

Search Tuning Rules

Search tuning rules allow you to have granular control over the products that appear, and the order they appear in, in search results and on category pages.

For example, you have a dress category page on your storefront. Whenever a shopper performs a search for "red" on the dress category page, you want to promote specific dresses to the top in a specific order, and you want to block specific dresses on the search results. You can use search tuning rules to accomplish this behavior. You can also use search tuning rules to sequence products in a specific order on a category page.

Search tuning rules define the products that you want to promote, as well as any products that you want to block in the search results and on category pages. Search tuning rules also define specific keywords that trigger the rule, as well as filters, or category pages, that can always have the rule be applied to.

Search Tuning Rules and the API

Use the [GetSearchTuningRules](#), [UpdateSearchTuningRule](#), [AddSearchTuningRule](#), and the [DeleteSearchTuningRule](#) operations of the [commerce/catalog/admin/search](#) resource to interact with the search tuning rule.

Search Tuning Rule Context

Search tuning rules include a context that specifies when the search tuning rule triggers. For example, you can make the rule trigger when either a shopper searches for a keyword on your storefront, when a shopper navigates to a category page (filter), or when a shopper searches for a keyword in a specific category page.

Specifying Only Search Keywords

When you specify only `keywords` and no `filters`, then the search tuning rule will only trigger whenever a shopper performs a sitewide search for any keywords you specified. If a shopper searches for the keyword on a category page, then the search tuning rule will not trigger and will not be in effect.

For example, you specify `"keywords":["red"]` in the search tuning rule. A shopper then goes to your site and performs a search for red in your entire site. The search tuning rule then triggers and promotes and/or blocks the products you specified.

Specifying Only Categories

When you specify only `filters` and no `keywords`, then the search tuning rule will only trigger whenever a shopper navigates to the specified category pages. If a shopper searches for a keyword on a category page or your entire site, then the search tuning rule will not trigger and will not be in effect.

For example, you specify `"filters":[{"field": "categoryCode", "value": "9"}]` in the search tuning rule. A shopper then goes to the dresses category, which has the `categoryCode` of 9, on your site. The search tuning rule then triggers and promotes and/or blocks the products you specified.

The `filters.value` field correlates to the `categoryCode` field of the category you want to select. Depending on the

id of the category, this field can either be a word or a number. Kibo defaults the `categoryCode` field to the same value as the `id` of the category; however, you can change this depending on your needs. For example, you could have a dresses category whose id is set to `"id": 8` and `categoryCode` set to `"categoryCode": "dresses"`.

Specifying Search Keywords and Categories

When you specify both `keywords` and `filters`, then the search tuning rule will only trigger whenever a shopper performs a search for any of the keywords you specified in any of the categories you specified.

If a shopper performs a sitewide search for any of the keywords, then the search tuning rule will not trigger. Likewise, if a shopper goes to any of the category pages you specified, then the search tuning rule will not trigger.

For example, you specify `"keywords":["red"],"filters":[{"field": "categoryCode","value": "9"}]` in the search tuning rule context. A shopper then goes to your site and performs a search for red in the dresses category page, which has the `categoryCode` of 9. The search tuning rule then triggers and promotes and/or blocks the products you specified.

Enabling Sitewide Search Keywords and Categories

In order for search tuning to trigger when a shopper performs a sitewide keyword search, or when a shopper navigates to a category page, you must create two separate search tuning rules that target the two different contexts. One search tuning rule should specify the keywords you want to trigger the rule, and none of the filters. The other search tuning rule should specify the filters you want to trigger and none of the search keywords.

For example, you want search tuning to be in effect whenever a shopper performs a sitewide search for red and well as when they navigate to the dresses category page. You create one search tuning rule that includes the keyword "red", but not the dresses filter. You then create a second search tuning rule that includes the filter "Dresses", but not the red keyword.

Comparison of Context Scenarios

Refer to the following table for more information about the various context scenarios you can enable:

Keywords	Filters	Outcome
✓	X	The search tuning rule only triggers whenever a shopper performs a sitewide search for any specified keywords.
X	✓	The search tuning rule only triggers whenever a shopper navigates to any specified category pages.
✓	✓	The search tuning rule only triggers whenever a shopper performs a search for any specified keywords in any of the specified categories.

Create a Search Tuning Rule

The search tuning rule's request and response body is a JSON object that defines the behavior of the rule.

To create a new search tuning rule:

1. Run the [AddSearchTuningRule](#) operation.

In the request body, specify the following fields:

1. Enter a `searchTuningRuleCode`, a `searchTuningRuleName`, and a `searchTuningRuleDescription`.

The search tuning rule code is referenced by Kibo; the search tuning rule name and description are for your own reference.

2. Enter a `siteId` for which you want the search tuning rule to apply to.
3. **(Optional)** Specify the `keywords` that trigger the search tuning rule.

For example, you only want the rule to apply when a shopper searches for the word "red".

If you want the search tuning rule to apply to all searches instead of a specific keyword search, leave the `keywords` field blank and set the rule as the default search tuning rule using the `isDefault` field.

4. **(Optional)** Specify the `filters`, which are category pages, that the rule is limited to. The only supported filters are categories. For example, you only want the rule to apply on a dress category page. If you want the search tuning rule to apply only to search results instead of a specific category page, leave the `filters` field blank. You must define at least one keyword or filter; they cannot both be empty.
5. Specify whether the search tuning rule is the default search tuning rule using the `isDefault` field.

The default search tuning rule applies when no other search tuning rule is present or applies. The default rule also ignores keywords and filters. Typically, most search tuning rules will not be the default.

