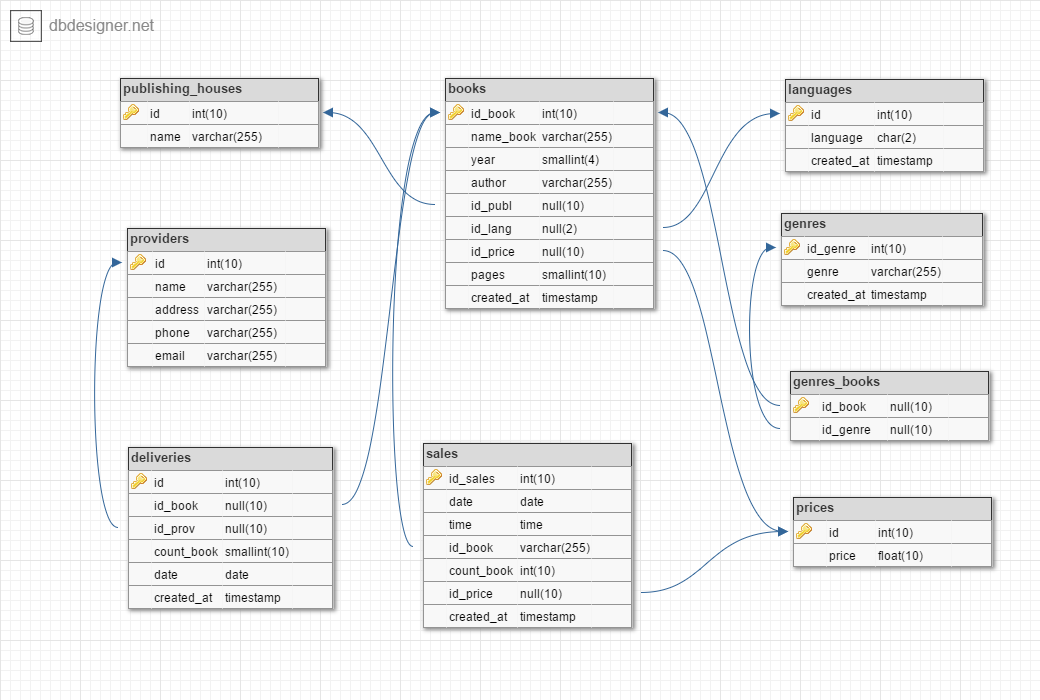
ПРОЕКТ ПО БАЗАМ ДАННЫХ

По теме “Книжный магазин”

Проект рассматривает данные о книжном магазине, полученные за период с 16.05.2017 по 07.06.2017.  
Магазин работает ежедневно с 10:00 до 20:00 без перерывов и выходных.  
Поставки новых книг осуществляются каждые 5 дней.



**Связь между таблицами**

**Таблица `languages`**

В таблице `languages` хранится список языков, на которых написаны книги, имеющиеся в магазине.

**Создание**

CREATE TABLE IF NOT EXISTS `languages` (

`id` INT(10) NOT NULL AUTO\_INCREMENT,

`languages` CHAR(2) NOT NULL,

`created\_at` TIMESTAMP NOT NULL,

PRIMARY KEY (`id`)

);

**Заполнение**

INSERT INTO `languages` (`language`)

VALUES

("en"), /английский

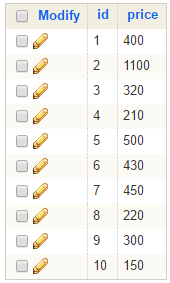
("ru"), /русский

("fr"), /французский

("de"), /немецкий

("cn"); /китайский

**Вид в БД**



**Таблица `publishing\_houses`**

В таблице представлен список издательств, которые выпустили книги, имеющиеся в магазине.

