

Связи между таблицами, соединение таблиц

Реляционная модель

Преподаватель: канд. тех. наук, доц.
Озерова Г.П.

Предметная область «Склад»

На некотором складе хранятся книги. Каждая книга имеет одного автора и относится к одному жанру. Для каждой книги известно ее название, цена и количество ее экземпляров.

Связи между таблицами

genre	
PK	genre_id
	name_genre

author	
PK	author_id
	name_author

book	
PK	book_id
	title
	author_id
	genre_id
	price
	amount

PK – первичный ключ в таблице

Связи между таблицами

genre	
PK	genre_id
	name_genre

author	
PK	author_id
	name_author

book	
PK	book_id
	title
	author_id
	genre_id
	price
	amount

Связи реализуются между двумя таблицами

Связи между таблицами

author	
PK	author_id
	name_author

← главная таблица

book	
PK	book_id
	title
	author_id
	genre_id
	price
	amount

← связанная (зависимая) таблица

Связи между таблицами

author	
PK	author_id
	name_author

← главная таблица

book	
PK	book_id
	title
	author_id
	genre_id
	price
	amount

← связанная (зависимая) таблица,
в ней используется значение
первичного ключа главной
таблицы

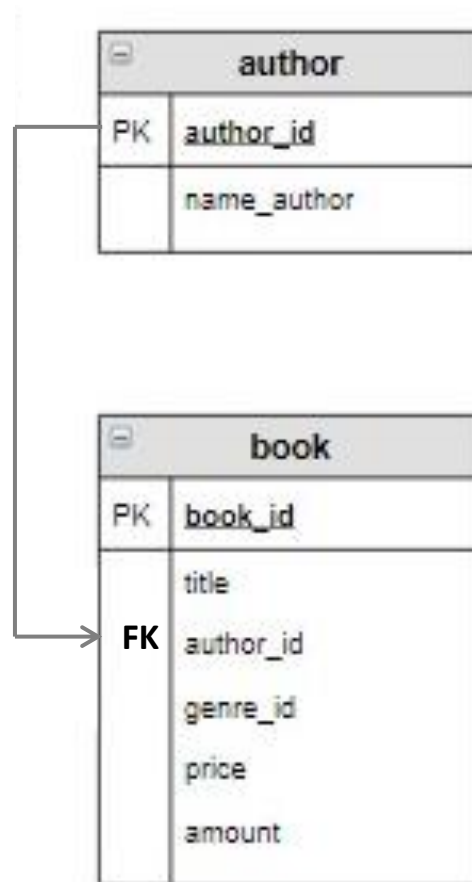
Связи между таблицами

author	
PK	author_id
	name_author

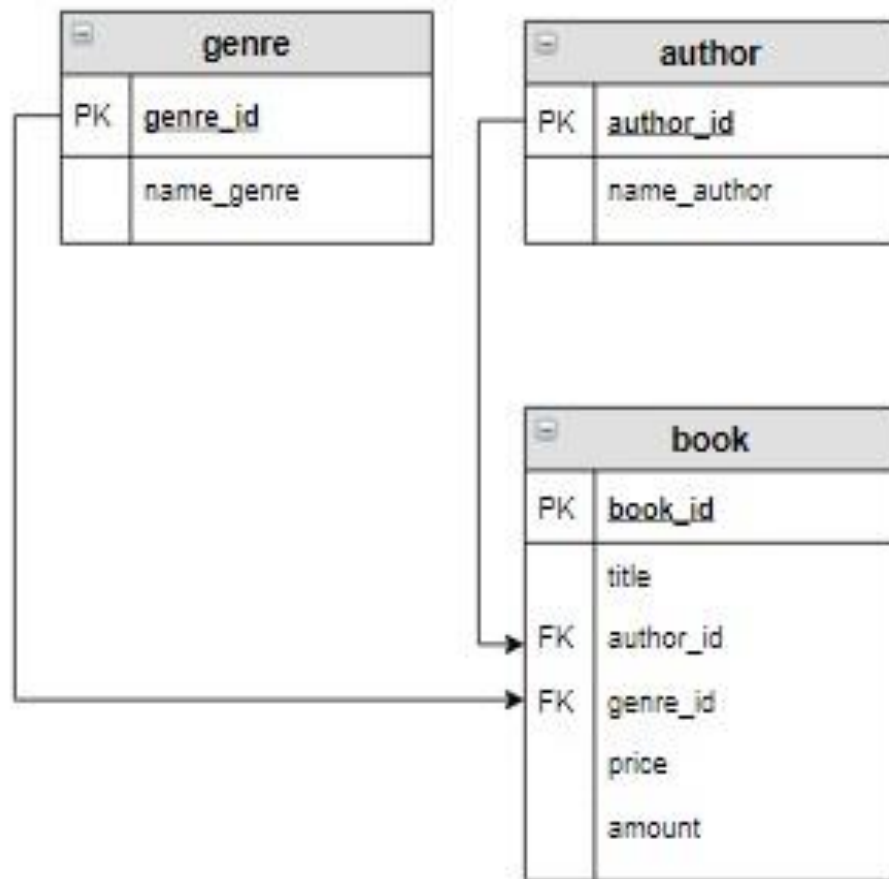
book	
PK	book_id
	title
FK	author_id
	genre_id
	price
	amount

← этот столбец называется внешним ключом, обозначается **FK**, используется для соединения таблиц

Связи между таблицами



Связи между таблицами



Внешний ключ

Внешний ключ - столбец в зависимой таблице, который служит для связи с главной таблицей.

Внешний ключ должен быть совместимым по типу с первичным ключом главной таблицы (если первичный ключ **INT**, то внешний ключ тоже **INT**).

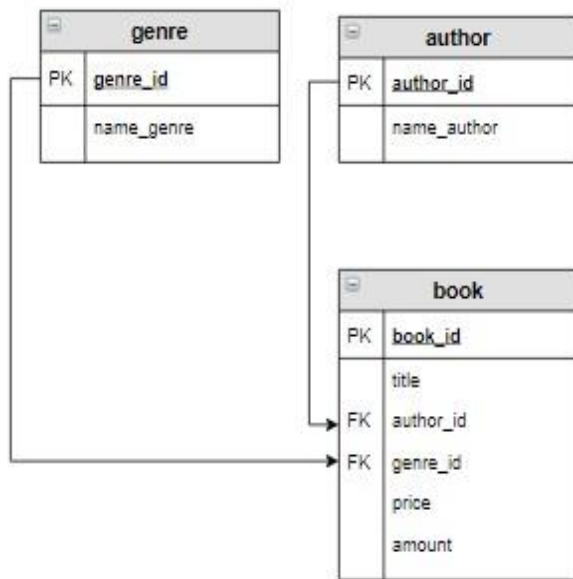
Для внешнего ключа необходимо включить описание его свойств при создании зависимой таблицы.

