Sort Clauses reference

Intro

Sort Clauses are the *sorting options* for Content and Location Search in eZ Platform. For generic use of API Search see Search Criteria and Sort Clauses.

A Sort Clause consists of two parts just like Criterion and FacetBuilder:

- The API Value: `SortClause`
- Specific handler per search engine: `SortClausesHandler`

The `SortClause` represents the value you use in the API, while `SortClauseHandler` deals with the business logic in the background, translating the value to something the Search engine can understand.

> Implementation and availability of a handler sometimes depends on search engine capabilities and limitations.

Common concepts for all Sort Clauses

For how to use each and every Sort Clause, see list below as it depends on the Sort Clause Value constructor, but *in general* you should be aware of the following common concept:

- `sortDirection`: The direction to perform the sort, either `Query::`SORT_ASC *(default)* or `Query::` SORT_DESC

List of Sort Clauses

The list below reflects Sort Clauses available in the `eZ\Publish\API\Repository\Values\Content\Query\SortClause` namespace (it is also possible to make a custom Sort Clause):

> Arguments starting with "`?`" are optional.

Only for LocationSearch

Sort Clause	Constructor arguments description
`Location\Depth` **>= 5.3/2014.03**	`?sortDirection`
`Location\Id` **>= 5.3/2014.03**	`?sortDirection`
`Location\IsMainLocation` **>= 5.3/2014.03**	`?sortDirection`
`Location\Depth` **>= 5.3/2014.03**	`?sortDirection`
`Location\Priority` **>= 5.3/2014.03**	`?sortDirection`
`Location\Visibility` **>= 5.3/2014.03**	`?sortDirection`

Only for ContentSearch

Sort Clause	Constructor arguments description
`LocationDepth` **DEPRECATED IN 5.3/2014.03**	Use LocationSearch with `Location\Depth`, since `LocationDepth` can give unpredictable results when content has several Locations.

`LocationPath` **DEPRECATED IN 5.3/2014.03**	
`LocationPathString` **DEPRECATED IN 5.3/2014.03**	
`LocationPriority` **DEPRECATED IN 5.3/2014.03**	Use LocationSearch with `Location\Priority`, since `LocationPriority` can give unpredictable results when content has several Locations.

Common

Sort Clause	Constructor arguments description
`ContentId`	`?sortDirection`
`ContentName`	`?sortDirection`
`DateModified`	`?sortDirection`
`DatePublished`	`?sortDirection`
`Field`	`typeIdentifier` as string, `fieldIdentifier` as string, `?sortDirection`, `?languageCode` **>= 5.3/2014.01** as string
`MapLocationDistance` **>= 5.3/2014.03**	`typeIdentifier` as string , `fieldIdentifier` as string , `latitude` as float , `longitude` as float, `?sortDirection`, `?languageCode` as string
`SectionIdentifier`	`?sortDirection`
`SectionName`	`?sortDirection`

Related topics:

- Search Criteria and Sort Clauses
- Criteria reference