

Search Help

Wildcards in Queries

Use wildcard matching sparingly, because it slows server performance. You can use these wildcards in queries:

- ? to match one character.
- * to match zero, one or more characters..

Note: Do not use wildcards within an exact phrase search.

Exact Phrase Search

If you submit a phrase in quotation marks, the server matches the phrase in its exact pre-stemmed form. For example:

“fresh and lovely”

The server removes any stop words that the query contains (the example query above contains the stop word *and*) but does not apply stemming. It is as if the query were:

“fresh lovely”

When it matches the query, the server returns only documents that contain a phrase that matches the phrase in the query string. The query “fresh and lovely” returns only documents that contain a phrase that matches the phrase *fresh lovely* (for example, *fresh lovely*, *fresh and lovely*, *fresh or lovely*, and so on.)

Boolean Search Operators

Operator	Explanation
No operator used	<p>Default operator is AND if none provided. Ensures that every document that returns contains both terms. For example:</p> <p>cat dog</p> <p>This query returns only documents that contain both <i>cat</i> and <i>dog</i>.</p>
AND	<p>Binary operator. This is the default behaviour if no explicit operator is given between two terms. Ensures that every document that returns contains both terms. For example:</p> <p>cat AND dog</p> <p>This query returns only documents that contain both <i>cat</i> and <i>dog</i>.</p>
NOT	<p>Unary operator. Ensures that the term following NOT is excluded from any of the returned documents. For example:</p> <p>cat NOT dog</p> <p>This query returns only documents that contain <i>cat</i> but not <i>dog</i>.</p> <p><u>Note:</u> NOT applies only to the term that immediately follows it. To exclude multiple terms, place them in brackets. To exclude a phrase, put the phrase in quotation marks and in brackets. For example:</p> <p>Doc 1: I went to Old York Town for the New Year. Doc 2: I went to New York City for the New Year.</p> <p>The following query does not match either of these documents: year NOT (Town OR City)</p> <p>The following query matches both documents: year NOT (Town AND City)</p> <p>The following query matches the first document but not the second: year NOT ("New York")</p>
OR	<p>Binary operator. One or both terms must appear for the document to return. For example:</p> <p>cat OR dog</p> <p>This query returns only documents that contain either <i>cat</i>, <i>dog</i> or both terms.</p>
EOR or XOR	<p>Binary operator. Logical exclusive OR. Only one of the terms is permitted to appear for the document to return. This operator is rarely used. For example:</p> <p>cat XOR dog</p> <p>This query returns only documents that contain either the term <i>cat</i> or the term <i>dog</i>. Documents that contain both <i>cat</i> and <i>dog</i> do not return.</p>
()	<p>Bracketed expressions. These expressions are evaluated left to right and can be nested. They dictate the precedence and behaviour of combined operator statements. For example:</p> <p>(cat EOR dog) AND (fish EOR bird)</p> <p>This query returns only documents that contain one of these combinations: <i>cat</i> and <i>fish</i> <i>cat</i> and <i>bird</i> <i>dog</i> and <i>fish</i> <i>dog</i> and <i>bird</i></p>

Proximity Search Operators

Operator	Explanation
NEARN	<p>Returns only documents in which the second term is within <i>N</i> words of the first term – that is, the terms are <i>N</i> or fewer words apart. If you do not specify <i>N</i>, NEAR defaults to 5. For example:</p> <p>red NEAR1 green</p> <p>This query returns only documents in which the term <i>red</i> is adjacent to the term <i>green</i>. For example, documents that contain <i>red green</i> or <i>green red</i> return. Documents that only contain <i>red orange green</i> do not return (because the terms are not close enough to each other).</p>
DNEARN	<p>Directed NEAR. Returns only documents in which the second term is within <i>N</i> words of the first term, in the specified order. If you do not specify <i>N</i>, DNEAR defaults to 5. For example:</p> <p>red DNEAR2 green</p> <p>This query returns only documents in which the term <i>green</i> follows the term <i>red</i>, and is no more than two words away from the term <i>red</i>. For example documents that contain <i>red orange green</i> return, while documents that contain <i>green orange red</i> or <i>red orange blue green</i> do not return.</p>
WNEARN	<p>Weighted NEAR (with OR operation). This proximity operator returns documents that contain either of the two terms. It promotes relevance when the terms are <i>N</i> or fewer words apart (closer together implies higher relevance). If you do not specify <i>N</i>,WNEAR defaults to 5. For example:</p> <p>dog WNEAR7 cat</p> <p>This query returns documents that contain either <i>dog</i> or <i>cat</i>. It gives extra relevance to documents in which <i>dog</i> and <i>cat</i> appear seven or fewer words apart in a piece of text. This weight increases as the terms get closer to each other. Documents in which the terms occur more than seven words apart, or in which only one term occurs, return with normal relevance.</p>
YNEARN	<p>Weighted NEAR (with AND operation). This proximity operator returns documents that contain both of the terms. It promotes relevance when the terms are <i>N</i> or fewer words apart (closer together implies higher relevance). If you do not specify <i>N</i>,YNEAR defaults to 5. For example:</p> <p>dog YNEAR7 cat</p> <p>This query returns documents that contain both <i>dog</i> and <i>cat</i>. It gives extra relevance to documents in which <i>dog</i> and <i>cat</i> appear seven or fewer words apart in a piece of text. This weight increases as the terms get closer to each other. Documents in which the terms occur more than seven words apart return with normal relevance.</p>
BEFORE	<p>Returns only documents in which the first term precedes the second one. For example:</p> <p>red BEFORE green</p> <p>This query returns only documents in which the term <i>green</i> appears later than the term <i>red</i>.</p>
AFTER	<p>Returns only documents in which the first term appears later than the second one. For example:</p> <p>red AFTER green</p> <p>This query returns only documents in which the term <i>red</i> appears later than the term <i>green</i>.</p>
XNEAR	<p>Returns only documents in which the second term is exactly <i>N</i> words from the first term. For example:</p> <p>cat XNEAR2 dog</p> <p>This query returns only documents in which the term <i>dogs</i> follows the term <i>cats</i> and is exactly two words away from the term <i>cats</i>. This means that documents which contain <i>cats</i> and <i>dogs</i> return, while documents that contain <i>dogs and cats</i> or <i>cats, dogs</i> do not return.</p>