Внешний ключ

Описание внешнего ключа в зависимой таблице

FOREIGN KEY (внешний_ключ)
REFERENCES главная_таблица
(первичный_ключ)

Внешний ключ



```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Пустое значение

NULL используется для того, чтобы указать, что поле содержит пустое значение.

Для проверки поля на пустоту (не пустоту) используются выражения:

IS NULL и **IS NOT NULL**

Ключевые поля НЕ МОГУТ содержать пустые значения.

Пустое значение

При создании таблицы можно указать, что любое поле не может содержать пустых значений:

```
CREATE TABLE таблица (  
    столбец_1 тип_данных NOT NULL,  
    столбец_2 тип_данных  
);
```


Внешний ключ, непустое значение

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Внешний ключ

При описании внешнего ключа можно указать действия, которые необходимо выполнить для записей зависимой таблицы при удалении и изменении записей главной таблицы.

Для этого используются опции

- **ON DELETE**
- **ON UPDATE**

Внешний ключ, ON DELETE

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

CASCADE - автоматически удаляет строки из зависимой таблицы при удалении связанных строк в главной таблице;

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id) ON DELETE CASCADE  
);
```

Внешний ключ, ON DELETE CASCADE

Пример. Из таблицы **genre** удалить жанр «Роман» .

```
DELETE FROM genre  
WHERE name_genre = "Роман";
```

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

ON DELETE CASCADE

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE CASCADE

author

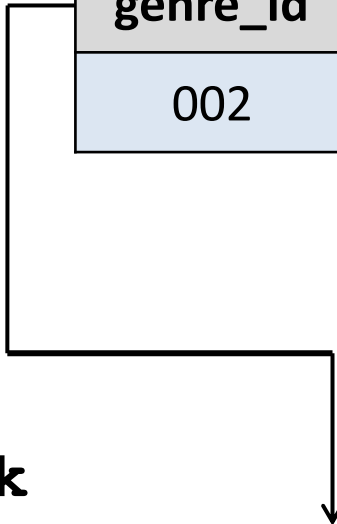
author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
4	Стихотворения и поэмы	03	002	650.00	15



Внешний ключ, ON DELETE SET NULL

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

SET NULL при удалении связанной строки из главной таблицы устанавливает для столбца внешнего ключа значение **NULL**.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id) ON DELETE SET NULL,  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Внешний ключ, ON DELETE SET NULL

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

SET NULL при удалении связанной строки из главной таблицы устанавливает для столбца внешнего ключа значение **NULL**.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id) ON DELETE SET NULL,  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

удаление не будет выполнено

Внешний ключ, ON DELETE SET NULL

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

SET NULL при удалении связанной строки из главной таблицы устанавливает для столбца внешнего ключа значение **NULL**.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id) ON DELETE SET NULL,  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Внешний ключ, ON DELETE SET NULL

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

SET NULL при удалении связанной строки из главной таблицы устанавливает для столбца внешнего ключа значение **NULL**.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id) ON DELETE SET NULL,  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Внешний ключ, ON DELETE SET NULL

Пример. Из таблицы **author** удалить автора «Булгаков М.А.».

```
DELETE FROM author  
WHERE name_author = "Булгаков М.А.";
```

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET NULL

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET NULL

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET NULL

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

ON DELETE SET NULL

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET NULL

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET NULL

author

author_id	name_author
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	NULL	001	670.99	3
2	Белая гвардия	NULL	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

SET DEFAULT при удалении связанной строки из главной таблицы в подчиненной таблице внешний ключ устанавливается в значение по умолчанию.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT DEFAULT 03,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id)  
        ON DELETE SET DEFAULT,  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)  
);
```

Внешний ключ, ON DELETE SET DEFAULT

Пример. Из таблицы **author** удалить автора «Булгаков М.А.».

```
DELETE FROM author  
WHERE name_author = "Булгаков М.А.";
```

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

ON DELETE SET DEFAULT

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE SET DEFAULT

author

author_id	name_author
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	03	001	670.99	3
2	Белая гвардия	03	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

С помощью выражения **ON DELETE** можно установить действия, которые выполняются для записей подчиненной таблицы при удалении связанной строки из главной таблицы.

RESTRICT отклоняет удаление строк в главной таблице при наличии связанных строк в зависимой таблице.

```
CREATE TABLE book(
    book_id INT PRIMARY KEY AUTO_INCREMENT,
    title VARCHAR(50),
    author_id INT,
    genre_id INT,
    price DECIMAL(8,2),
    amount INT,
    FOREIGN KEY (author_id) REFERENCES author(author_id),
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id)
    ON DELETE RESTRICT
);
```

Внешний ключ, ON DELETE RESTRICT

Пример. Из таблицы **genre** удалить жанры «Роман» и «Приключения».

```
DELETE FROM genre  
WHERE
```

```
name_genre = "Роман" OR name_genre = "Фэнтези";
```

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

ON DELETE RESTRICT

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Ошибка при выполнении запроса

Внешний ключ, ON UPDATE

С помощью выражения **ON UPDATE** можно установить действия, которые выполняются для записей подчиненной таблицы при обновлении связанного поля главной таблицы.

CASCADE при изменении значения связанного поля в главной таблицы, на эти значения автоматически обновляются связанные поля подчиненной таблицы

Внешний ключ, ON UPDATE

С помощью выражения **ON UPDATE** можно установить действия, которые выполняются для записей подчиненной таблицы при обновлении связанного поля главной таблицы.

CASCADE при изменении значения связанного поля в главной таблицы, на эти значения автоматически обновляются связанные поля подчиненной таблицы

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT NOT NULL,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id) ON UPDATE CASCADE  
);
```

Внешний ключ, ON UPDATE, пример

Пример. В таблицы **genre** изменить **genre_id** жанра «Роман» на 100.

```
UPDATE genre  
SET genre_id = 100  
WHERE name_genre = "Роман";
```

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

ON UPDATE CASCADE

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
100	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE CASCADE

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
100	Роман
002	Поэзия

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	100	670.99	3
2	Белая гвардия	01	100	540.50	5
3	Игрок	02	100	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE

С помощью выражения **ON UPDATE** можно установить действия, которые выполняются для записей подчиненной таблицы при обновлении связанного поля главной таблицы.

RESTRICT отклоняет изменение поле в главной таблице при наличии связанных полей в зависимой таблице.

```
CREATE TABLE book(  
    book_id INT PRIMARY KEY AUTO_INCREMENT,  
    title VARCHAR(50),  
    author_id INT,  
    genre_id INT,  
    price DECIMAL(8,2),  
    amount INT,  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
    FOREIGN KEY (genre_id) REFERENCES genre(genre_id) ON UPDATE RESTRICT  
);
```

Внешний ключ, ON UPDATE, пример

Пример. В таблице **genre** обновить **id_genre** жанра «Роман» на 100, а обновить **id_genre** жанра «Фэнтези» на 200.

```
UPDATE genre
SET genre_id = IF(name_genre = "Роман", 100,
                  IF(name_genre = "Фэнтези", 200, genre_id)
);
```

Внешний ключ, ON UPDATE

С помощью выражения **ON UPDATE** можно установить действия, которые выполняются для записей подчиненной таблицы при обновлении связанного поля главной таблицы.

