

Iniciando o Elasticsearch em Docker

Docker Desktop Windows

```
PS E:\projetos\docker-elasticsearch\elastic> wsl -l -v
NAME                STATE              VERSION
* docker-desktop-data Running            2
  docker-desktop    Running            2
  Ubuntu-20.04       Running            2
```

```
PS E:\projetos\docker-elasticsearch\elastic> wsl -d docker-desktop
LAPTOP-V176DRSL:/tmp/docker-desktop-root/mnt/host/e/projetos/docker-elasticsearch/elastic# sysctl -w vm.max_map_count=262144
vm.max_map_count = 262144
LAPTOP-V176DRSL:/tmp/docker-desktop-root/mnt/host/e/projetos/docker-elasticsearch/elastic#
```

Docker Wsl2 Linux

```
feliciani@LAPTOP-V176DRSL:~$ sudo sysctl -w vm.max_map_count=262144
[sudo] password for feliciani:
vm.max_map_count = 262144
```

Docker Desktop Windows


```
PS E:\projetos\docker-elasticsearch\elastic> docker-compose up -d
Docker Compose is now in the Docker CLI, try 'docker compose up'




Starting elastic_elasticsearch_1 ... done
Starting elastic_kibana_1         ... done
Starting elastic_logstash_1       ... done
```

```
PS E:\projetos\docker-elasticsearch\elastic> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
		NAMES			
d3d012693acc	docker.elastic.co/logstash/logstash:7.9.2	"/usr/local/bin/dock...	17 hours ago	Up 33 minutes	0.0.0.0:5044->5044/tcp,
:::5044->5044/tcp,	0.0.0.0:9600->9600/tcp, :::9600->9600/tcp	elastic_logstash_1			
ca700688aa0d	docker.elastic.co/kibana/kibana:7.9.2	"/usr/local/bin/dumb...	17 hours ago	Up 33 minutes	0.0.0.0:5601->5601/tcp,
:::5601->5601/tcp		elastic_kibana_1			
37a2fb5958f4	docker.elastic.co/elasticsearch/elasticsearch:7.9.2	"/tini -- /usr/local...	17 hours ago	Up 34 minutes	0.0.0.0:9200->9200/tcp,
:::9200->9200/tcp,	9300/tcp	elastic_elasticsearch_1			

Docker



 docker




Upgrade    Sign in

Containers / Apps


Images


Dev Environments


  elastic
E:\projetos\docker-elasticsearch\elastic

Open in Visual Studio Code   



CONTAINERS

 elastic_logstash_1
docker.elastic.co/logstash/logstash:7.9.2
RUNNING PORT: 5044

 elastic_kibana_1
docker.elastic.co/kibana/kibana:7.9.2
RUNNING PORT: 5601

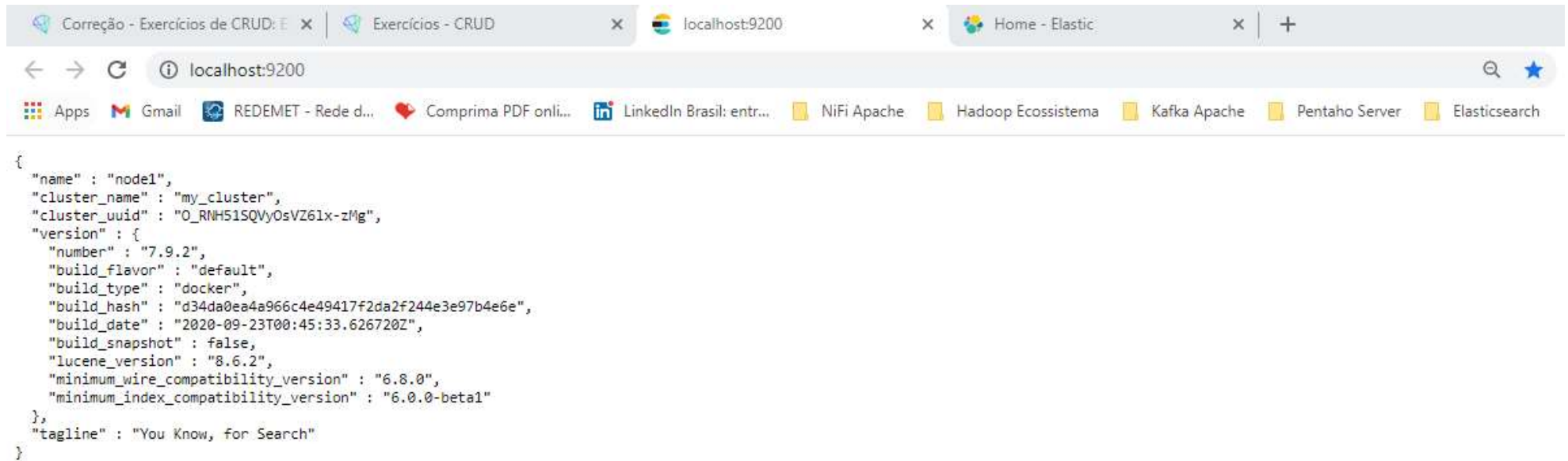
 elastic_elasticsearch_1
docker.elastic.co/elasticsearch/elasticsearch:7.9.2
RUNNING PORT: 9200