**Создание**

CREATE TABLE IF NOT EXISTS `publishing\_houses` (

`id` INT(10) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(255) NOT NULL,

PRIMARY KEY (`id`)

);

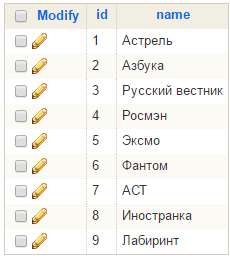
**Заполнение**

INSERT INTO `publishing\_houses` (`name`)

VALUES

("Астрель"),  
("Азбука"),  
("Русский вестник"),  
("Росмэн"),  
("Эксмо"),  
("Фантом"),  
("АСТ"),  
("Иностранка"),  
("Лабиринт");

**Вид в БД**



**Таблица `genres`**

Таблица содержит информацию о жанрах книг.

**Создание**

CREATE TABLE IF NOT EXISTS `genres` (

`id\_genre` INT(10) NOT NULL AUTO\_INCREMENT,

`genre` VARCHAR(255) NOT NULL,

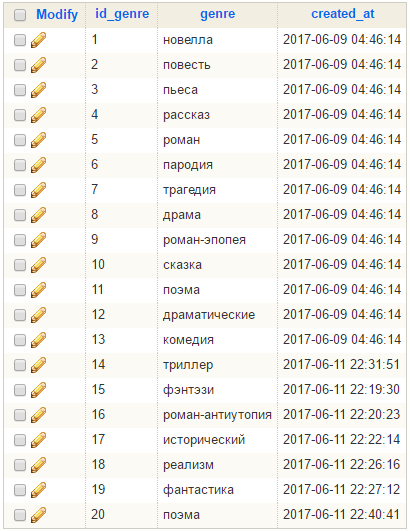
`created\_at` TIMESTAMP NOT NULL,

PRIMARY KEY (`id\_genre`)  
);

**Заполнение**

INSERT INTO `genres` (`genre`)  
VALUES  
("новелла"),  
("повесть"),  
("пьеса"),  
("рассказ"),  
("роман"),  
("пародия"),  
("трагедия"),  
("драма"),  
("роман-эпопея"),  
("сказка"),  
("поэма"),  
("драматические"),  
("комедия"),  
("триллер"),  
("фэнтэзи"),  
("роман-антиутопия"),  
("исторический"),  
("реализм"),  
("фантастика"),  
("поэма");

**Вид в БД**



**Таблица `sales`**

В таблице находится информация о продажах (журнал продаж).

**Создание**

CREATE TABLE IF NOT EXISTS `sales` (

`id\_sales` INT(10) NOT NULL AUTO\_INCREMENT,

`date` DATE NOT NULL,

`time` TIME NOT NULL,

`id\_book` VARCHAR(255) NOT NULL,

`count\_book` INT(10) NOT NULL,

`id\_price` INT(10) NOT NULL,

`created\_at` TIMESTAMP NOT NULL,

PRIMARY KEY (`id\_sales`),

KEY `id\_book` (`id\_book`),

KEY `id\_price` (`id\_price`)

);

**Заполнение**

INSERT INTO `sales` (`date`,`time`,`id\_book`,`count\_book`,`id\_price`)

VALUES

("2017-05-23", "10:08:57",1,1,4),  
("2017-05-25", "16:13:30",3,2,3),  
("2017-05-29", "17:00:57",8,1,6),  
("2017-05-31", "10:58:50",4,3,2),  
("2017-06-02", "12:16:24",6,1,3),  
("2017-06-02", "14:37:57",5,1,4),  
("2017-06-04", "15:49:00",11,2,9),  
("2017-06-05", "11:08:07",17,1,1),  
("2017-06-05", "17:59:16",13,1,7),  
("2017-06-05", "18:12:00",21,1,9),  
("2017-06-06", "18:21:24",9,1,7),  
("2017-06-06", "13:13:57",7,1,5),  
("2017-06-07", "15:51:45",16,1,5),  
("2017-06-07", "17:08:28",19,1,3),  
("2017-06-07", "19:58:36",22,1,9);

**Вид в БД**



**Таблица `deliveries`**

Таблица ​ `deliveries` **-** это журнал о поставках книг, определённым поставщиком.