RESTRICT отклоняет изменение поле в главной таблице при наличии связанных полей в зависимой таблице.

Логическая схема базы данных

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book

ON UPDATE RESTRICT

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON UPDATE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
300	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Внешний ключ, ON DELETE RESTRICT

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
300	Фэнтези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Ошибка при выполнении запроса

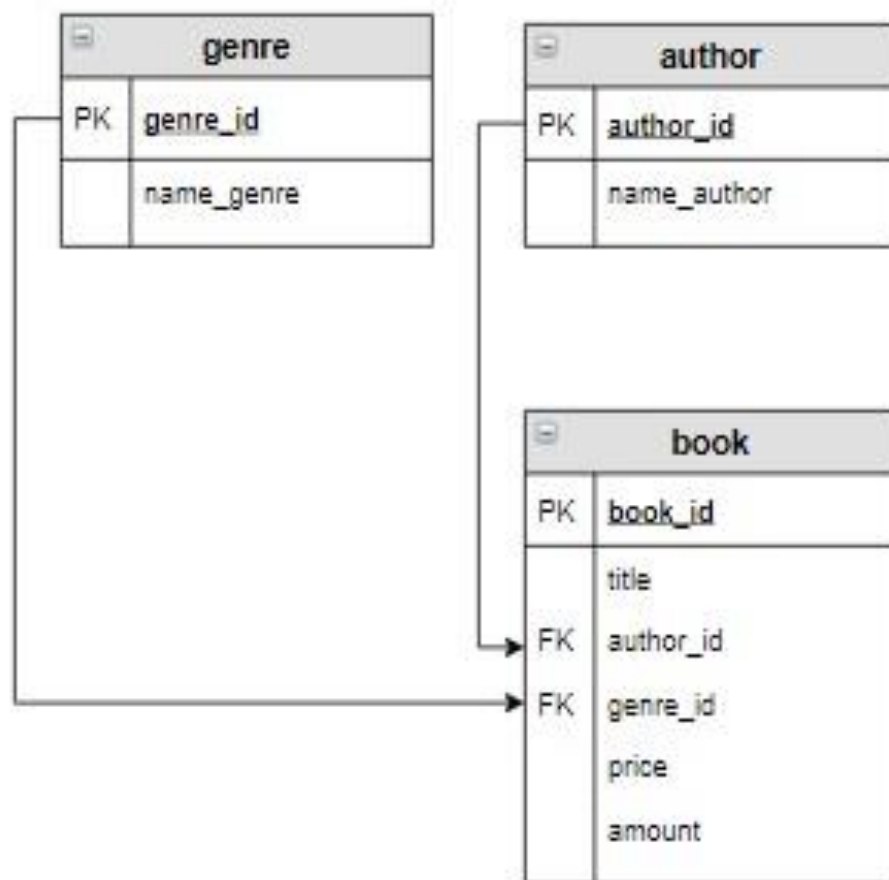
Внешний ключ, ON UPDATE

С помощью выражения **ON UPDATE** можно установить действия, которые выполняются для записей подчиненной таблицы при обновлении связанного поля главной таблицы.

- ✓ **SET NULL** при изменении поля строки из главной таблицы для соответствующих полей внешнего ключа связанной таблицы устанавливает значение **NULL**.
- ✓ **SET DEFAULT** похоже на SET NULL за тем исключением, что значение внешнего ключа устанавливается не в NULL, а в значение по умолчанию для данного столбца.

Самостоятельно разобраться с этими двумя опциями.

Логическая модель



Структура и наполнение таблиц

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фантези

book

book_id	title	author_id	genre_id	price	amount
1	Мастер и Маргарита	01	001	670.99	3
2	Белая гвардия	01	001	540.50	5
3	Игрок	02	001	480.50	10
4	Стихотворения и поэмы	03	002	650.00	15

Соединение таблиц

Соединение двух таблиц заключается в объединении каждой строки первой таблицы с каждой строкой второй и проверки условия соединения.

Результатом операции соединения являются объединенные строки двух таблиц, для которых верно условие соединения.

Для реализации соединения в SQL используется оператор **JOIN**.

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
---------	-------	---------------	----------------	------------

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001		

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001		

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001		

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001		

Соединение таблиц

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фэнтези

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман

Объединенные таблицы

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Соединение таблиц

Типы соединений:

- ✓ внутреннее соединение (**INNER JOIN**);
- ✓ внешнее соединение (**LEFT JOIN, RIGHT JOIN**);
- ✓ перекрестное соединение (**CROSS JOIN**).

Внутреннее соединение (INNER JOIN)

При использовании **внутреннего соединения** отбираются те объединенные строки, в которых совпадают значения одного или нескольких полей.

Внутреннее соединение (INNER JOIN)

Синтаксис:

```
таблица_1  
INNER JOIN таблица_2  
ON таблица_1.поле = таблица_2.поле
```

Внутреннее соединение (INNER JOIN)

Соединение таблиц **book** и **genre**.

book

INNER JOIN genre

ON book.genre_id = genre.genre_id

Объединенные таблицы

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	 001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	 002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001 	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Структура и наполнение таблиц

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002 	003	Фэнтези

Внутреннее соединение

book

genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
2	Белая гвардия	001	001	Роман
3	Стихотворения	002	002	Поэзия

Внутреннее соединение (INNER JOIN)

Порядок соединения таблиц не имеет значения.

```
таблица_1  
INNER JOIN таблица_2  
ON таблица_1.поле = таблица_2.поле
```



```
таблица_2  
INNER JOIN таблица_1  
ON таблица_2.поле = таблица_1.поле
```

Внутреннее соединение (INNER JOIN)

В запросе соединение таблиц описывается в разделе **FROM**.

