

Quickly update the value of a field in a document

```
POST <index>/<type>/<id>/_update
{
  "doc": {
    "<field>": {
      "<optional_inner_field>": "<new value>"
    }
  }
}
```

Quickly increment the value of a field in a document

```
POST <index>/<type>/<id>/_update
{
  "script" : {
    "source": "ctx._source.<field>.<optional_inner_field> += params.count",
    "lang": "painless",
    "params" : {
      "count" : 1
    }
  }
}
```

Get active index templates

```
GET _template/<template_name_pattern>
```

Update or create an index template

```
PUT /_template/<template_name>
{
  "template": "<index_pattern>",
  "settings": {
    "index.refresh_interval": "5s",
    "index.number_of_replicas": "0"
  },
  "index.number_of_shards": "1",
  "analysis": {}
},
"mappings": {
  "<type>": {
    "properties": {
      "<date_field>": {
        "type": "date"
      }
    }
  }
},
"aliases": { "<index_alias>": {} }
}
```

Quickly reindex (after template or mapping changes)

```
POST _reindex
{
  "source": {
    "index": "<index1>"
  },
  "dest": {
    "index": "<index1a>"
  }
}
```

Activate the search slow log on an index

```
PUT <index>/_settings
{
  "index": {
    "search.slowlog.threshold.query.warn": "0s",
    "search.slowlog.threshold.fetch.warn": "0s",
  }
}
$ tail -f <path.logs>/<index>_index_search_slowlog.log
```

Get nodes information

```
GET _cat/nodes?v
```

Get the cluster health

```
GET _cluster/health
```

Get all the cluster metadata

```
GET _cluster/state
PUT /_snapshot/<backup>/<snapshot>
?wait_for_completion=false
{
  "indices": "index1,index2",
  "ignore_unavailable": false,
  "include_global_state": false
}
GET /_snapshot/<backup>/<snapshot>
```

Quickly restore

```
POST /_snapshot/<backup>/<snapshot>/_restore
{
  "indices": "index1",
  "index_settings": {
    "index.number_of_replicas": 0
  }
}
```

Run the support diagnostics

Go to <https://github.com/elastic/elasticsearch-support-diagnostics/releases/latest>

```
$ wget https://github.com/elastic/support-diagnostics/releases/download/X.Y/support-diagnostics-X.Y-dist.zip
```

```
$ unzip support-diagnostics-X.Y-dist.zip
```

```
$ cd support-diagnostics-X.Y/
```

```
$ ./diagnostics.sh --host localhost --port 9200 -type remote
```

See the output in remote-diagnostics-<date>.tar.gz

Quickly snapshot

```
GET /_snapshot/_all
PUT /_snapshot/<backup>
```

```
{
  "type": "fs",
  "settings": {
    "compress": true,
    "location": "<path.repo>"
  }
}
```



CHEAT SHEET

Get the Elasticsearch version and tagline

```
GET /
```

Browse the documentation related to your elasticsearch version

Go to

<https://www.elastic.co/guide/en/elasticsearch/reference/X.Y/index.html>

Quickly explore existing indices

```
GET _cat/indices/<index_pattern>?v
```

Quickly explore existing aliases

```
GET /_cat/aliases/<alias_pattern>?v
```

Quickly get settings, mappings and aliases of an index

```
GET <index>/
```



<https://github.com/adelean/elasticsearch-cheatsheet>



Quickly get data matching several fields and explain the relevancy

```
GET /<index_or_alias>/_search
{
  "explain": true,
  "query": {
    "multi_match": {
      "query": "<query>",
      "fields": [
        "<field1>^10",
        "<field2>"
      ]
    }
  }
}
```

Switch / Manage aliases

```
POST /_aliases
{
  "actions": [
    {
      "remove": {
        "index": "<old_index>",
        "alias": "<index_alias>"
      }
    },
    {
      "add": {
        "index": "<new_index>",
        "alias": "<index_alias>"
      }
    }
  ]
}
```

Quickly get some data based on exact values and sorted

```
GET /<index_or_alias>/_search
{
  "from": 0,
  "size": 20,
  "query": {
    "bool": {
      "filter": {
        "terms": {
          "<field>.keyword": [
            "value_1",
            "value_2"
          ]
        }
      }
    }
  },
  "sort": [
    {
      "<sort_field>": {
        "order": "desc"
      }
    }
  ]
}
```

Quickly explore the data in an index

```
GET /<index>/_search
{
  "from": 0,
  "size": 20,
  "_source": {
    "include": [
      "<field_patterns_to_include>"
    ],
    "exclude": [
      "<field_patterns_to_exclude>"
    ]
  },
  "query": {
    "match_all": {}
  },
  "aggs": {
    "type": {
      "terms": {
        "field": "_type",
        "size": 10
      }
    }
  }
}
```

Quickly check how a text is analyzed

```
GET <index>/_analyze
{
  "field" : "<field>",
  "text" : "<text to analyze>"
}

GET <index>/_analyze
{
  "analyzer" : "standard",
  "text" : "<text to analyze>"
}
```

Quickly explore top terms

```
GET /<index_or_alias>/_search
{
  "size": 0,
  "aggs": {
    "<aggl>": {
      "terms": {
        "field": "<field1>.keyword",
        "size": 10
      }
    }
  }
}
```

Explore significant terms within one group

```
GET /<index_or_alias>/_search
{
  "size": 0,
  "aggs": {
    "<aggl>": {
      "terms": {
        "field": "<field1>.keyword",
        "size": 10
      },
      "aggs": {
        "<aggl.1>": {
          "significant_terms": {
            "field": "<field2>.keyword"
          }
        }
      }
    }
  }
}
```

Explore data grouped by a field

```
GET /<index_or_alias>/_search
{
  "size": 10,
  "_source": {
    "excludes": "*"
  },
  "query": {
    "match": {
      "<field>": {
        "query": "<query>"
      }
    }
  },
  "collapse": {
    "field": "<collapse_field>.keyword",
    "inner_hits": {
      "size": 5,
      "name": "inner_hits",
      "highlight": {
        "fields": {
          "<field>": {}
        }
      }
    }
  },
  "sort": [
    {
      "<inner_hits_sort_field>": {
        "order": "asc"
      }
    }
  ]
}
```

Explore top terms and get details on top hits

```
GET /<index_or_alias>/_search
{
  "size": 0,
  "aggs": {
    "<aggl>": {
      "terms": {
        "field": "<field>.keyword",
        "size": 10
      },
      "aggs": {
        "<details>": {
          "top_hits": {
            "size": 1,
            "_source": [
              "<details_field>.*"
            ],
            "highlight": {
              "pre_tags": [
                "<em>"
              ],
              "post_tags": [
                "</em>"
              ],
              "fields": {
                "<highlight_fields>.*": {
                  "number_of_fragments": 0
                }
              }
            }
          }
        }
      }
    }
  }
}
```

Quickly find min and max values of one field

```
GET /<index_or_alias>/_search
{
  "size": 0,
  "aggs": {
    "min": {
      "min": {
        "field": "<field>"
      }
    },
    "max": {
      "max": {
        "field": "<field>"
      }
    }
  }
}
```

Quickly display a date histogram

```
GET /<index_or_alias>/_search
{
  "size": 0,
  "aggs": {
    "histogram": {
      "date_histogram": {
        "field": "<date_field>",
        "interval": "day"
      }
    }
  }
}
```