**Создание**

CREATE TABLE IF NOT EXISTS `deliveries` (

`id` INT(10) NOT NULL AUTO\_INCREMENT,

`id\_book` INT(10) NOT NULL,

`id\_prov` INT(10) NOT NULL,

`count\_book` INT(10) NOT NULL,

`date` DATE NOT NULL,

`created\_at` TIMESTAMP NOT NULL,

PRIMARY KEY (`id`),

KEY `id\_book` (`id\_book`),

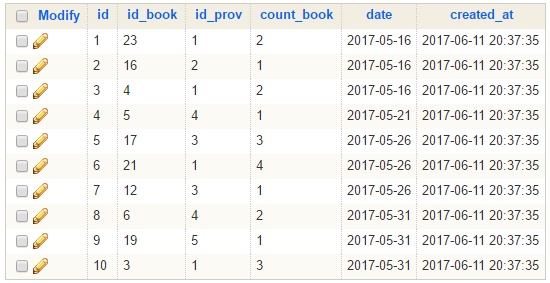
KEY `id\_prov` (`id\_prov`)

);

**Заполнение**

INSERT INTO `deliveries` (`id\_book`,`id\_prov`,`count\_book``date`)  
VALUES  
(23,1,2,"2017-05-16"),  
(16,2,1,"2017-05-16"),  
(4,1,2,"2017-05-16"),  
(5,4,1,"2017-05-21"),  
(17,3,3,"2017-05-26"),  
(21,1,4,"2017-05-26"),  
(12,3,1,"2017-05-26"),  
(6,4,2,"2017-05-31"),  
(19,5,1,"2017-05-31"),  
(3,1,3,"2017-05-31"),  
(10,5,2,"2017-06-04");  
(10,5,"2017-06-04");

**Вид в БД**



**Таблица `providers`**

В таблице хранится информация о поставщиках книг.

**Создание**

CREATE TABLE IF NOT EXISTS `providers` (

`id` INT(10) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(255) NOT NULL,

`address` VARCHAR(255) NOT NULL,

`phone` VARCHAR(255) NOT NULL,

`email` VARCHAR(255) NOT NULL,

PRIMARY KEY (`id`)

);

**Заполнение**

INSERT INTO `providers` (`name`,`address`,`phone`,`email`)

VALUES

("Книги к вам спешат","ул. Байкальская 21","8999546521","booktoyou@mail.ru"),

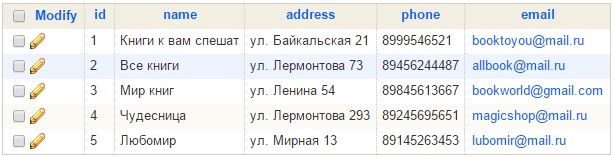
("Все книги","ул. Лермонтова 73","89456244487","allbook@mail.ru"),

("Мир книг","ул. Ленина 54","89845613667","bookworld@gmail.com"),

("Чудесница","ул. Лермонтова 293","89245695651","magicshop@mail.ru"),

("Любомир","ул. Мирная 13","89145263453","lubomir@mail.ru");

**Вид в БД**



**Таблица `books`**

В таблице хранятся данные о книгах, которые имеются в магазине.

**Создание**

CREATE TABLE IF NOT EXISTS `books` (

`id\_book` INT(10) NOT NULL AUTO\_INCREMENT,  
 `name\_book` VARCHAR(255) NOT NULL,  
 `year` SMALLINT(4) NOT NULL,  
 `author` VARCHAR(255) NOT NULL,  
 `id\_publ` INT(10) NOT NULL,  
 `id\_lang` CHAR(2) NOT NULL,  
 `id\_price` INT(10) NOT NULL,  
 `pages` SMALLINT(10) NOT NULL,  
 `created\_at` TIMESTAMP NOT NULL,  
 PRIMARY KEY (`id\_book`),  
 KEY `id\_publ` (`id\_publ`),  
 KEY `id\_price` (`id\_price`),  
 KEY `id\_lang` (`id\_lang`)  
);