FROM

таблица_1

INNER JOIN таблица_2

ON таблица_1.поле = таблица_2.поле

Внутреннее соединение, пример

Пример. Вывести названия книг и их жанры.

```
SELECT title, name_genre  
FROM  
    book  
    INNER JOIN genre ON book.genre_id = genre.genre_id;
```

Как выполняется запрос, шаг 1

FROM book **JOIN** genre

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Как выполняется запрос, шаг 1

```
FROM book INNER JOIN genre  
ON book.genre_id = genre.genre_id
```

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
1	Мастер и Маргарита	001	003	Фэнтези
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
2	Белая гвардия	001	003	Фэнтези
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия
3	Стихотворения	002	003	Фэнтези

Как выполняется запрос, шаг 2

```
FROM book INNER JOIN genre  
ON book.genre_id = genre.genre_id
```

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
2	Белая гвардия	001	001	Роман
3	Стихотворения	002	002	Поэзия

Как выполняется запрос, шаг 2

```
SELECT title, name_genre
```

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
2	Белая гвардия	001	001	Роман
3	Стихотворения	002	002	Поэзия

Как выполняется запрос, шаг 2

```
SELECT title, name_genre
```

title	name_genre
Мастер и Маргарита	Роман
Белая гвардия	Роман
Стихотворения	Поэзия

Соединение таблиц

Типы соединений:

- ✓ внутреннее соединение (**INNER JOIN**);
- ✓ внешнее соединение (**LEFT JOIN**);

Внешнее соединение (LEFT JOIN)

Синтаксис:

```
таблица_1  
LEFT JOIN таблица_2  
ON таблица_1.поле = таблица_2.поле
```


Внешнее соединение (LEFT JOIN)

При использовании внешнего соединения **LEFT JOIN**:

- в результат включается внутреннее соединение (**INNER JOIN**) первой и второй таблицы в соответствии с условием;
- затем в результат добавляются те записи первой таблицы, которые не вошли во внутреннее соединение на шаге 1, для таких записей поля второй таблицы заполняются значениями NULL.

Внешнее соединение (LEFT JOIN)

Соединение таблиц **genre** и **book**.

genre

LEFT JOIN book

ON genre.genre_id = book.genre_id

LEFT JOIN, объединенные таблицы

genre

book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
002	Поэзия	1	Мастер и Маргарита	001
003	Фэнтези	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	2	Белая гвардия	001
003	Фэнтези	2	Белая гвардия	001
001	Роман	3	Стихотворения	002
002	Поэзия	3	Стихотворения	002
003	Фэнтези	3	Стихотворения	002

LEFT JOIN, шаг 1

genre

book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
002	Поэзия	1	Мастер и Маргарита	001
003	Фэнтези	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	2	Белая гвардия	001
003	Фэнтези	2	Белая гвардия	001
001	Роман	3	Стихотворения	002
002	Поэзия	3	Стихотворения	002
003	Фэнтези	3	Стихотворения	002

LEFT JOIN, шаг 1

genre

book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002

LEFT JOIN, шаг 2

genre

book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002
003	Фэнтези			

LEFT JOIN, шаг 2

genre

book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002
003	Фэнтези	NULL	NULL	NULL

Внешнее соединение LEFT JOIN, пример

Пример. Вывести те жанры, книг которых нет на складе.

```
SELECT name_genre
FROM
    genre
    LEFT JOIN book ON genre.genre_id = book.genre_id
WHERE title IS NULL;
```


Как выполняется запрос, шаг 1

FROM genre **JOIN** book

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
002	Поэзия	1	Мастер и Маргарита	001
003	Фэнтези	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	2	Белая гвардия	001
003	Фэнтези	2	Белая гвардия	001
001	Роман	3	Стихотворения	002
002	Поэзия	3	Стихотворения	002
003	Фэнтези	3	Стихотворения	002

Как выполняется запрос, шаг 1

FROM genre

LEFT JOIN book **ON** genre.genre_id = book.genre_id

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002

Как выполняется запрос, шаг 1

FROM genre

LEFT JOIN book **ON** genre.genre_id = book.genre_id

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002
003	Фэнтези	NULL	NULL	NULL

Как выполняется запрос, шаг 2

WHERE title **IS NULL**

genre.genre_id	name_genre	book_id	title	book.genre_id
001	Роман	1	Мастер и Маргарита	001
001	Роман	2	Белая гвардия	001
002	Поэзия	3	Стихотворения	002
003	Фэнтези	NULL	NULL	NULL

Как выполняется запрос, шаг 2

WHERE title **IS NULL**

genre.genre_id	name_genre	book_id	title	book.genre_id
003	Фэнтези	NULL	NULL	NULL

Как выполняется запрос, шаг 3

SELECT name_genre

genre.genre_id	name_genre	book_id	title	book.genre_id
003	Фэнтези	NULL	NULL	NULL

Как выполняется запрос, шаг 3

```
SELECT name_genre
```

name_genre
Фэнтези

Соединение таблиц

Типы соединений:

- ✓ внутреннее соединение (**INNER JOIN**);
- ✓ внешнее соединение (**RIGHT JOIN**);

Внешнее соединение (RIGHT JOIN)

Синтаксис:

```
таблица_1  
RIGHT JOIN таблица_2  
ON таблица_1.поле = таблица_2.поле
```

Внешнее соединение (RIGHT JOIN)

При использовании внешнего соединения **RIGHT JOIN**:

- в результат включается внутреннее соединение (**INNER JOIN**) первой и второй таблицы в соответствии с условием;
- затем в результат добавляются те записи второй таблицы, которые не вошли во внутреннее соединение на шаге 1, для таких записей поля первой таблицы заполняются значениями NULL.

Внешнее соединение (LEFT и RIGHT JOIN)

таблица_1
RIGHT JOIN таблица_2
ON таблица_1.поле = таблица_2.поле



таблица_2
LEFT JOIN таблица_1
ON таблица_1.поле = таблица_2.поле

Соединение таблиц

Типы соединений:

- ✓ внутреннее соединение (**INNER JOIN**);
- ✓ внешнее соединение (**LEFT JOIN, RIGHT JOIN**);
- ✓ перекрестное соединение (**CROSS JOIN**).

Перекрёстное соединение (CROSS JOIN)

Синтаксис:

таблица_1
CROSS JOIN таблица_2



таблица_1, таблица_2

Перекрёстное соединение (CROSS JOIN)

При использовании перекрёстного соединения **CROSS JOIN** результат запроса формируется так:

- каждая строка одной таблицы соединяется с каждой строкой другой таблицы,
- в результате получаются все возможные сочетания строк двух таблиц.

Перекрёстное соединение (CROSS JOIN)

Соединение таблиц **genre** и **book**.

`genre, book`

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001		

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001		

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
2	Белая гвардия	001		

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
2	Белая гвардия	001	001	Роман

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
3	Стихотворения	002	001	Роман

Перекрёстное соединение (CROSS JOIN)

book

book_id	title	genre_id
1	Мастер и Маргарита	001
2	Белая гвардия	001
3	Стихотворения	002

genre

genre_id	name_genre
001	Роман
002	Поэзия

book_id	title	book.genre_id	genre.genre_id	name_genre
1	Мастер и Маргарита	001	001	Роман
1	Мастер и Маргарита	001	002	Поэзия
2	Белая гвардия	001	001	Роман
2	Белая гвардия	001	002	Поэзия
3	Стихотворения	002	001	Роман
3	Стихотворения	002	002	Поэзия

Перекрестное соединение, пример

Пример. Составить все возможные сочетания авторов и жанров. Каждому сочетанию случайным образом поставить в соответствие числа от 0 до 100, вероятность того, что автор действительно пишет в этом жанре.

```
SELECT name_genre, name_author,  
       ROUND(RAND() * 100) AS Вероятность  
FROM  
       genre, author  
ORDER BY 3 DESC;
```

