Методы, в которых содержатся запросы к БД, возвращают корректные данные, корректно производится запись, создаются корректные связи между сущностями БД.

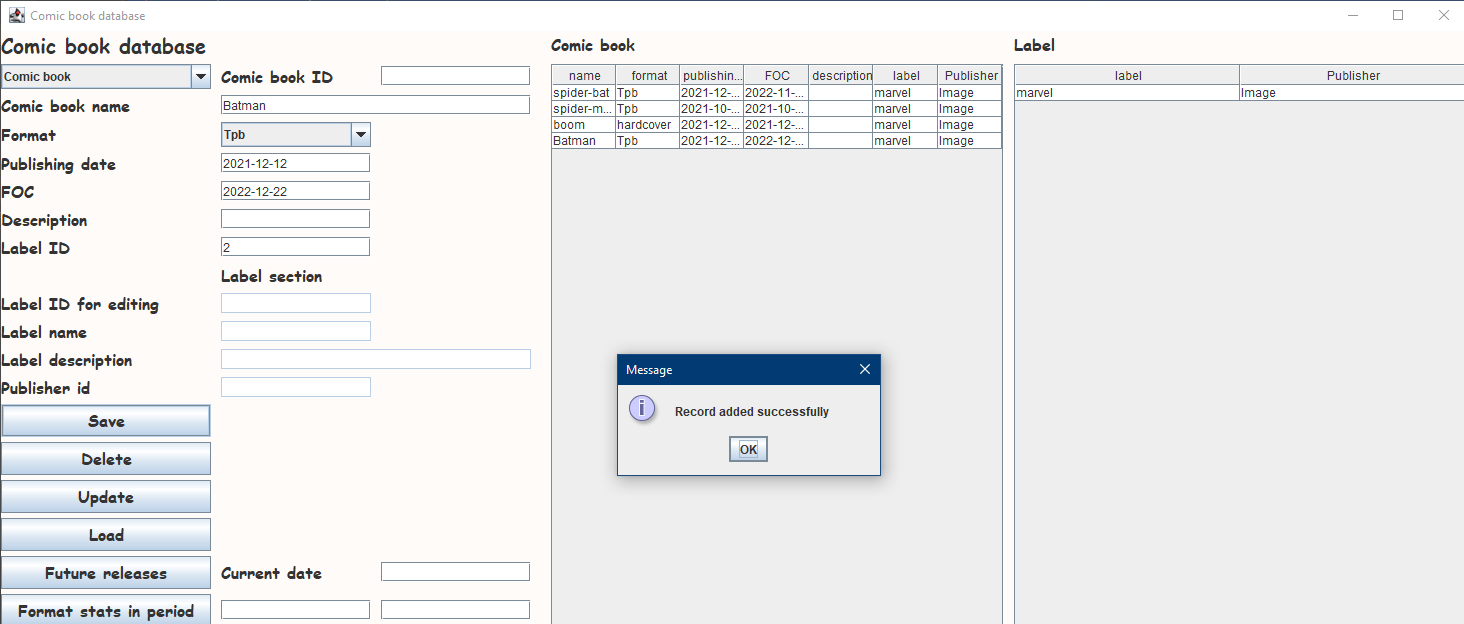
Результаты работы приложения:



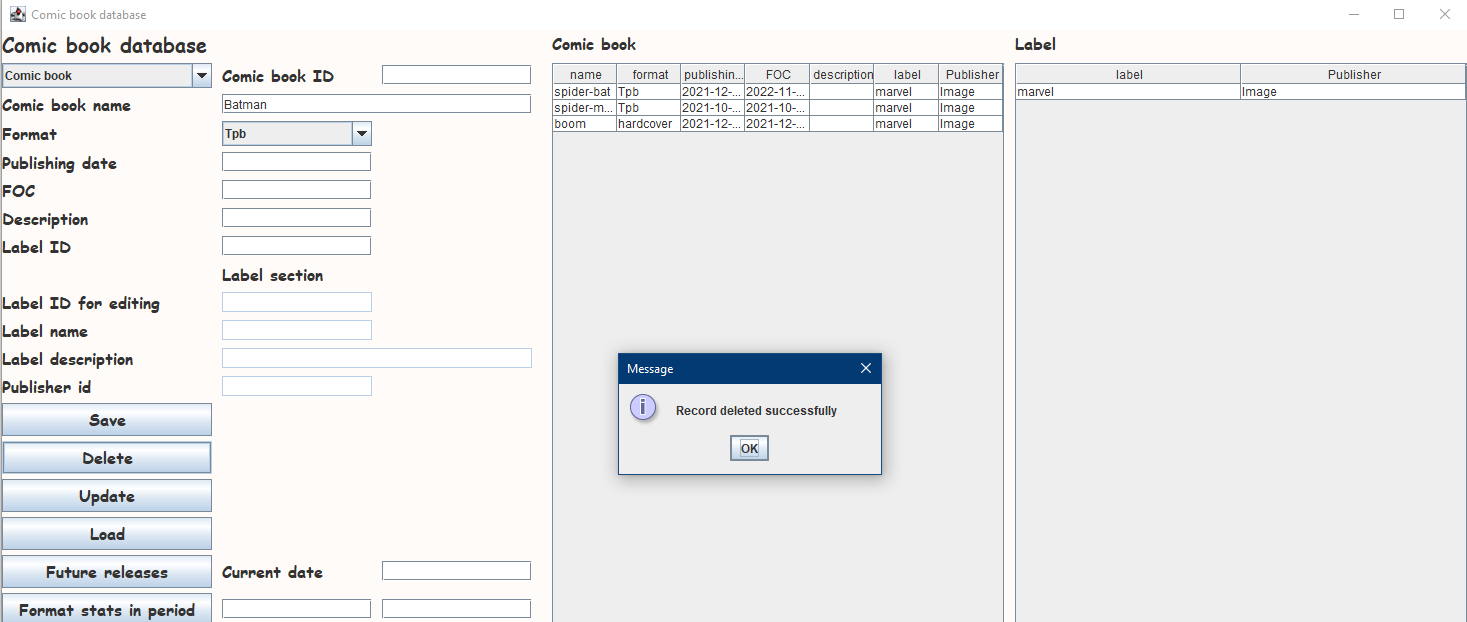
Загрузка данных из базы:



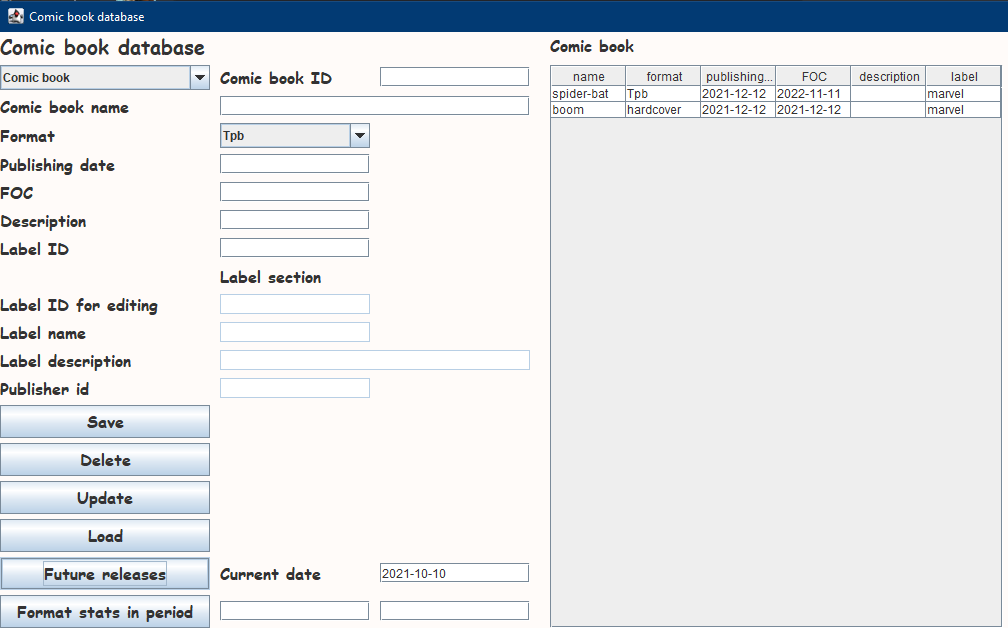
Добавление комикса в базу:



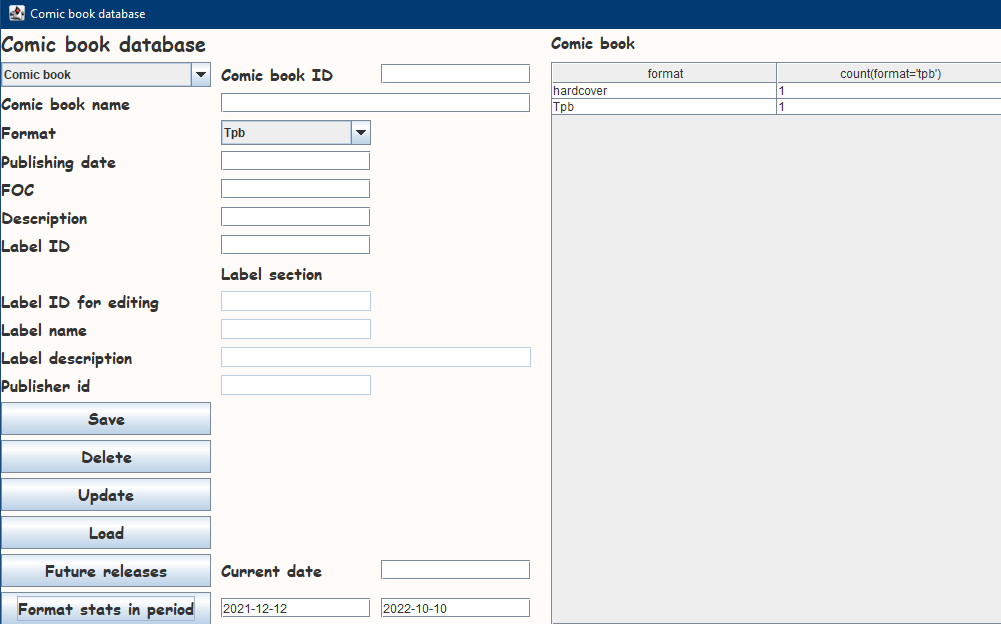
Удаление комикса из базы



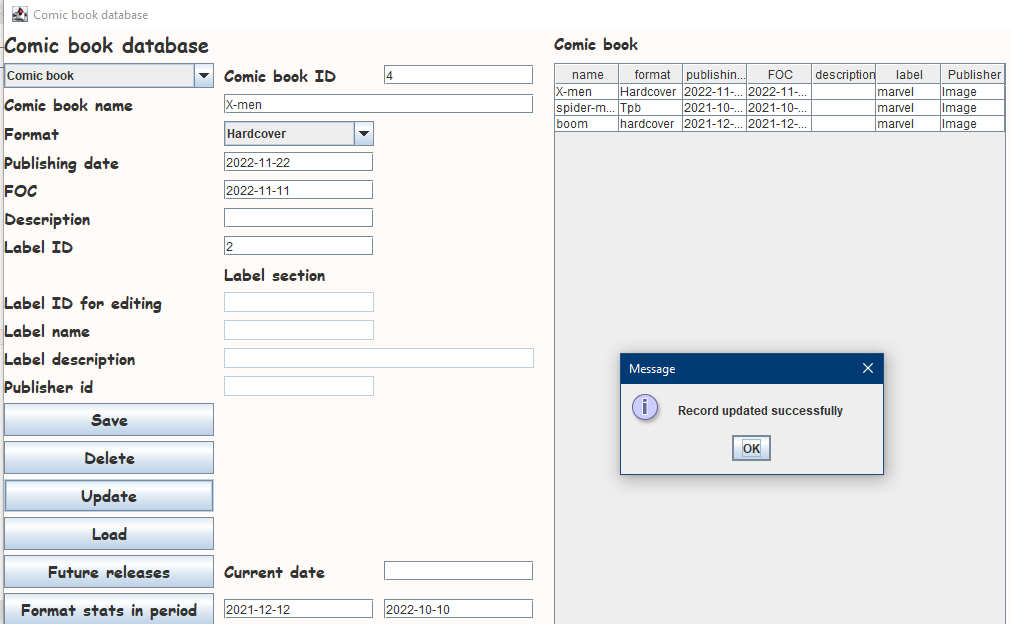
Комиксы, которые будут изданы в будущем(Future release):