```
dest":"empty","referer":"http://localhost:5601/app/home","accept-encoding":"gzip, deflate, br","accept-language":"pt-BR,pt;q=0.9,en-US;q=0.8,en;q=0.7"},"remoteAddress":"172.18.0.1","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.77 Safari/537.36","referer":"http://localhost:5601/app/home"},"res":{"statusCode":200,"responseTime":751,"contentLength":9,"message":"POST /api/ui_metric/report 200 751ms - 9.0B"}
elasticsearch_1 | {"type": "server", "timestamp": "2021-06-10T13:42:53,810Z", "level": "INFO", "component": "o.e.m.j.JvmGcMonitorService", "cluster.name": "my_cluster", "node.name": "node1", "message": "[gc][1024] overhead, spent [304ms] collecting in the last [1s]", "cluster.uuid": "O_RNH51SQVyOsVZ6lx-zMg", "node.id": "I0MovYK2T0eNK1jQUwWviA" }
elasticsearch_1 | {"type": "server", "timestamp": "2021-06-10T13:50:02,818Z", "level": "WARN", "component": "o.e.m.f.FsHealthService", "cluster.name": "my_cluster", "node.name": "node1", "message": "health check of [/usr/share/elasticsearch/data/nodes/0] took [5569ms] which is above the warn threshold of [5s]", "cluster.uuid": "O_RNH51SQVyOsVZ6lx-zMg", "node.id": "I0MovYK2T0eNK1jQUwWviA" }
elasticsearch_1 | {"type": "server", "timestamp": "2021-06-10T13:50:12,644Z", "level": "WARN", "component": "o.e.m.j.JvmGcMonitorService", "cluster.name": "my_cluster", "node.name": "node1", "message": "[gc][young][1454][20] duration [2.5s], collections [1]/[3.1s], total [2.5s]/[4.5s], memory [367.7mb]->[86.2mb]/[512mb], all_pools {[young][282mb]->[0b]/[0b]}{[old][76.2mb]->[76.2mb]/[512mb]}{[survivor][9.4mb]->[10mb]/[0b]}", "cluster.uuid": "O_RNH51SQVyOsVZ6lx-zMg", "node.id": "I0MovYK2T0eNK1jQUwWviA" }
elasticsearch_1 | {"type": "server", "timestamp": "2021-06-10T13:50:12,645Z", "level": "WARN", "component": "o.e.m.j.JvmGcMonitorService", "cluster.name": "my_cluster", "node.name": "node1", "message": "[gc][1454] overhead, spent [2.5s] collecting in the last [3.1s]", "cluster.uuid": "O_RNH51SQVyOsVZ6lx-zMg", "node.id": "I0MovYK2T0eNK1jQUwWviA" }
```

Search...  Stick to bottom 

Acessado o Elasticsearch

<http://localhost:9200>



