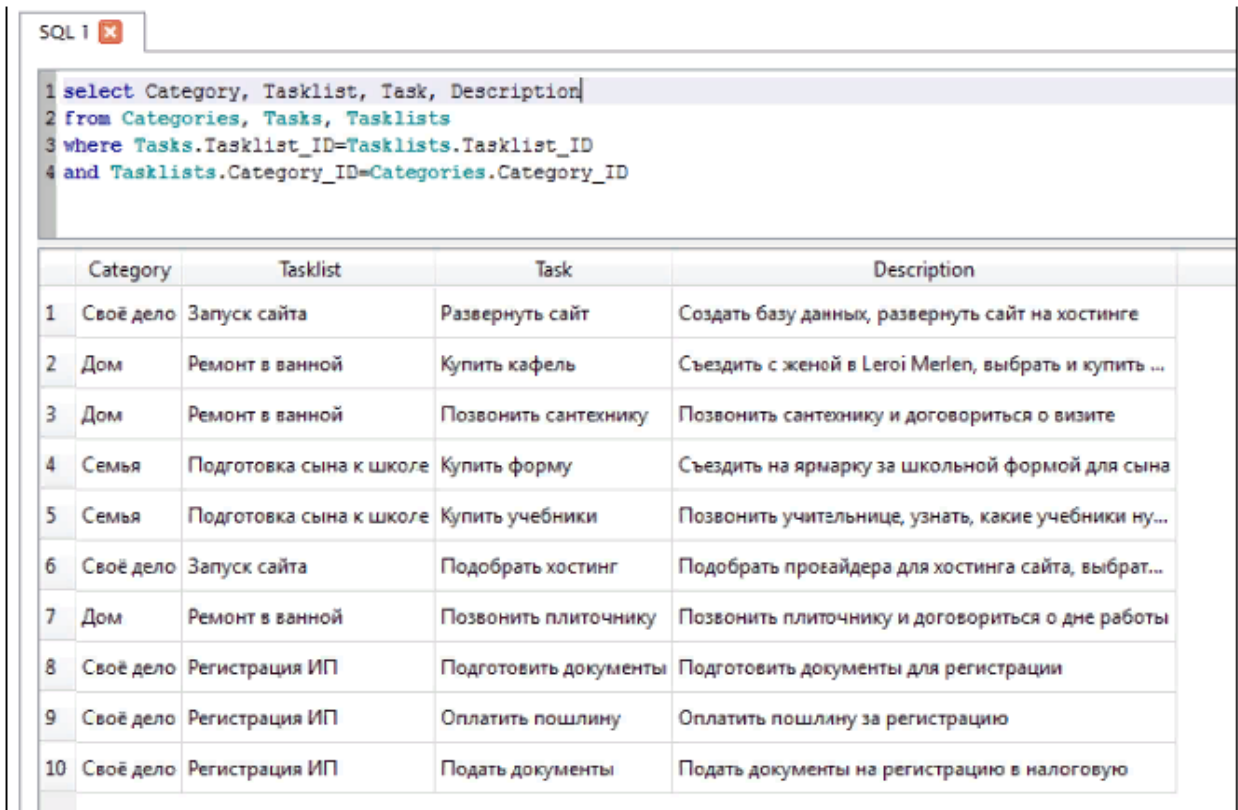


Запросы с сортировкой

В предыдущем примере мы получили перечень всех задач с указанием списка, в который они входят, а также категории, к которой относится этот список.



The screenshot shows a SQL query editor window with a tab labeled 'SQL 1'. The query is as follows:

```
1 select Category, Tasklist, Task, Description
2 from Categories, Tasks, Tasklists
3 where Tasks.Tasklist_ID=Tasklists.Tasklist_ID
4 and Tasklists.Category_ID=Categories.Category_ID
```

Below the query, the results are displayed in a table with the following columns: Category, Tasklist, Task, and Description. The results are sorted by Tasklist.