Перекрестное соединение, пример

author

author_id	name_author
01	Булгаков М.А.
02	Достоевский Ф.М.
03	Есенин С.А.

genre

genre_id	name_genre
001	Роман
002	Поэзия
003	Фантези

Перекрестное соединение, пример

FROM genre, author

author_id	name_author	genre_id	name_genre
01	Булгаков М.А.	001	Роман
01	Булгаков М.А.	002	Поэзия
01	Булгаков М.А.	003	Фантези
02	Достоевский Ф.М.	001	Роман
02	Достоевский Ф.М.	002	Поэзия
02	Достоевский Ф.М.	003	Фантези
03	Есенин С.А.	001	Роман
03	Есенин С.А.	002	Поэзия
03	Есенин С.А.	003	Фантези

Перекрестное соединение, пример

```
SELECT name_genre, name_author
```

name_author	name_genre
Булгаков М.А.	Роман
Булгаков М.А.	Поэзия
Булгаков М.А.	Фантези
Достоевский Ф.М.	Роман
Достоевский Ф.М.	Поэзия
Достоевский Ф.М.	Фантези
Есенин С.А.	Роман
Есенин С.А.	Поэзия
Есенин С.А.	Фантези

Перекрестное соединение, пример

```
SELECT name_genre, name_author, ROUND (RAND () * 100)
```

name_author	name_genre	ROUND (RAND () * 100)
Булгаков М.А.	Роман	23.0
Булгаков М.А.	Поэзия	49.0
Булгаков М.А.	Фантези	78.0
Достоевский Ф.М.	Роман	55.0
Достоевский Ф.М.	Поэзия	83.0
Достоевский Ф.М.	Фантези	15.0
Есенин С.А.	Роман	6.0
Есенин С.А.	Поэзия	39.0
Есенин С.А.	Фантези	70.0

Перекрестное соединение, пример

```
SELECT name_genre, name_author, ROUND(RAND() * 100)  
AS Вероятность
```

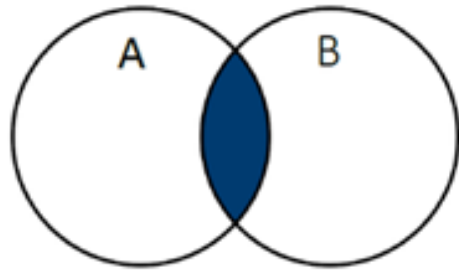
name_author	name_genre	Вероятность
Булгаков М.А.	Роман	23.0
Булгаков М.А.	Поэзия	49.0
Булгаков М.А.	Фантези	78.0
Достоевский Ф.М.	Роман	55.0
Достоевский Ф.М.	Поэзия	83.0
Достоевский Ф.М.	Фантези	15.0
Есенин С.А.	Роман	6.0
Есенин С.А.	Поэзия	39.0
Есенин С.А.	Фантези	70.0

Перекрестное соединение, пример

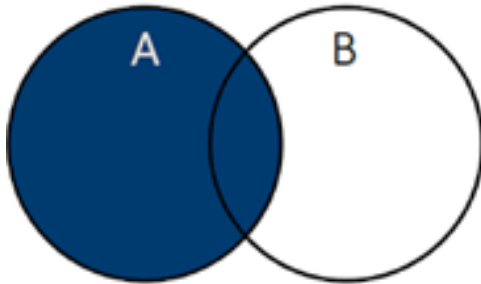
ORDER BY 3 DESC

name_author	name_genre	Вероятность
Достоевский Ф.М.	Поэзия	83.0
Булгаков М.А.	Фантези	78.0
Есенин С.А.	Фантези	70.0
Достоевский Ф.М.	Роман	55.0
Булгаков М.А.	Поэзия	49.0
Булгаков М.А.	Роман	23.0
Есенин С.А.	Поэзия	39.0
Достоевский Ф.М.	Фантези	15.0
Есенин С.А.	Роман	6.0