6. Specify the `boostedProductCodes` that you want to promote to the top of the search results or the specified category pages and their sequence.

For example, you want to promote both RED-DRESS-001 and BLUE-DRESS-001 to the top, but you want RED-DRESS-001 to appear before the product BLUE-DRESS-001 so you list RED-DRESS-001 first.

7. Specify the `blockedProductCodes` that you want to block from either the search results or the specified category.

Blocked products will not appear in the search results. If you do not want to block products and only want to boost products instead, leave the blocked products field null.

All promoted or blocked products are optional. For example, you can have a search tuning rule that only consists of blocked products.

Preview Your Search Tuning Rule

Before setting your search tuning rule, you can preview its effect by completing the following steps:

1. Set your site publishing setting to Staged.

You can set your site publishing setting using either the Kibo eCommerce API or Admin. To set it via the Kibo eCommerce API, set the Data View Mode header value to `Pending`. To set it via Admin, refer to [Publishing Settings](#).

2. Set the `activeStartDate` and `activeEndDate` fields in the search tuning rule request body to a desired date.
3. Preview your staged site on the specified date. Refer to [View Your Site](#) for more information about viewing your staged site on a specific date.

Search Tuning Rules and Sorting

Search tuning rules include the ability to control the product relevance whenever shoppers sort products on a page. You can use this ability by creating a white list or a black list for the search tuning rule. The white list specifies the product properties that should always be affected by search tuning rules, and the black list specifies the product properties that should never be affected by search tuning rules.



[Delete](#)

Important Note

If you have a white list, only the product properties explicitly in the white list are affected by search tuning rules. If you have a black list, all product properties not explicitly in the black list are affected by search tuning rules except for unsupported properties. Using black lists, you can accidentally make product properties be affected by search tuning rules without your knowledge. Because of this, Kibo eCommerce strongly suggests using only white lists when controlling the product property sort fields. Black lists can potentially create unwanted behaviors when sorting products on your site.

The `inclusionExclusionType` field specifies whether the list is a white list or a black list. The valid values for this field are `WhiteList` and `BlackList`.

Default Sort By

Your site's theme determines what product property the default sort by uses.

The property your default sort by uses determines whether the default sort by can be affected by search tuning rules. If your default sort by uses a property that is unsupported by search tuning rules, such as `productCode`, then you will not be able to affect the default sort using search tuning rules.

For example, if your site's default sort by is `price`, then you will be unable to affect this default sort by using search tuning rules because `price` is unsupported by search tuning rules. If your site's default sort by is `createDate`, then you can affect this default sort by using search tuning rules.

Refer to [Unsupported Sort Field Properties](#) for a list of properties that are unsupported by search tuning rules.

Sort Fields and the API

Use the [GetSearchTuningRuleSortFields](#) and the [UpdateSearchTuningRuleSortFields](#) operations of the `commerce/catalog/admin/search` resource to interact with the search tuning rule sort fields.

Sort Fields Example

For example, you want search tuning rules that affect product popularity, margin, and rating to always be in effect whenever a shopper sorts a page by either of these properties. To accomplish this, you write the following white list JSON that specifies any search tuning rules that affect popularity to be in effect whenever a page is sorted by popularity:

```
{
  "sortFields": [
    "popularity",
    "margin",
    "rating"
  ],
  "inclusionExclusionType": "WhiteList"
}
```

Unsupported Sort Field Properties

The sort fields rule can only control custom product properties, such as popularity and rating. The sort fields rule does not support the following product properties, and these properties will always have a logical sort behavior:

- `price`
- `price.catalogListPrice`
- `price.catalogSalePrice`
- `price.salePrice`
- `Measurements.PackageWeight.Value`
- `Measurements.PackageHeight.Value`
- `Measurements.PackageLength.Value`
- `Measurements.PackageWidth.Value`
- `categoryCode`
- `productTypeId`
- `DaysAvailableInCatalog`
- `productName`
- `productCode`
- `categoryNames`

Search Tuning Rule Examples

Example 1

The following example shows a search tuning rule that boosts a pair of gold shoes whenever a shopper searches for the keyword "shoes" on any page:

```
{
  "searchTuningRuleCode": "gold-shoes",
  "searchTuningRuleName": "Gold Shoes Rule",
  "searchTuningRuleDescription": "Promote the pair of custom gold shoes in any search for shoes",
  "keywords": [
    "shoes"
  ],
  "filters": [
  ],
  "active": "true",
  "isDefault": "false",
  "boostedProductCodes": [
    "GOLD-SHOES-001"
  ],
  "blockedProductCodes": [
  ]
}
```

Example 2

The following example shows a search tuning rule that boosts several red dresses, and blocks a blue dress and a black dress whenever a shopper searches for the keyword "red" on the dresses category page:

```
{
  "searchTuningRuleCode": "red-dress",
  "searchTuningRuleName": "Red Dresses Rule",
  "searchTuningRuleDescription": "Promote the three red dresses, and block the blue, and black dresses in any search for red on the dresses category",
  "keywords": [
    "red"
  ],
  "filters": [
    {
      "field": "categoryCode",
      "value": "9"
    }
  ],
  "active": "true",
  "isDefault": "false",
  "boostedProductCodes": [
    "RED-DRESS-001",
    "RED-DRESS-002",
    "RED-DRESS-003"
  ],
  "blockedProductCodes": [
    "BLUE-DRESS-001",
    "BLCK-DRESS-002"
  ]
}
```