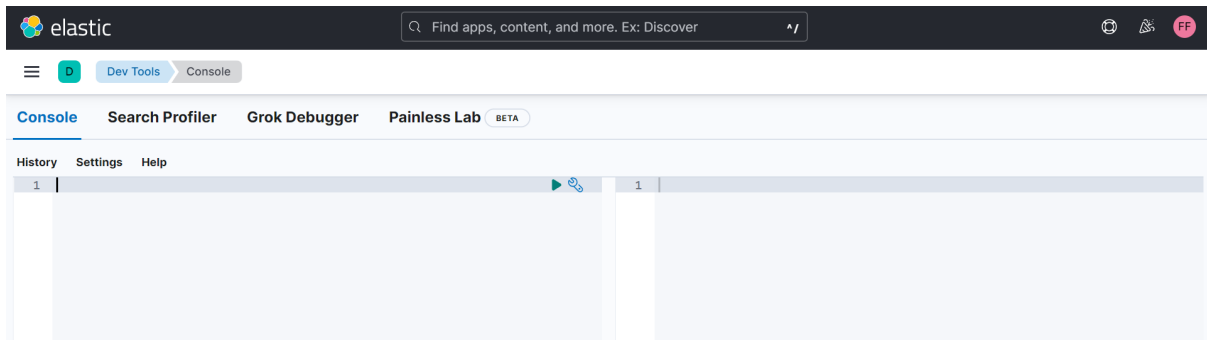


# Práctica Elasticsearch

Fernando Fortanel Rojas

## Creación de un índice

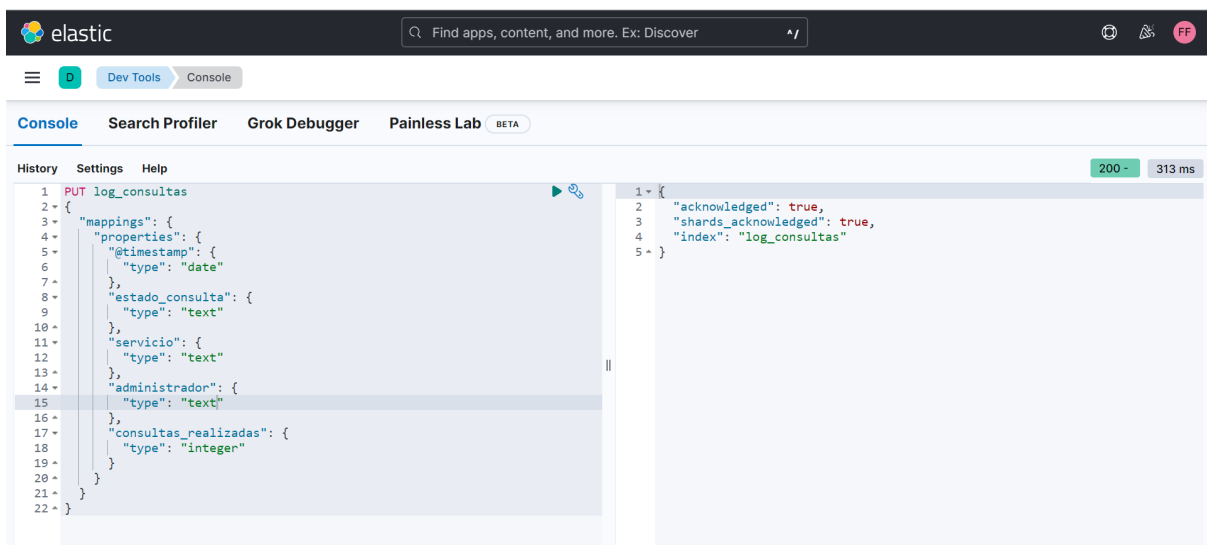
1.- Selecciona tu consola de Dev Tools para realizar la siguiente actividad



2.- Crea un índice con el nombre de **log\_consultas** a partir del siguiente **JSON**:

```
{
  "@timestamp": "2010-05-15T22:00:54",
  "estado_consulta": "consumo",
  "servicio": "consulta",
  "administrador": "Juan Carlos",
  "consultas_realizadas": 52
}
```

Para crear el índice se utilizó el siguiente código. Primero se define el nombre del índice, A continuación, se define el mapping, el cual contiene la información sobre los campos y los tipos de datos. Se definió estos campos y los tipos de datos de acuerdo al json anterior.