	Category	Tasklist	Task	Description
1	Своё дело	Запуск сайта	Развернуть сайт	Создать базу данных, развернуть сайт на хостинге
2	Дом	Ремонт в ванной	Купить кафель	Съездить с женой в Leroi Merlen, выбрать и купить ...
3	Дом	Ремонт в ванной	Позвонить сантехнику	Позвонить сантехнику и договориться о визите
4	Семья	Подготовка сына к школе	Купить форму	Съездить на ярмарку за школьной формой для сына
5	Семья	Подготовка сына к школе	Купить учебники	Позвонить учительнице, узнать, какие учебники ну...
6	Своё дело	Запуск сайта	Подобрать хостинг	Подобрать провайдера для хостинга сайта, выбрат...
7	Дом	Ремонт в ванной	Позвонить плиточнику	Позвонить плиточнику и договориться о дне работы
8	Своё дело	Регистрация ИП	Подготовить документы	Подготовить документы для регистрации
9	Своё дело	Регистрация ИП	Оплатить пошлину	Оплатить пошлину за регистрацию
10	Своё дело	Регистрация ИП	Подать документы	Подать документы на регистрацию в налоговую

Но результат запроса выглядит не очень красиво, и его неудобно читать. Категории и списки «размазаны» по строкам, и хочется их сгруппировать, чтобы работать со списком было удобнее.

SQL предоставляет такую возможность с помощью ещё одного ключевого слова: **ORDER BY** (сортировать по). Следующий запрос отсортирует результаты по названию списка задач.

```
select Category, Tasklist, Task, Description
from Categories, Tasks, Tasklists
where Tasks.Tasklist_ID=Tasklists.Tasklist_ID
and Tasklists.Category_ID=Categories.Category_ID
order by Tasklist
```

	Category	Tasklist	Task	Description
1	Своё дело	Запуск сайта	Развернуть сайт	Создать базу данных, развернуть сайт на хостинге
2	Своё дело	Запуск сайта	Подобрать хостинг	Подобрать провайдера для хостинга сайта, выбрат...
3	Семья	Подготовка сына к школе	Купить форму	Съездить на ярмарку за школьной формой для сына
4	Семья	Подготовка сына к школе	Купить учебники	Позвонить учительнице, узнать, какие учебники ну...
5	Своё дело	Регистрация ИП	Подготовить документы	Подготовить документы для регистрации
6	Своё дело	Регистрация ИП	Оплатить пошлину	Оплатить пошлину за регистрацию
7	Своё дело	Регистрация ИП	Подать документы	Подать документы на регистрацию в налоговую
8	Дом	Ремонт в ванной	Купить кафель	Съездить с женой в Leroy Merlen, выбрать и купить ...
9	Дом	Ремонт в ванной	Позвонить сантехнику	Позвонить сантехнику и договориться о визите
10	Дом	Ремонт в ванной	Позвонить плиточнику	Позвонить плиточнику и договориться о дне работы

В обобщённом виде запрос теперь выглядит так:

```
SELECT {список полей} FROM {список таблиц} WHERE {условие отбора} ORDER BY {список полей}
```

Список полей для сортировки может включать несколько полей. При этом результаты запроса сначала сортируются по первому полю в списке. Строки, в которых значения первого поля одинаковы, дополнительно сортируются по второму полю, и так далее, пока не будет выполнена сортировка по всем перечисленным полям.

Например, мы можем отсортировать результаты сначала по названию категории, а внутри этой сортировки — по названию списка задач. Запрос будет выглядеть так:

```
select Category, Tasklist, Task, Description
from Categories, Tasks, Tasklists
where Tasks.Tasklist_ID=Tasklists.Tasklist_ID
and Tasklists.Category_ID=Categories.Category_ID
order by Category, Tasklist
```

Category	Tasklist	Task	Description
1 Дом	Ремонт в ванной	Купить кафель	Съездить с женой в Leroi Merlen, выбрать и купить ...
2 Дом	Ремонт в ванной	Позвонить сантехнику	Позвонить сантехнику и договориться о визите
3 Дом	Ремонт в ванной	Позвонить плиточнику	Позвонить плиточнику и договориться о дне работы
4 Своё дело	Запуск сайта	Развернуть сайт	Создать базу данных, развернуть сайт на хостинге
5 Своё дело	Запуск сайта	Подобрать хостинг	Подобрать провайдера для хостинга сайта, выбрат...
6 Своё дело	Регистрация ИП	Подготовить документы	Подготовить документы для регистрации
7 Своё дело	Регистрация ИП	Оплатить пошлину	Оплатить пошлину за регистрацию
8 Своё дело	Регистрация ИП	Подать документы	Подать документы на регистрацию в налоговую
9 Семья	Подготовка сына к школе	Купить форму	Съездить на ярмарку за школьной формой для сына
10 Семья	Подготовка сына к школе	Купить учебники	Позвонить учительнице, узнать, какие учебники ну...