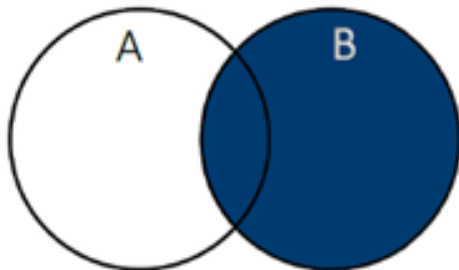
Соединение таблиц, диаграмма Венна



INNER JOIN

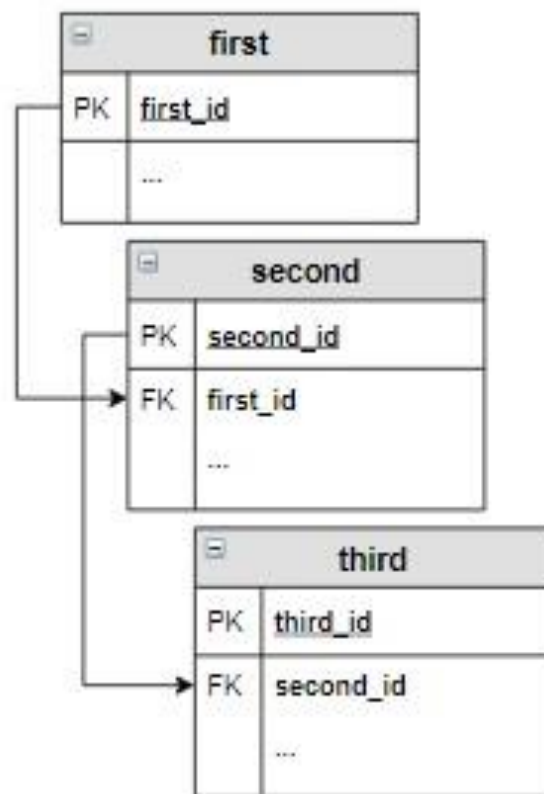


LEFT JOIN

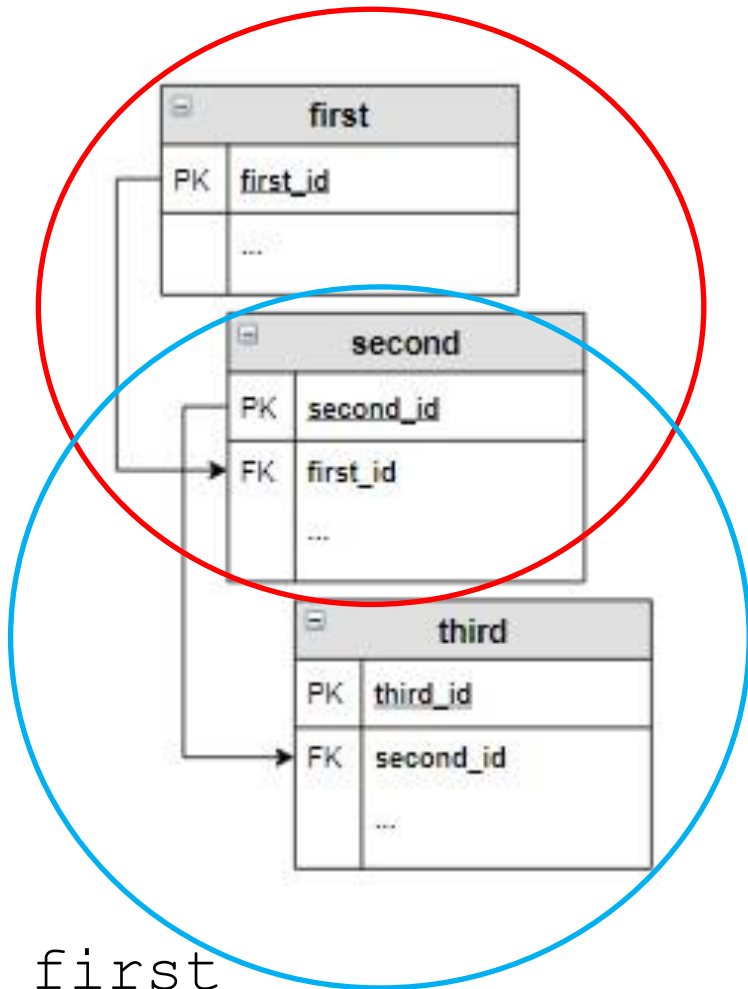


RIGHT JOIN

Соединение нескольких таблиц



Соединение нескольких таблиц

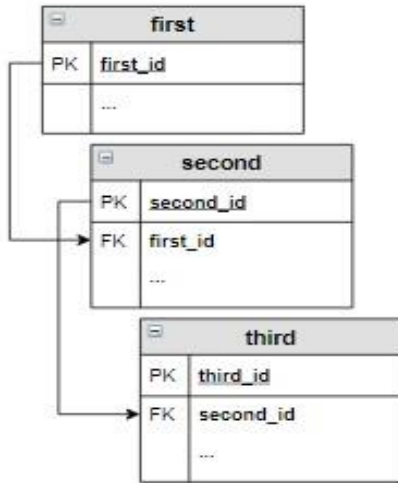


`first`

INNER JOIN `second` **ON** `first.first_id = second.first_id`

INNER JOIN `third` **ON** `second.second_id = third.second_id`

Соединение нескольких таблиц



Вариант 1 (рекомендуемый first -> second -> third)

first

INNER JOIN second **ON** first.first_id = second.first_id

INNER JOIN third **ON** second.second_id = third.second_id

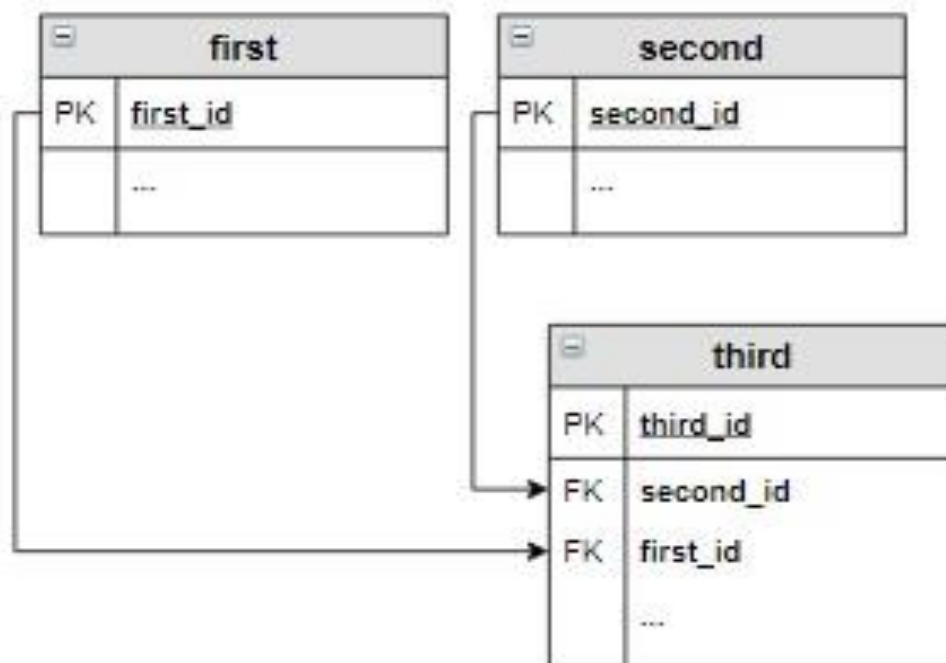
Вариант 2 (third -> second -> first)

third

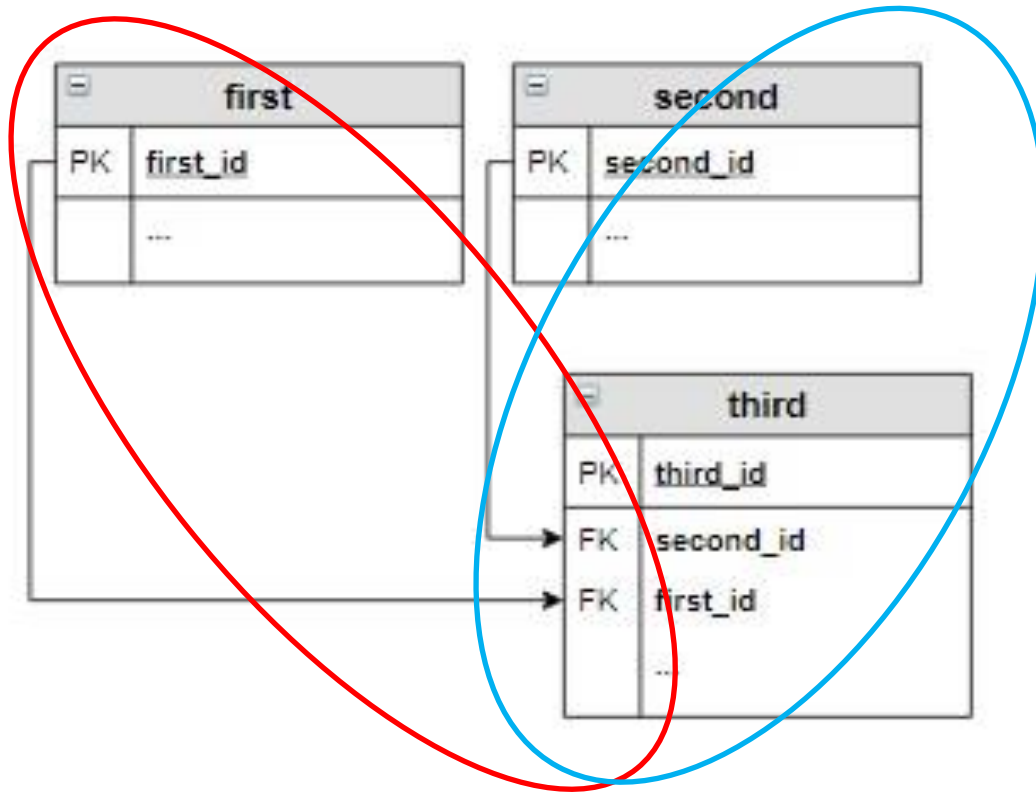
INNER JOIN second **ON** third.second_id = second.second_id

