Quick Reference

This chapter is a convenient reference for common search-related request parameters. It is assumed that you have already read the related material in the book and are just looking for something to jog your memory.

You can find an electronic PDF version of this chapter at `http://www.solrenterprisesearchserver.com`. Having it printed makes it quite convenient.

The third column indicates whether a parameter can be specified only once (single) or multiple times (multi).

Core search

The following parameters are commonly used in most search queries. These are also covered in *Chapter 5, Searching*:

Parameter	Description	Single/ multi
`qt=/select`	A named request handler.	single
`q`	The query string. Usually, as entered by an end user.	single
`defType=lucene`	The query parser for `q`. The recommended one is `edismax` (or `dismax`).	single
`fq`	A filter query.	multi
`start=0`	The index into the search results to start returning documents.	single
`rows=10`	The number of search result document rows to return.	single
`fl=*`	The field list to retrieve, comma separated. To get scores: `*,score`	multi

Parameter	Description	Single/ multi
`sort=score desc`	The sort order. A comma-separated list with `asc` or `desc`.	single
`wt=xml`	The writer type for the response format. One of `xml`, `json`, `python`, `php`, `phps`, `ruby`, `javabin`, `csv`, `xslt`, or `velocity`.	single

Other parameters are: `version=2.2`, `omitHeader=off`, and `timeAllowed=-1`.

Diagnostic

Diagnostic parameters covered in *Chapter 5*, *Searching*, are: `indent=off`, `debugQuery=off`, `explainOther` (a query for one doc), `debug.explain.structured=off`, `echoParams=explicit` (`none`/`explicit`/`all`), and `echoHandler=off`.

Use `wt=xslt&tr=example.xsl&debugQuery=true&fl=*,score`.

The Lucene query parser

The following table shows parameters for the Lucene query parser. These are covered in *Chapter 5*, *Searching*:

Parameter	Description	Single/multi
`df`	The default field to search.	single
`q.op=OR`	The default query operator. One of `AND` or `OR`.	single

The DisMax query parser

The following table shows parameters for the DisMax query parser. These are covered in *Chapter 5*, *Searching*, and *Chapter 6*, *Search Relevancy*:

Parameter	Description	Single/multi
`q.alt`	This is an alternate query to run when `q` is absent. The recommended one is: `*:* (all docs)`	single

Parameter	Description	Single/multi
qf	This stands for query fields, including optional boosts, for example, id^5.0 name^2.0 body.	multi
mm=100%	This is the min-should-match specification. It is used to change to all-optional, use 0%	single
qs=0	This is the query slop for phrases explicitly in the query string.	single
pf	This stands for phrase fields for automatic phrase boosting. This is same as qf syntax.	single
ps=0	This is the phrase slop for pf.	single
tie=0	This is the score tie-breaker. The recommended one is 0.1.	single
bq	This is a boost query. The boost is added.	multi
bf	This is a boost function. The boost is added.	multi
boost	This is a boost function. The boost is multiplied. Works for edismax only.	multi

The other edismax additions are lowercaseOperators=on, pf2, pf3, ps2, ps3, stopwords=on, and uf.

The Lucene query syntax

Lucene query syntax, covered in *Chapter 5*, *Searching*, has the following Boolean operators: AND, OR, NOT, &&, and ||with leading + or -. Here is an example:

```
{!lucene df=title q.op=$myop} "phrase query slop"~2 w?ldcard*
fuzzzy~0.7 -(updatedAt:[* TO NOW/DAY-2YEAR] +boostMe^5)
```

Faceting

The following parameters are commonly used in facet queries, and are covered in *Chapter 7*, *Faceting*:

- **Field specific parameter**: (Works for highlighting too) f.myfieldname.facet.mincount=1
- **Field value faceting**: facet=on, facet.field=myfieldname, facet.sort=count (count, index), facet.limit=100, facet.offset=0, facet.mincount=0, facet.missing=off, facet.prefix, facet.method (enum, fc, or fcs)

- **Range faceting**: `facet=on`, `facet.range=myfieldname`, `facet.range.start`, `facet.range.end`, `facet.range.gap` (for example, `+1DAY`), `facet.range.hardend=off`, `facet.range.other=off`, `facet.range.include=lower` (`lower` `upper`, `edge`, `outer`, or `all`)
- **Facet queries**: `facet=on, facet.query`
- **Facet pivots**: `facet.pivot=field1,field2,field3`
- **Facet keys**: `facet.field={!key=Type}r_type`
- **Filter exclusion**: `fq={!tag=r_type}r_type:Album&facet.field={!ex=r_type}r_type`

Highlighting

The following parameters are applicable to the highlighting component, covered in *Chapter 8*, *Search Components*: `hl=off`, `hl.fl`, `hl.requireFieldMatch=off`, `hl.usePhraseHighlighter=off`(the recommended one is `on`), `hl.highlightMultiTerm=off`, `hl.snippets=1`, `hl.fragsize=100`, and `hl.mergeContiguous=off`.

Spell checking

These parameters are applicable to the spellcheck component, detailed in *Chapter 8*, *Search Components*: `spellcheck=off`, `spellcheck.dictionary=default`, `spellcheck.q` (alternative to `q`), `spellcheck.count=1`, `spellcheck.onlyMorePopular=off`, `spellcheck.extendedResults=off`, `spellcheck.collate=off`, `spellcheck.maxCollations=1`, `spellcheck.maxCollationTries=0`, `spellcheck.maxCollationEvaluations=10000`, and `spellcheck.collateExtendedResults=off`.

Miscellaneous nonsearch

- **Commit**: `/update?commit=true (optimize=true to optimize)`
- **Delete**: `/update?stream.body=<delete><query>*:*</query></delete>`
- **Reload config**: `/admin/cores?action=RELOAD&core=mycorename`