**Заполнение**

INSERT INTO `books` (`name\_book`,`year`,`author`,`id\_publ`,`id\_lang`,`id\_price`,`pages`)

VALUES  
("13 причин почему",2011,"Джей Эшер",1,1,4,320),  
("Хорошо быть тихоней",1999,"Стивен Чбоски",2,1,1,352),   
("Преступление и наказание",1866,"Ф.М. Достоевский",3,2,3,574),  
("Гарри Поттер и дары смерти",200,"Джоан Роулинг",4,1,2,329),  
("451 градус по фаренгейту",1953,"Рэй Брэдбери",2,1,4,224),  
("Унесенные ветром",1936,"Маргарет Митчелл",5,1,3,640),  
("Прислуга",2009,"Кэтрин Стокетт",6,1,5,512),  
("А зори здесь тихие",1971,"Б.Л. Васильев",2,2,6,384),  
("Убить перемешника",1960,"Харпер Ли",7,1,7,288),  
("Над пропастью во ржи",1951,"Джером Сэлинджер",5,1,5,213),  
("Мастер и маргарита",1966,"М.А. Булгаков",2,2,9,358),  
("Книжный вор",2005,"Маркус Зусак",5,1,10,560),  
("Гордость и предубеждение",1813,"Джейн Остин",2,1,7,374),  
("Бегущий по лабиринту",2009," Джеймс Дэшнер",7,1,5,448),  
("Три мушкетера",1844,"Александр Дюма",5,3,3,667),  
("До встречи с тобой",2012,"Джоджо Мойес",8,1,5,480),  
("Лев, колдунья и платяной шкаф",1984,"Клайв Льюис",5,1,1,280),  
("Война и мир. Том 1",1869,"Л.Н. Толстой",5,2,10,308),

("Анна Каренина",1877,"Ф. М. Достоевский",5,2,3,766),  
("Алиса в стране чудес",1865,"Льюис Кэрролл",9,1,4,272),  
("Бойцовский клуб",1996,"Чак Паланик",7,1,6,256),  
("Великий гэтсби",1922,"Фрэнсис Фицджеральд",2,1,9,256),  
("Маленький принц",1943,"Антуан де Сент-Экзюпери",7,3,8,86),  
("Герой нашего времени",1940,"М.Ю. Лермонтов",2,2,1,257),  
("Мертвые души",1842,"Н.В. Гоголь",2,2,3,550);

**Вид в БД**



**Таблица `genres\_books`**

Данная таблица помогает осуществить связь “многие-ко-многим”.

**Создание**

CREATE TABLE IF NOT EXISTS `genres\_books` (

`id\_book` INT(10) NOT NULL,

`id\_genre` INT(10) NOT NULL,

PRIMARY KEY (`id\_book`,`id\_genre`),

KEY `id\_book` (`id\_book`),

KEY `id\_genre` (`id\_genre`)

);

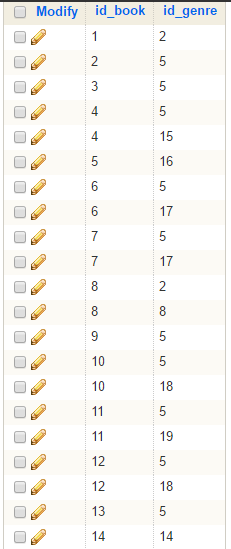
**Заполнение**

INSERT INTO `genres\_books` (`id\_book`,`id\_genre`)

VALUES  
(1,2),  
(2,5),  
(3,5),  
(4,15),  
(4,5),  
(5,16),  
(6,17),  
(6,5),  
(7,5),  
(7,17),  
(8,2),  
(8,8),  
(9,5),  
(10,5),  
(10,18),  
(11,5),  
(11,19),  
(12,5),  
(12,18),  
(13,5),  
(14,14),  
(14,16),  
(15,5),  
(15,17),

(16,8),  
(17,15),  
(18,9),  
(18,17),  
(19,17),  
(19,5),  
(19,18),  
(20,10),  
(21,8),  
(22,5),  
(22,8),  
(23,1),  
(23,19),  
(24,18),  
(24,5),  
(25,20);

**Вид в БД**

****

**Таблица `prices`**

В таблице хранятся значения цен книг.