```
{
  "name" : "node1",
  "cluster_name" : "my_cluster",
  "cluster_uuid" : "O_RNH51SQVyOsVZ61x-zMg",
  "version" : {
    "number" : "7.9.2",
    "build_flavor" : "default",
    "build_type" : "docker",
    "build_hash" : "d34da0ea4a966c4e49417f2da2f244e3e97b4e6e",
    "build_date" : "2020-09-23T00:45:33.626720Z",
    "build_snapshot" : false,
    "lucene_version" : "8.6.2",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

Acessando o KIBANA

<http://localhost:5601>

The screenshot shows the Kibana web interface in a browser window. The browser's address bar displays `localhost:5601/app/home/`. The page features a left-hand navigation sidebar with a hamburger menu icon, a Kibana logo, and a 'Home' button. The sidebar lists several categories with expandable menus: 'Recently viewed' (currently empty), 'Kibana' (containing Discover, Dashboard, Canvas, Maps, Machine Learning, and Visualize), 'Enterprise Search' (containing App Search and Workplace Search), and 'Observability' (containing Overview and Logs). The main content area is titled 'Home' and contains several interactive cards. These include 'Logs' (with an 'Add log data' button), 'Metrics' (with an 'Add metric data' button), and 'Security' (with an 'Add events' button). Below these are three larger cards: 'Add sample data' (with a subtext 'Import a CSV, NDJSON, or log file'), 'Upload data from log file' (with a subtext 'Import a CSV, NDJSON, or log file'), and 'Use Elasticsearch data' (with a subtext 'Connect to your Elasticsearch index'). At the bottom, there are two more sections: 'Explore Data' (with a subtext 'Leverage dashboards, analytics, and APIs for advanced application') and 'Manage and Administer the Elastic Stack' (containing 'Console' and 'Rollups' options).

Correção - Exercícios de CRUD: E x | Exercícios - CRUD x | localhost:9200 x | Home - Elastic x +

localhost:5601/app/home/

Apps Gmail REDEMET - Rede d... Comprima PDF onli... LinkedIn Brasil: entr... NiFi Apache Hadoop Ecosistema Kafka Apache Pentaho Server Elasticsearch

Home

Home

Recently viewed

No recently viewed items

Kibana

Discover

Dashboard

Canvas

Maps

Machine Learning

Visualize

Enterprise Search

App Search

Workplace Search

Observability

Overview

Logs

Availability

Logs

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Add log data

Metrics

Collect metrics from the operating system and services running on your servers.

Add metric data

Security

SIEM + Endpoint Security

Protect hosts, analyze security information and events, hunt threats, automate detections, and create cases.

Add events

Add sample data

Import a CSV, NDJSON, or log file

Upload data from log file

Import a CSV, NDJSON, or log file

Use Elasticsearch data

Connect to your Elasticsearch index

Explore Data

App Search

Leverage dashboards, analytics, and APIs for advanced application

Manage and Administer the Elastic Stack

Console

Skip cURL and use this JSON interface to work with your data directly

Rollups

Summarize and store historical data in a smaller interval for future analysis

Exercitando Querys e Filtros

Acessar o MENU -> Dev Tools

The screenshot displays the Elastic Home interface in a web browser. The browser's address bar shows the URL `localhost:5601/app/home#/`. The page layout includes a sidebar on the left with a menu structure:

- Home
- Recently viewed (No recently viewed items)
- Security
 - Overview
 - Detections
 - Hosts
 - Network
 - Timelines
 - Cases
 - Administration
- Management
 - Dev Tools (highlighted)
 - Ingest Manager
 - Stack Monitoring
 - Stack Management

The main content area is divided into several sections:

- Logs**: Ingest logs from popular data sources and easily visualize in preconfigured dashboards. Includes an [Add log data](#) button.
- Metrics**: Collect metrics from the operating system and services running on your servers. Includes an [Add metric data](#) button.
- Security**: SIEM + Endpoint Security. Protect hosts, analyze security information and events, hunt threats, automate detections, and create cases. Includes an [Add events](#) button.
- Add sample data**: Set up a sample data source and a Kibana dashboard.
- Upload data from log file**: Import a CSV, NDJSON, or log file.
- Use Elasticsearch data**: Connect to your Elasticsearch index.
- Explore Data**: Leverage dashboards, analytics, and APIs for advanced application search made simple. Includes an [App Search](#) button.
- Manage and Administer the Elastic Stack**: Includes [Console](#) (Skip cURL and use this JSON interface to work with your data directly.) and [Rollups](#) (Summarize and store historical data in a smaller index for future analysis.)

The Windows taskbar at the bottom shows the system time as 17:29 and a battery level of 97%.

Realizar todas as buscas a seguir no índice produto

Verificando a existência do INDEX produto

HEAD produto

The screenshot shows a web browser window with the DevTools console open. The browser's address bar shows the URL `localhost:5601/app/dev_tools#/console`. The DevTools interface includes tabs for Console, Search Profiler, Grok Debugger, and Painless Lab (marked as BETA). The Console tab is active, displaying a list of REST client requests and their responses.

The requests listed in the console are:

- 1 HEAD produto
- 2
- 3
- 4 #-----#
- 5
- 6
- 7 POST produto/_open
- 8
- 9 GET produto/_search
- 10
- 11 POST produto/_close
- 12
- 13 GET produto2/_mapping
- 14
- 15 GET produto/_count
- 16
- 17 GET produto2/_count
- 18
- 19 GET produto2/_search
- 20
- 21 POST _reindex
- 22 {
- 23 "source": {
- 24 "index": "produto"
- 25 },
- 26 "dest": {
- 27 "index": "produto2"
- 28 }
- 29 }
- 30
- 31 GET produto2/_mapping
- 32
- 33 PUT produto2/_mapping
- 34

The response for the first request (HEAD produto) is shown on the right side of the console:

```
1 200 - OK
```

The status bar at the bottom of the console indicates a 200 - OK response with a response time of 186 ms.

1. Buscar no termo nome o valor mouse

GET produto/_search {"query": {"term": {"nome": "mouse"}}}

The screenshot displays the Elastic Dev Tools interface in a web browser. The browser's address bar shows the URL `localhost:5601/app/dev_tools#/console`. The Dev Tools panel is open, showing the 'Console' tab. The left pane contains a list of HTTP requests, with the first one selected: `GET produto/_search`. The request body is visible as `{"query": {"term": {"nome": "mouse"}}}`. The right pane shows the response, which is a JSON object indicating a successful search. The response includes metadata like `"took": 1`, `"timed_out": false`, and `"successful": 1`. The `"hits"` array contains one document with `"_index": "produto"`, `"_type": "_doc"`, `"_id": "1"`, `"_score": 1.5697745`, and `"source": {"nome": "mouse", "qtd": 50, "descricao": "com fio USB, compativel com Windows, Mac e Linux"}`. The status bar at the bottom right of the console indicates `200 - OK` and `57 ms`. The Windows taskbar is visible at the bottom of the screen.

```
1 GET produto/_search
2 {
3   "query": {
4     "term": {
5       "nome": "mouse"
6     }
7   }
8 }
9
10 HEAD produto
11
12 #-----#
13
14
15 POST produto/_open
16
17 GET produto/_search
18
19 POST produto/_close
20
21 GET produto2/_mapping
22
23 GET produto/_count
24
25 GET produto2/_count
26
27 GET produto2/_search
28
29 POST _reindex
30 {
31   "source": {
32     "index": "produto"
33   }
34 }
```

```
1 {
2   "took" : 1,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 1,
13      "relation" : "eq"
14    },
15    "max_score" : 1.5697745,
16    "hits" : [
17      {
18        "_index" : "produto",
19        "_type" : "_doc",
20        "_id" : "1",
21        "_score" : 1.5697745,
22        "_source" : {
23          "nome" : "mouse",
24          "qtd" : 50,
25          "descricao" : "com fio USB, compativel com Windows, Mac e Linux"
26        }
27      }
28    ]
29  }
30 }
```