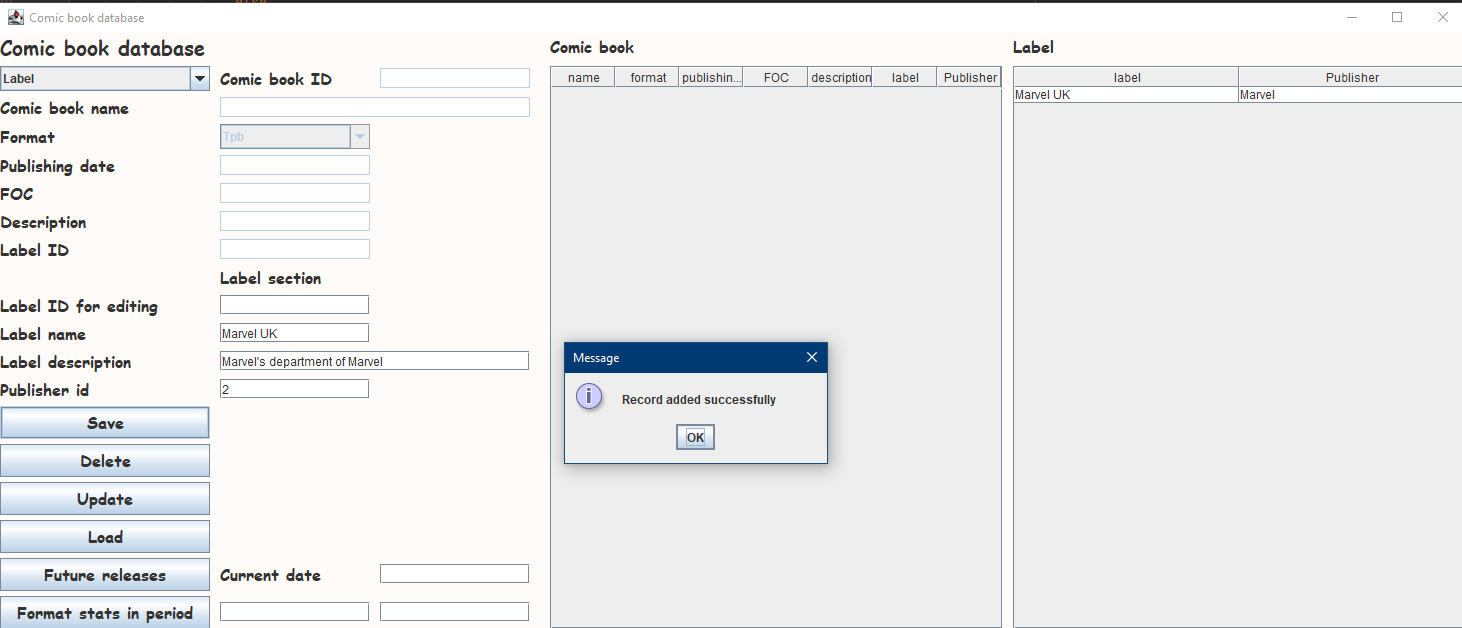
Статистика (количество) форматов за конкретный период (Format stats in period):



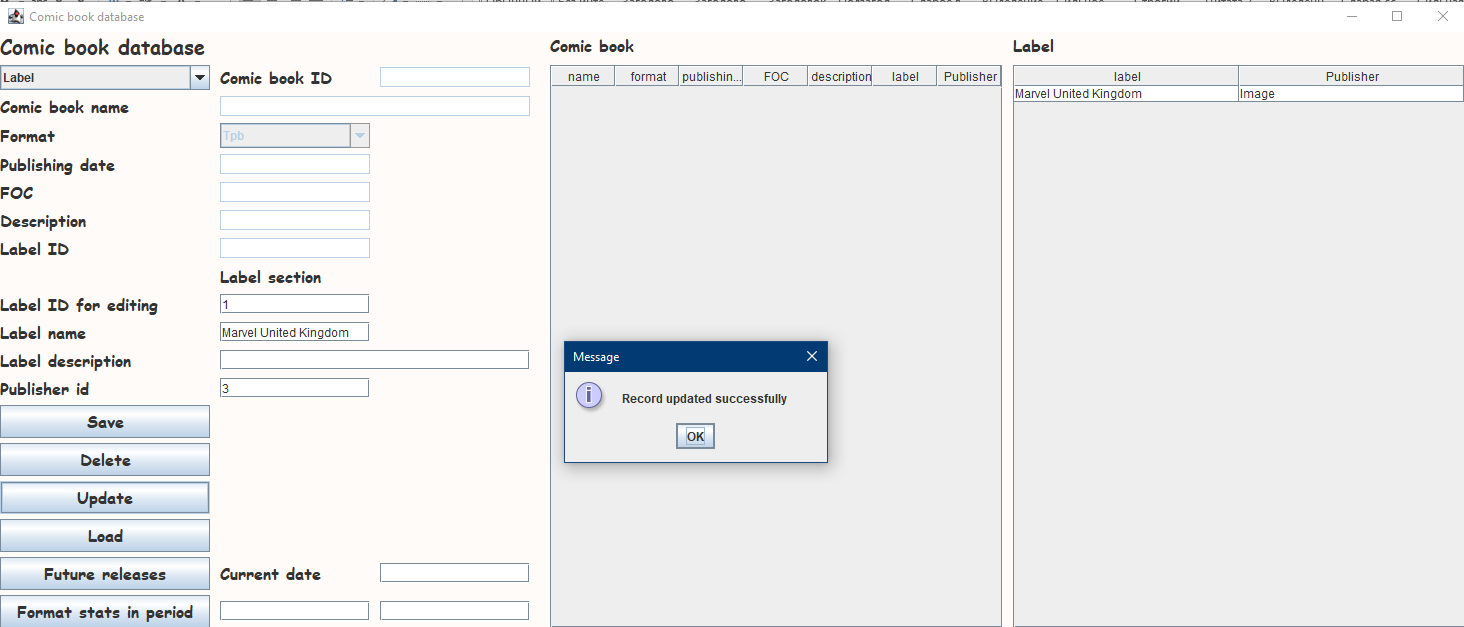
Редактирование записи(spider-bat из предыдущих скриншотов меняем на x-men):