Сортировка по номерам полей вместо имён

В списке полей вместо названий полей можно использовать их порядковые номера в результатах запроса. В предыдущем примере вместо

`order by category, Tasklist`

можно написать:

`order by 1, 2`

Весь запрос тогда будет выглядеть таким образом:

```
select Category, Tasklist, Task, Description
from Categories, Tasks, Tasklists
where Tasks.Tasklist_ID=Tasklists.Tasklist_ID
and Tasklists.Category_ID=Categories.Category_ID
order by 1, 2
```

Результат его работы не изменится.

	Category	Tasklist	Task	Description
1	Дом	Ремонт в ванной	Купить кафель	Съездить с женой в Leroi Merlen, выбрать и купить ...
2	Дом	Ремонт в ванной	Позвонить сантехнику	Позвонить сантехнику и договориться о визите
3	Дом	Ремонт в ванной	Позвонить плиточнику	Позвонить плиточнику и договориться о дне работы
4	Своё дело	Запуск сайта	Развернуть сайт	Создать базу данных, развернуть сайт на хостинге
5	Своё дело	Запуск сайта	Подобрать хостинг	Подобрать провайдера для хостинга сайта, выбрат...
6	Своё дело	Регистрация ИП	Подготовить документы	Подготовить документы для регистрации
7	Своё дело	Регистрация ИП	Оплатить пошлину	Оплатить пошлину за регистрацию
8	Своё дело	Регистрация ИП	Подать документы	Подать документы на регистрацию в налоговую
9	Семья	Подготовка сына к школе	Купить форму	Съездить на ярмарку за школьной формой для сына
10	Семья	Подготовка сына к школе	Купить учебники	Позвонить учительнице, узнать, какие учебники ну...

Сортировка в обратном порядке

По умолчанию ORDER BY сортирует числовые значения по возрастанию, а текстовые поля по алфавиту. Но этот порядок можно изменить на обратный с помощью ключевого слова DESC, которое добавляется после имени (или номера) поля, по которому выполняется сортировка.

Сравните результат выполнения следующих двух запросов. Первый запрос выбирает из таблицы **Tasks** все краткие описания задач и сортирует их по алфавиту в порядке возрастания.

```
select Task from Tasks order by Task
```

Второй запрос делает то же самое (только вместо имени поля указан его номер), но в выражение ORDER BY добавлено слово DESC.

```
select Task from Tasks order by 1 desc
```

SQL 1	
1 select Task from Tasks order by Task	
Task	
1	Купить кафель
2	Купить учебники
3	Купить форму
4	Оплатить пошлину
5	Подать документы
6	Подготовить документы
7	Подобрать хостинг
8	Позвонить плиточнику
9	Позвонить сантехнику
10	Развернуть сайт

SQL 1	
1 select Task from Tasks order by 1 desc	
Task	
1	Развернуть сайт
2	Позвонить сантехнику
3	Позвонить плиточнику
4	Подобрать хостинг
5	Подготовить документы
6	Подать документы
7	Оплатить пошлину
8	Купить форму
9	Купить учебники
10	Купить кафель

Противоположностью ключевого слова DESC является ключевое слово ASC. Эти слова являются сокращениями от английских слов **ascending** (восходящий) и **descending** (нисходящий).

Слово ASC обычно не указывают, потому что порядок сортировки по возрастанию используется по умолчанию.

Если используется сортировка по нескольким полям, то слова ASC и DESC относятся только к тому полю, после которого они указаны в запросе. Сравните результаты следующих двух запросов, которые отличаются только позицией, в которой стоит слово DESC.

Оба запроса выбирают из таблиц Tasklists и Tasks название списка задач (поле **Tasklist**) и краткое описание задачи (поле **Task**). В обоих запросах для сортировки используются последовательно поля **Tasklist** и **Task**. Но в первом запросе слово DESC стоит после Tasklist, а во втором — после Task.

```
select Tasklist, Task
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.TaskList_ID
order by Tasklist desc, Task
```

```
select Tasklist, Task
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.TaskList_ID
order by Tasklist, Task desc
```