200 - OK 57 ms

2. Buscar no termo nome os valores mouse e teclado

GET produto/_search {"query": {"terms": {"nome": ["mouse", "teclado"]}}}

The screenshot shows a web browser window with the Elastic Dev Tools console open. The console displays a REST client request and its response.

Request:

```
1 GET produto/_search
2 {
3   "query": {
4     "terms": {
5       "nome": ["mouse", "teclado"]
6     }
7   }
8 }
9
10
11 GET produto/_search
12 {
13   "query": {
14     "term": {
15       "nome": "mouse"
16     }
17   }
18 }
19
20 HEAD produto
21
22 #-----#
23
24
25
26 POST produto/_open
27
28 GET produto/_search
29
30 POST produto/_close
31
32 GET produto2/_mapping
33
34 GET produto/_search
```

Response:

```
10 hits: {
11   "total": {
12     "value": 2,
13     "relation": "eq"
14   },
15   "max_score": 1.0,
16   "hits": [
17     {
18       "_index": "produto",
19       "_type": "_doc",
20       "_id": "1",
21       "_score": 1.0,
22       "_source": {
23         "nome": "mouse",
24         "qtd": 50,
25         "descricao": "com fio USB, compatível com Windows, Mac e Linux"
26       }
27     },
28     {
29       "_index": "produto",
30       "_type": "_doc",
31       "_id": "6",
32       "_score": 1.0,
33       "_source": {
34         "nome": "teclado",
35         "qtd": 100,
36         "descricao": "USB",
37         "data": "2020-09-18"
38       }
39     }
40   ]
41 }
42
43
```

The status bar at the top right of the console shows "200 - OK" and "122 ms".

3. Realizar a mesma busca do item 1 e 2, desconsiderando o score

GET produto/_search {"query": {"constant_score": {"filter": {"term": {"nome": "mouse"}}}}}

The screenshot shows the Elastic Dev Tools console interface. The left pane displays the search query, and the right pane shows the JSON response. The status bar at the top right indicates a 200 OK response with a 141 ms execution time.

Query (Left Pane):

```
1 GET produto/_search
2 {
3   "query": {
4     "constant_score": {
5       "filter": {
6         "term": {
7           "nome": "mouse"
8         }
9       }
10    }
11  }
12 }
13
14 GET produto/_search
15 {
16   "query": {
17     "terms": {
18       "nome": ["mouse", "teclado"]
19     }
20   }
21 }
22
23
24 GET produto/_search
25 {
26   "query": {
27     "term": {
28       "nome": "mouse"
29     }
30   }
31 }
32
33 HEAD produto
34
```

Response (Right Pane):

```
1 {
2   "took" : 1,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 1,
13      "relation" : "eq"
14    },
15    "max_score" : 1.0,
16    "hits" : [
17      {
18        "_index" : "produto",
19        "_type" : "_doc",
20        "id" : "1",
21        "_score" : 1.0,
22        "source" : {
23          "nome" : "mouse",
24          "qtd" : 50,
25          "descricao" : "com fio USB, compativel com Windows, Mac e Linux"
26        }
27      }
28    ]
29  }
30 }
31
```

GET produto/_search {"query": {"constant_score": {"filter": {"terms": {"nome": ["mouse", "teclado"]}}}}}

The screenshot shows the Elastic Dev Tools console interface. The left pane displays the search query being executed, and the right pane shows the resulting JSON response.