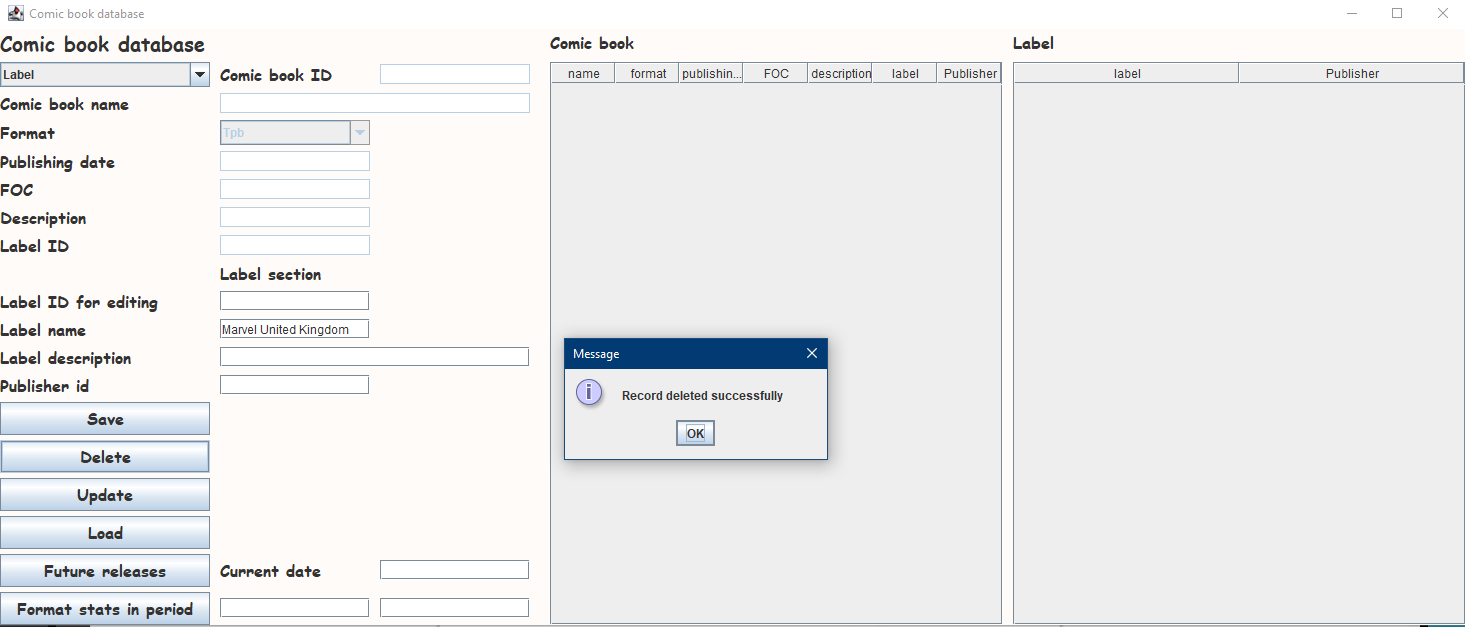
Добавление нового лейбла:



Редактирование лейбла:



Удаление лейбла:



**Вывод:**

В ходе работы была разработана программа на Java для работы с базой данных на SQL server.

**Заключение:**

В ходе работы была разработана CУБД «Магазин шин и дисков» в среде разработки MySQL Workbench и программа на языке Java для непосредственной работы с ней.

# Ссылка на репозиторий:

<https://github.com/wardennn1/DB_lab_067>

# Список литературы

1. Учебно-методическое пособие БД
2. Интернет-статья https://wiki.gentoo.org/wiki/MySQL/Startup\_Guide/ru
3. https://metanit.com/web/nodejs/