INNER JOIN first **ON** second.first_id = first.first_id

Соединение нескольких таблиц



Соединение нескольких таблиц

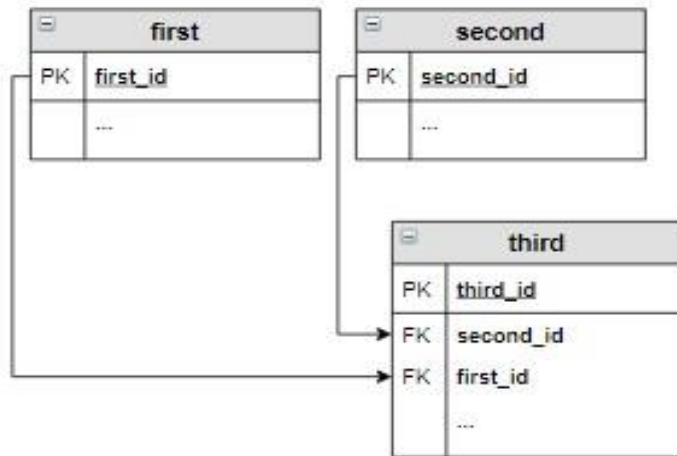


`first`

INNER JOIN `third` **ON** `first.first_id = third.first_id`

INNER JOIN `second` **ON** `third.second_id = second.second_id`

Соединение нескольких таблиц



Вариант 1 (first -> third -> second)

first

INNER JOIN third **ON** first.first_id = third.first_id

INNER JOIN second **ON** third.second_id = second.second_id

Вариант 2 (second -> third -> first)

second

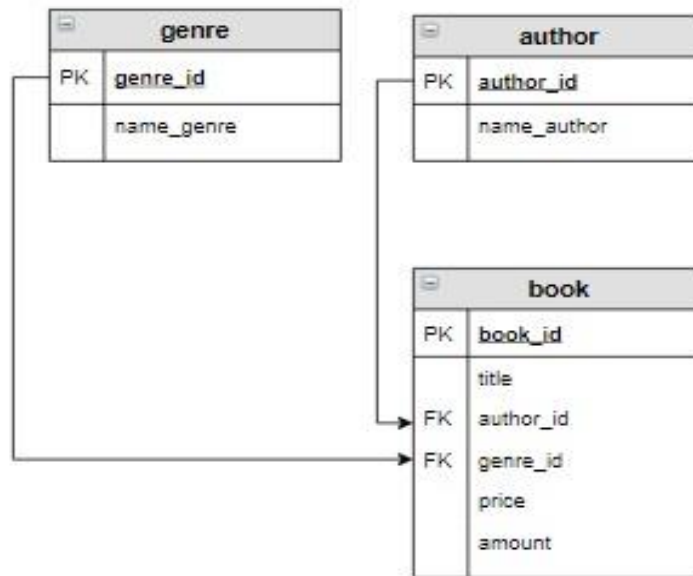
INNER JOIN third **ON** second.second_id = third.second_id

INNER JOIN first **ON** third.first_id = first.first_id

Соединение таблиц, пример

Пример. Вывести информацию о тех книгах, их авторах и жанрах, цена которых принадлежит интервалу от 500 до 700 рублей включительно.

Шаг 1. Отобразить логическую схему и описать связи



author

```
INNER JOIN book ON author.author_id = book.author_id
INNER JOIN genre ON genre.genre_id = book.genre_id
```


Соединение таблиц, пример

Пример. Вывести информацию о тех книгах, их авторах и жанрах, цена которых принадлежит интервалу от 500 до 700 рублей включительно.

Шаг 2. Реализовать запрос

SELECT

title, name_author, name_genre, price, amount

FROM

author

INNER JOIN book **ON** author.author_id = book.author_id

INNER JOIN genre **ON** genre.genre_id = book.genre_id

WHERE price **BETWEEN** 500 **AND** 700;

Соединение таблиц, пример

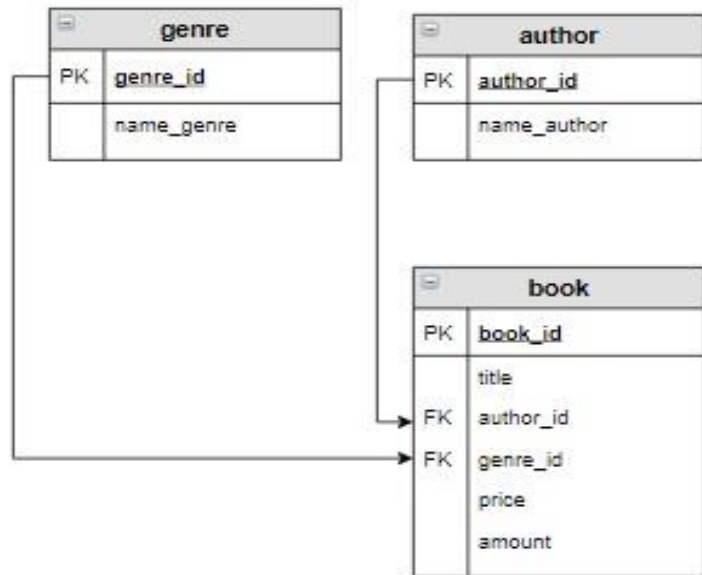
Пример. Вывести информацию о тех книгах, их авторах и жанрах, цена которых принадлежит интервалу от 500 до 700 рублей включительно.

title	name_author	name_genre	price	amount
Мастер и Маргарита	Булгаков М.А.	Роман	670.99	3
Белая гвардия	Булгаков М.А.	Роман	540.50	5
Стихотворения и поэмы	Есенин С.А.	Поэзия	650.00	15

ON и USING

В том случае, если соединение таблиц осуществляется по столбцам, которые имеют одинаковое имя, в условии соединения можно использовать оператор **USING**.

