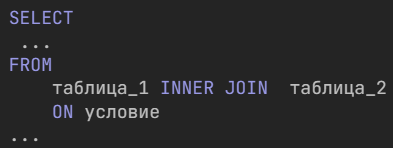
Оператор внутреннего соединения INNER JOIN соединяет две таблицы.

Порядок таблиц для оператора неважен, поскольку оператор является симметричным. Имеется в виду что неважно какая из соединяемых таблиц стоит слева а какая справа от join-a, результат одинаковый.



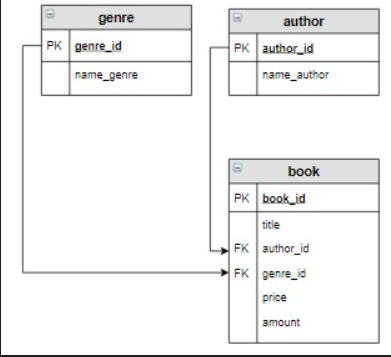
Результат запроса формируется так:

каждая строка одной таблицы соединяется с каждой строкой второй таблицы;

для полученной «соединённой» строки проверяется условие соединения;

если условие истинно, в таблицу результата добавляется соответствующая «соединённая» строка;

Пример: Вывести название книг и их авторов.

В данном запросе осуществляется соединение главной таблицы **author** и зависимой таблицы **book** по ключевому столбцу author.author\_id и внешнему ключу book.author\_id. При этом в результирующую таблицу запроса включаются все строки, в которых значения этих столбцов совпадают. Другими словами строки зависимой таблицы book дополняются фамилией и инициалами авторов из таблицы author.

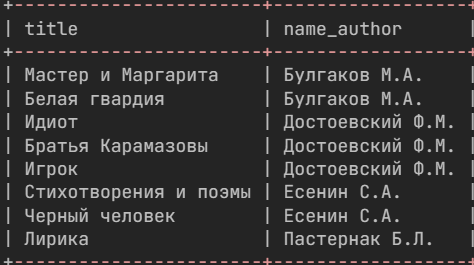
**SELECT title, name\_author**

**FROM author INNER JOIN book** ← соединяем 2 таблицы

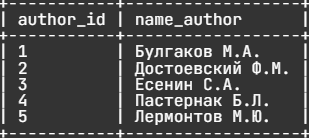
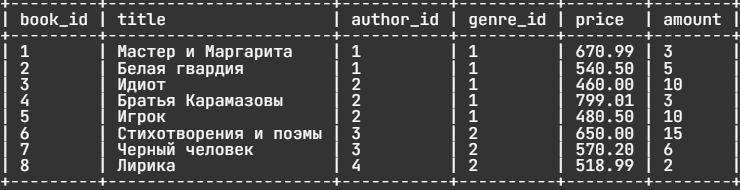
**ON**

**author.author\_id = book.author\_id**

**Результат**



**author book**

Как сказано чуть выше, берётся первая строка из каждой таблицы

и если **author.author\_id = book.author\_id** то выводится то что указано в **SELECT** , потом видимо строка из главной таблицы остаётся, а строка из подчинённой переключается на следующую и если там **author.author\_id = book.author\_id** то опять же выводим. Наверно строка из главной таблицы остаётся пока мы не прошли всю подчинённую таблицу, затем строка в главной таблице переключается на следующую и всё повторяется.

**Ключевая особенность этого JOIN-a, если в главной таблице(левой / первой) есть значение а в подчинённой / правой / второй значения в принципе нет, т.е. сравнить даже нечего, то такая строка НЕ ВЫВОДИТСЯ вообще. Для сравнения в LEFT JOIN вывелось бы значение из левой таблицы а в правое значение подставился бы NULL.**

Пример: Вывести название, жанр и цену тех книг, количество которых больше 8, в отсортированном по убыванию цены виде

**SELECT title, name\_genre, price**

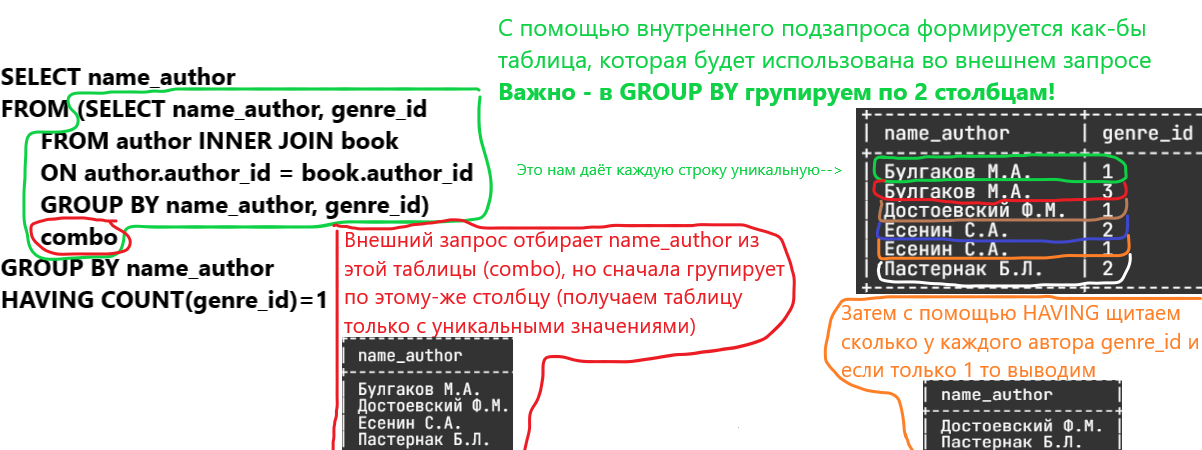
**FROM genre INNER JOIN book**

**ON genre.genre\_id = book.genre\_id**

**WHERE book.amount>8**

**ORDER BY book.price DESC**

Пример: Вывести в алфавитном порядке всех авторов, которые пишут только в одном жанре.



Из интересного:

- Есть вложенный подзапрос у которого есть JOIN

- у вложенного подзапроса есть имя ‘combo’ это важно, иначе FROM не сработает.

- в вложенном подзапросе группируем по 2 столбцам

inner join

An INNER JOIN permits rows from either table to appear in the result if and only if both tables meet the conditions

specified in the ON clause.

SELECT 'nazvani\_colonok' FROM 'nazvanie\_tablici' INNER JOIN 'nazvanie\_vtoroi\_tablici' ON 'nazvanie\_kolonki\_pervoi\_tablici'=

'nazvanie\_kolonko\_vtoroi\_tablici';

Naprimer:

SELECT distinct product.maker, laptop.speed FROM product inner join laptop ON product.model=laptop.model WHERE laptop.hd>=10

left join

snachala vivodit mnozzistvo s levoi tablici (vse) a zatem s pravogo mnozzestva no tolko te kotorie sovpodayt s ysloviem

right join

tozze chto i left no naoborot.