**Создание**

CREATE TABLE IF NOT EXISTS `prices` (

`id` INT(10) NOT NULL AUTO\_INCREMENT,

`price` FLOAT(10) NOT NULL,

PRIMARY KEY (`id`)

);

**Заполнение**

INSERT INTO `prices` (`price`)

VALUES

(400),  
(1100),  
(320),  
(210),  
(500),  
(430),  
(450),  
(220),  
(300),  
(150);

**Вид в БД**



**Запросы**

1. Запрос выдаст наименование книги, какой компанией она была доставлена и дата ее доставки.

SELECT

`b`.`name\_book`,

`prov`.`name`,

`d`.`date`

FROM `deliveries` `d`

JOIN `books` `b` ON `d`.`id\_book`=`b`.`id`

JOIN `providers` `prov` ON `d`.`id\_prov`=`prov`.`id`

ORDER BY `b`.`name\_book` ASC;



1. Запрос выдаст выручку магазина за определённый день (2017-06-04)

SELECT

`s`.`date`,

SUM(`s`.`count\_book`\*`p`.`price`) `receipts`

FROM `sales` `s`

JOIN `prices` `p` ON `s`.`id\_price`=`p`.`id`

WHERE `s`.`date` = "2017-06-04";



1. Запрос выдаст информацию о количестве книг доставленных каждым поставщиком.

SELECT

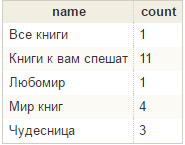
`prov`.`name`,

SUM(`d`.`count\_book`) `count`

FROM `deliveries` `d`

JOIN `providers` `prov` ON `d`.`id\_prov`=`prov`.`id`

GROUP BY `prov`.`name`;



1. Запрос выдаст день с самой большой выручкой.

SELECT

`s`.`date`,

MAX(`s`.`count\_book`\*`p`.`price`) `receipts`

FROM `sales` `s`

JOIN `prices` `p` ON `s`.`id\_price`=`p`.`id`

ORDER BY `s`.`date`;



**Представление**

Создаем представления для таблиц `info` и `book\_info`, чтобы была  
возможность использовать эти данные для более удобного представления информации.

В представлении `info` содержится информация о книгах в алфавитном порядке, издательстве, страницах, цене и языке, на котором написана книга.

**Создание**

CREATE VIEW `info` AS   
SELECT  
`b`.`name\_book`,  
`pub`.`name` `name\_publ`,  
`b`.`pages`,  
`p`.`price`,  
`lang`.`language`  
FROM `books` `b`  
JOIN `publishing\_houses` `pub` ON `b`.`id\_publ`=`pub`.`id`  
JOIN `prices` `p` ON `b`.`id\_price`=`p`.`id`  
JOIN `languages` `lang` ON `b`.`id\_lang`=`lang`.`id`  
ORDER BY `b`.`name\_book`;

**Вид в БД**



В представлении `book\_info` содержится информация о книгах, годе выпуска, авторе и жанрах книг.

**Создание**

CREATE VIEW `book\_info` AS  
SELECT  
`b`.`name\_book`,  
`b`.`year`,  
`b`.`author`,  
GROUP\_CONCAT(`g`.`genre`) as `genres\_book`  
FROM `books` `b`  
JOIN `genres\_books` `gb` ON `b`.`id`=`gb`.`id\_book`  
JOIN `genres` `g` ON `g`.`id\_genre`=`gb`.`id\_genre`  
GROUP BY `b`.`name\_book`;

**Вид в БД**