Query (Left Pane):

```
1 GET produto/_search
2 {
3   "query": {
4     "constant_score": {
5       "filter": {
6         "terms": {
7           "nome": [
8             "mouse",
9             "teclado"
10          ]
11        }
12      }
13    }
14  }
15 }
16
17 GET produto/_search
18 {
19   "query": {
20     "constant_score": {
21       "filter": {
22         "term": {
23           "nome": "mouse"
24         }
25       }
26     }
27   }
28 }
29
30 GET produto/_search
31 {
32   "query": {
33     "terms": {
34       "nome": ["mouse", "teclado"]
35     }
36   }
37 }
```

Response (Right Pane):

```
8   "failed" : 0
9 },
10 "hits" : {
11   "total" : {
12     "value" : 2,
13     "relation" : "eq"
14   },
15   "max_score" : 1.0,
16   "hits" : [
17     {
18       "_index" : "produto",
19       "_type" : "_doc",
20       "id" : "1",
21       "_score" : 1.0,
22       "_source" : {
23         "nome" : "mouse",
24         "qtd" : 50,
25         "descricao" : "com fio USB, compativel com Windows, Mac e Linux"
26       }
27     },
28     {
29       "_index" : "produto",
30       "_type" : "_doc",
31       "id" : "6",
32       "_score" : 1.0,
33       "_source" : {
34         "nome" : "teclado",
35         "qtd" : 100,
36         "descricao" : "USB",
37         "data" : "2020-09-18"
38       }
39     }
40   ]
41 }
```

The status bar at the top right of the console indicates a 200 status code and a response time of 69 ms.

4. Buscar os documentos que contenham a palavra “USB” no atributo descrição

GET produto/_search {"query": {"match": {"descricao": "usb"}}}

The screenshot shows the Elastic Dev Tools console interface. The left pane displays the search query: `GET produto/_search` with a `match` query on the `descricao` field for the value `usb`. The right pane shows the JSON response from the search. The response includes a `total` of 3 hits, a `successful` status of 1, and a `failed` status of 0. The `hits` array contains three documents. The first document is a keyboard with `nome: "teclado"`, `qtd: 100`, and `descricao: "USB"`. The second document is a mouse with `nome: "mouse"`, `qtd: 50`, and `descricao: "com fio USB, compatível com Windows, Mac e Linux"`. The third document is also a mouse with `nome: "mouse"`, `qtd: 50`, and `descricao: "com fio USB, compatível com Windows, Mac e Linux"`. The console also shows the execution history of the queries.

```
1 GET produto/_search
2 {
3   "query": {
4     "match": {
5       "descricao": "usb"
6     }
7   }
8 }
9
10 GET produto/_search
11 {
12   "query": {
13     "constant_score": {
14       "filter": {
15         "terms": {
16           "nome": [
17             "mouse",
18             "teclado"
19           ]
20         }
21       }
22     }
23   }
24 }
25
26 GET produto/_search
27 {
28   "query": {
29     "constant_score": {
30       "filter": {
31         "term": {
32           "nome": "mouse"
33         }
34       }
35     }
36   }
37 }
```

```
{
  "total": 1,
  "successful": 1,
  "skipped": 0,
  "failed": 0
},
{
  "hits": {
    "total": {
      "value": 3,
      "relation": "eq"
    },
    "max_score": 0.801211,
    "hits": [
      {
        "_index": "produto",
        "_type": "_doc",
        "_id": "6",
        "_score": 0.801211,
        "_source": {
          "nome": "teclado",
          "qtd": 100,
          "descricao": "USB",
          "data": "2020-09-18"
        }
      },
      {
        "_index": "produto",
        "_type": "_doc",
        "_id": "1",
        "_score": 0.40609324,
        "_source": {
          "nome": "mouse",
          "qtd": 50,
          "descricao": "com fio USB, compatível com Windows, Mac e Linux"
        }
      },
      {
        "_index": "produto",
        "_type": "_doc",
        "_id": "1",
        "_score": 0.40609324,
        "_source": {
          "nome": "mouse",
          "qtd": 50,
          "descricao": "com fio USB, compatível com Windows, Mac e Linux"
        }
      }
    ]
  }
}
```