	Tasklist	Task
1	Ремонт в ванной	Купить кафель
2	Ремонт в ванной	Позвонить плиточнику
3	Ремонт в ванной	Позвонить сантехнику
4	Регистрация ИП	Оплатить пошлину
5	Регистрация ИП	Подать документы
6	Регистрация ИП	Подготовить документы
7	Подготовка сына к школе	Купить учебники
8	Подготовка сына к школе	Купить форму
9	Запуск сайта	Подобрать хостинг
10	Запуск сайта	Развернуть сайт

	Tasklist	Task
1	Запуск сайта	Развернуть сайт
2	Запуск сайта	Подобрать хостинг
3	Подготовка сына к школе	Купить форму
4	Подготовка сына к школе	Купить учебники
5	Регистрация ИП	Подготовить документы
6	Регистрация ИП	Подать документы
7	Регистрация ИП	Оплатить пошлину
8	Ремонт в ванной	Позвонить сантехнику
9	Ремонт в ванной	Позвонить плиточнику
10	Ремонт в ванной	Купить кафель

Удаление дубликатов в результатах запроса

Следующий запрос покажет нам, к каким спискам относятся все задачи, присутствующие в таблице **Tasks**.

```
select Tasklist
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.Tasklist_ID
```

```

1 select Tasklist
2 from Tasklists, Tasks
3 where Tasklists.Tasklist_ID=Tasks.Tasklist_ID

```

	Tasklist
1	Запуск сайта
2	Ремонт в ванной
3	Ремонт в ванной
4	Подготовка сына к школе
5	Подготовка сына к школе
6	Запуск сайта
7	Ремонт в ванной
8	Регистрация ИП
9	Регистрация ИП
10	Регистрация ИП

В результате мы получили 10 строк (по числу задач в таблице **Tasks**), часть из которых дублируется. Возникает вопрос: как избавиться от этих дубликатов?

В SQL для этого предназначено ключевое слово **DISTINCT**, которое ставится сразу после ключевого слова SELECT. Это ключевое слово ещё называют *модификатором запроса* SELECT.

Запрос с модификатором DISTINCT будет выглядеть следующим образом.

```

select distinct Tasklist
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.Tasklist_ID

```

```

1 select distinct Tasklist
2 from Tasklists, Tasks
3 where Tasklists.Tasklist_ID=Tasks.Tasklist_ID

```

	Tasklist
1	Запуск сайта
2	Ремонт в ванной
3	Подготовка сына к школе
4	Регистрация ИП

В результате его выполнения, как мы видим, остались только строки с уникальными значениями, а все дубликаты исключены из результата запроса.

Противоположностью модификатора DISTINCT является ключевое слово (или модификатор запроса) **ALL**. Модификатор ALL сообщает движку базы данных, что не нужно удалять дубликаты записей из результатов запроса.

Если в запросе SELECT не указан никакой модификатор, то по умолчанию используется значение ALL, поэтому ключевое слово ALL чаще всего пропускают. Запрос с модификатором ALL выглядел бы следующим образом:

```
select all Tasklist
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.Tasklist_ID
```

Дубликатами считаются строки, в которых совпадают значения всех полей. Если хотя бы в одном поле значения не совпадают, то строки считаются уникальными. Добавим в наш запрос поле Done и посмотрим на результат.

```
select distinct Tasklist, Done
from Tasklists, Tasks
where Tasklists.Tasklist_ID=Tasks.Tasklist_ID
```

```
1 select distinct Tasklist, Done
2 from Tasklists, Tasks
3 where Tasklists.Tasklist_ID=Tasks.Tasklist_ID
```

	Tasklist	Done
1	Запуск сайта	0
2	Ремонт в ванной	0
3	Подготовка сына к школе	0
4	Регистрация ИП	0
5	Регистрация ИП	1

В списке «Регистрация ИП» есть задача, помеченная как выполненная, и несколько невыполненных задач. Поэтому этот список присутствует в результатах запроса в двух строках.

Сортировка и удаление дубликатов — практика

Составьте запрос, возвращающий названия всех категорий покупок, отсортированных по алфавиту.

Составьте запрос к таблице Spendings, возвращающий названия магазинов, даты совершенных покупок и суммы покупок, отсортированные в порядке убывания сумм.

Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке уменьшения.

Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 200 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке уменьшения.

Составьте запрос, возвращающий названия всех магазинов, в которых были совершены покупки, без удаления дубликатов из результатов запроса.

Составьте запрос, возвращающий названия всех категорий, к которым относятся совершённые покупки, без повторяющихся значений в результатах запроса.