ON vs USING



author

```
INNER JOIN book ON author.author_id = book.author_id
INNER JOIN genre ON genre.genre_id = book.genre_id
```



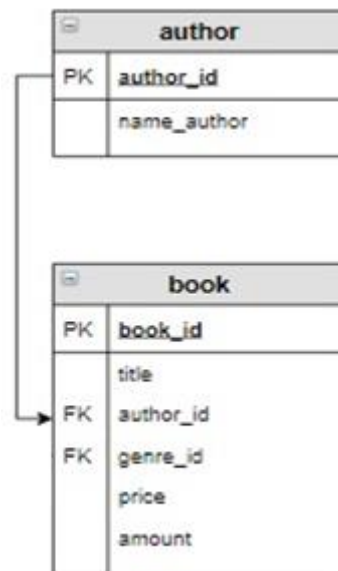
author

```
INNER JOIN book USING (author_id)
INNER JOIN genre USING (genre_id)
```

Соединение таблиц, пример

Пример. Вывести количество различных книг каждого автора. В результат включить и тех авторов, книг которых в данный момент нет на складе.

Шаг 1. Отобразить логическую схему и описать связи



author

LEFT JOIN book **USING** (author_id)

Соединение таблиц, пример

Пример. Вывести количество различных книг каждого автора. В результат включить и тех авторов, книг которых в данный момент нет на складе.

Шаг 2. Реализовать запрос

SELECT

name_author,
COUNT(title) **AS** Количество

FROM

author

LEFT JOIN book **USING**(author_id)

GROUP BY name_author;

Соединение таблиц, пример

Пример. Вывести количество различных книг каждого автора. В результат включить и тех авторов, книг которых в данный момент нет на складе.

name_author	Количество
Булгаков М.А.	2
Достоевский Ф.М.	1
Есенин С.А.	1

При описании внешнего ключа можно указать действия, которые необходимо выполнить для записей зависимой таблицы при удалении и изменении записей главной таблицы.

Для этого используются опции:

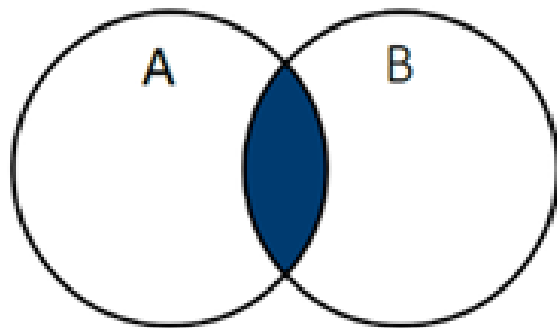
- **ON DELETE:**
 - ✓ **CASCADE** автоматически удаляет строки из зависимой таблицы при удалении связанных строк в главной таблице.
 - ✓ **SET NULL** при удалении связанной строки из главной таблицы устанавливает для столбца внешнего ключа значение **NULL**.
 - ✓ **SET DEFAULT** похоже на SET NULL за тем исключением, что значение внешнего ключа устанавливается не в NULL, а в значение по умолчанию для данного столбца.
 - ✓ **RESTRICT** отклоняет удаление строк в главной таблице при наличии связанных строк в зависимой таблице.

Соединение двух таблиц заключается в объединении каждой строки первой таблицы с каждой строкой второй и проверки условия соединения.

Типы соединений:

- ✓ внутренне соединение (**INNER JOIN**) ;
- ✓ внешнее соединение (**LEFT JOIN, RIGHT JOIN**);
- ✓ перекрестное соединение (**CROSS JOIN**).

Внутреннее соединение определяет те записи из двух таблиц, для которых условие соединения – истина.



Внутреннее соединение определяет те записи из двух таблиц, для которых условие соединения – истина.

Синтаксис:

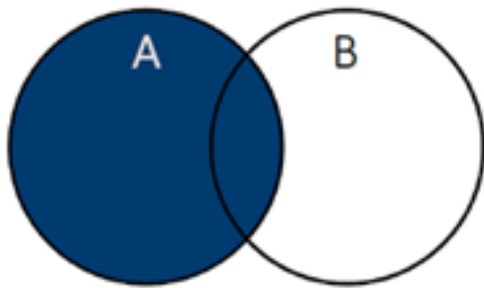
```
таблица_1  
INNER JOIN таблица_2 ON таблица_1.поле = таблица_2.поле
```

Если имена связанных полей одинаковы:

```
таблица_1  
INNER JOIN таблица_2 USING (поле)
```

Внешнее соединение (LEFT JOIN) , при его использовании

1. в результат включается внутреннее соединение (**INNER JOIN**) первой и второй таблицы в соответствии с условием;
2. затем в результат добавляются те записи первой таблицы, которые не вошли во внутреннее соединение на шаге 1, для таких записей поля второй таблицы заполняются значениями **NULL**.



Внешнее соединение (LEFT JOIN)

Синтаксис:

```
таблица_1  
LEFT JOIN таблица_2 ON таблица_1.поле = таблица_2.поле
```

Внешнее соединение (LEFT JOIN)

Синтаксис:

```
таблица_1  
LEFT JOIN таблица_2 ON таблица_1.поле = таблица_2.поле
```

Если имена связанных полей одинаковы:

```
таблица_1  
LEFT JOIN таблица_2 USING (поле)
```

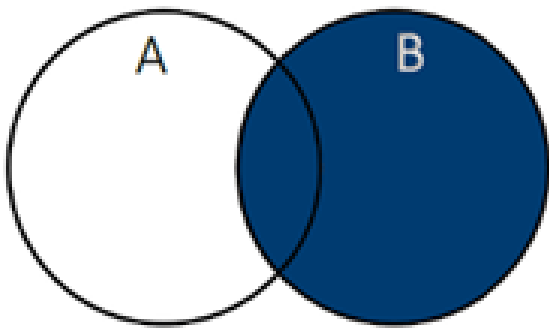
Внешнее соединение (RIGHT JOIN) , при его использовании

Внешнее соединение (RIGHT JOIN) , при его использовании

1. в результат включается внутреннее соединение (**INNER JOIN**) первой и второй таблицы в соответствии с условием;

Внешнее соединение (RIGHT JOIN) , при его использовании

1. в результат включается внутреннее соединение (**INNER JOIN**) первой и второй таблицы в соответствии с условием;
2. затем в результат добавляются те записи второй таблицы, которые не вошли во внутреннее соединение на шаге 1, для таких записей поля первой таблицы заполняются значениями **NULL**.



Внешнее соединение (RIGHT JOIN)

Синтаксис:

```
таблица_1  
RIGHT JOIN таблица_2 ON таблица_1.поле = таблица_2.поле
```

Если имена связанных полей одинаковы:

```
таблица_1  
RIGHT JOIN таблица_2 USING (поле)
```

При использовании **перекрестного соединения** **CROSS JOIN** результат запроса формируется так:

- каждая строка одной таблицы соединяется с каждой строкой другой таблицы, в результате получаются все возможные сочетания строк двух таблиц.

Перекрестное соединение (CROSS JOIN)

Синтаксис:

таблица_1
CROSS JOIN таблица_2

или

таблица_1, таблица_2

В соединение может входить любое количество таблиц. В этом случае

- сначала описывается соединение двух таблиц,
- затем описывается соединение между какой-либо таблицей, входящей в первую связь, и следующей таблицей.



Спасибо за внимание!