5. Buscar os documentos que contenham a palavra “USB” e não contenham a palavra “Linux” no atributo descrição

GET produto/_search {"query": {"bool": {"must": [{"match": {"descricao": "USB"}]}, "must_not": [{"match": {"descricao": "linux"}}]}}

The screenshot displays the Elastic Dev Tools console in a web browser. The left pane shows the following query:

```
1 GET produto/_search
2 {
3   "query": {
4     "bool": {
5       "must": [
6         {
7           "match": {
8             "descricao": "USB"
9           }
10        },
11      ],
12      "must_not": [
13        {
14          "match": {
15            "descricao": "linux"
16          }
17        }
18      ]
19    }
20  }
21 }
22
23 GET produto/_search
24 {
25   "query": {
26     "match": {
27       "descricao": "usb"
28     }
29   }
30 }
31
32 GET produto/_search
33 {
```

The right pane shows the response for the first query:

```
1 {
2   "took" : 129,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 2,
13      "relation" : "eq"
14    },
15    "max_score" : 0.801211,
16    "hits" : [
17      {
18        "_index" : "produto",
19        "_type" : "_doc",
20        "_id" : "6",
21        "_score" : 0.801211,
22        "_source" : {
23          "nome" : "teclado",
24          "qtd" : 100,
25          "descricao" : "USB",
26          "data" : "2020-09-18"
27        }
28      },
29      {
30        "_index" : "produto",
31        "_type" : "_doc",
32        "_id" : "2",
33        "_score" : 0.36152202,
```

The status bar at the top right of the console indicates "200 - OK" and "191 ms".

6. Buscar os documentos que podem ter a palavra “memória” no atributo nome ou contenham a palavra “USB” e não contenham a palavra “Linux” no atributo descrição

Should funciona como “ou”

GET produto/_search

```
{ "query": { "bool": { "should": [ { "match": { "nome": "memória" } }, { "match": { "descricao": "usb" } } ], "must_not": [ { "match": { "descricao": "linux" } } ] } } }
```

The screenshot shows the Elastic Dev Tools console interface. The top bar includes tabs for 'Correção - Exercícios de Query e', 'Exercícios - Query e Filtros', 'localhost:9200', and 'Dev Tools - Elastic'. The address bar shows 'localhost:5601/app/dev_tools#/console'. The left sidebar has a 'Dev Tools' tab. The main area is divided into 'Console', 'Search Profiler', 'Grok Debugger', and 'Painless Lab' (marked BETA). The 'Console' tab is active, showing a search query and its results.

Query:

```
1 GET produto/_search
2 {
3   "query": {
4     "bool": {
5       "should": [
6         {
7           "match": {
8             "nome": "memória"
9           }
10        },
11        {
12          "match": {
13            "descricao": "usb"
14          }
15        }
16      ],
17      "must_not": [
18        {
19          "match": {
20            "descricao": "linux"
21          }
22        }
23      ]
24    }
25  }
26 }
```

Results:

```
19 {
20   "_type": "_doc",
21   "_id": "6",
22   "_score": 0.801211,
23   "_source": {
24     "nome": "teclado",
25     "qtd": 100,
26     "descricao": "USB",
27     "data": "2020-09-18"
28   }
29 },
30 {
31   "_index": "produto",
32   "_type": "_doc",
33   "_id": "3",
34   "_score": 0.744874,
35   "_source": {
36     "nome": "memória ram",
37     "qtd": 30,
38     "descricao": "8GB, DDR4"
39   }
40 },
41 {
42   "_index": "produto",
43   "_type": "_doc",
44   "_id": "2",
45   "_score": 0.36152202,
46   "_source": {
47     "nome": "hd",
48     "qtd": 20,
49     "descricao": "Interface USB 2.0, 500GB, Sistema: Windows 10, Windows 8, Windows 7"
50   }
51 }
```

The bottom status bar shows '100%' zoom, system icons, and the time '11:06'.