

# Full-Text Searching Help

## Boolean Searching

A boolean search request consists of a group of words or phrases linked by connectors.

apple and pear	Both words must be present.
apple or pear	Either word can be present.
NEAR((apple, pear),5)	<i>Apple</i> must occur within 5 words of <i>pear</i> .
apple NOT pear	<i>Only apple must be present</i>

## AND Connector

Use the AND connector in a search request to connect two expressions, both of which must be found in any document retrieved. For example:

*apple pie* and *poached pear* would retrieve any document that contained both phrases.

## OR Connector

Use the OR connector in a search request to connect two expressions, at least one of which must be found in any document retrieved. For example, *apple pie* or *poached pear* would retrieve any document that contained *apple pie*, *poached pear*, or both.

## Proximity Searching with number

Use the NEAR connector in a search request to specify that one word or phrase must occur within a certain number of words of the other. For example, NEAR((apple, pear),5) would retrieve any document that contained apple within 5 words of pear.

## NOT Operator

Use NOT in front of any search expression to reverse its meaning. This allows you to exclude documents from a search. Example:

apple NOT pear  
(apple and banana) NOT pear

## Wildcards (% and \_)

A search word can contain the wildcard characters % and \_. A \_ in a word matches any single character, and a % matches any number of characters. The wildcard characters can be in any position in a word. For example:

*appl%* would match apple, application, etc.  
*%cipl%* would match principle, participle, etc.  
*appl\_* would match apply and apple but not apples.  
*ap%ed* would match applied, approved, etc.  
*grant%* would also find granting, granted, and grants.

## **TRUNCATION operator (\$ )**

Truncation(\$) extends a search to cover grammatical variations on a word. For example, a search for *fish* would also find *fishing*. A search for *applied* would also find *applying*, *applies*, and *apply*.

If you want to add Truncation, add a \$ at the beginning of words that you want stemmed in a search.

Example:

*\$grant* would also find *granting*, *granted*, and *grants*.

*\$applied* would also find *applying*, *applies*, and *apply*.