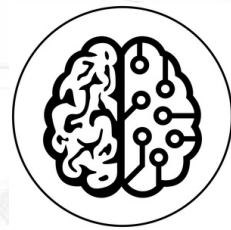


# Таблица поисковых операторов Obsidian



Оператор	Описание	Пример
<code>file:</code>	Поиск текста в имени файла. Ищется любой файл в хранилище	<code>file:.jpg</code> or <code>file:202209</code>
<code>path:</code>	Поиск текст в определенной директории	<code>path:"Daily notes"</code>
<code>content:</code>	Поиск текста в содержимом файла	<code>content:"happy cat"</code>
<code>match-case:</code>	Учитывать регистр текста	<code>match-case:HappyCat</code>
<code>ignore-case:</code>	Игнорировать регистр текста	<code>ignore-case:ikea</code>
<code>line:</code>	Поиск в рамках одной строки	<code>line:(mix flour)</code>
<code>block:</code>	Поиск в рамках одного блока	<code>block:(dog cat)</code>
<code>section:</code>	Поиск в рамках одной секции (текст между двумя заголовками)	<code>section:(dog cat)</code>
<code>task:</code>	Поиск по всем задачам	<code>task:позвонить</code>
<code>task-todo:</code>	Поиск по незавершенным задачам	<code>task-todo:уточнить</code>
<code>task-done:</code>	Поиск по завершенным задачам	<code>task-done:купить</code>

## Поиск по свойствам

Вы можете использовать данные, хранящиеся в Свойствах, в условиях поиска.

Для этого используйте квадратные скобки вокруг имени свойства [property], чтобы найти файлы с этим свойством: `[aliases]` ищет все файлы файлы, содержащие свойство aliases

Чтобы найти файлы со свойством и значением, используйте скобки и двоеточие:

`[aliases:Name]` ищет файлы, в которых значение свойства aliases равно Name

И свойство, и значение позволяют использовать подзапросы, такие как круглые скобки для группировки, оператор OR, двойные кавычки для точного соответствия и regex.

Пример: `[status:Draft OR Published]` ищет файлы, у которых значение свойства status равно Draft или Published