Después de ejecutar el código obtenemos un json informando que se tuvo éxito al crear el índice.

3.- Obtén el mapping del índice anterior y genera un template a partir de este índice haciendo uso del API de plantillas (**TEMPLATE**). El patrón para el índice debe ser: **"log\_consultas"**. Verifica los tipos de datos hagan sentido con la información almacenada.

Para obtener el mapping del índice que se creó, se utilizó el siguiente código en la consola de Dev Tools. La consola regresa el siguiente json, el cual contiene información sobre el mapping del índice.

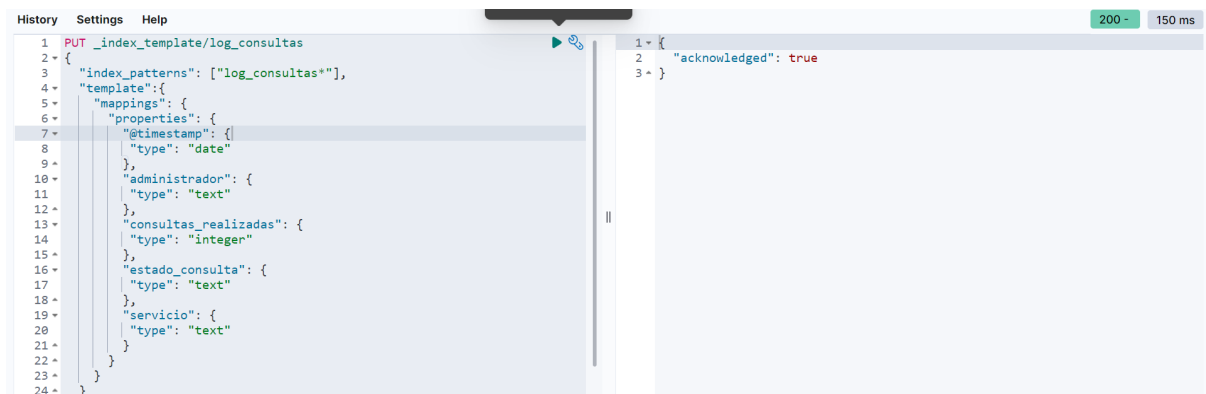


```
1 GET log_consultas/_mapping
200 - 262 ms

1 {
2   "log_consultas": {
3     "mappings": {
4       "properties": {
5         "@timestamp": {
6           "type": "date"
7         },
8         "administrador": {
9           "type": "text"
10        },
11       "consultas_realizadas": {
12         "type": "integer"
13       },
14       "estado_consulta": {
15         "type": "text"
16       },
17       "servicio": {
18         "type": "text"
19       }
20     }
21   }
22 }
23 }
```

Con el siguiente código, se creó el template. Al inicio se especifica la creación de un index template llamado log\_consultas. En el cuerpo del código es necesario especificar el index\_patterns y el template.

El **index\_patterns** nos permite especificar los nombres de los índices que al ser creados utilizaran la configuración establecida en el template. Para identificar estos índices se utiliza una expresión regular.



```
1 PUT _index_template/log_consultas
2 {
3   "index_patterns": ["log_consultas*"],
4   "template": {
5     "mappings": {
6       "properties": {
7         "@timestamp": {
8           "type": "date"
9         },
10        "administrador": {
11          "type": "text"
12        },
13        "consultas_realizadas": {
14          "type": "integer"
15        },
16        "estado_consulta": {
17          "type": "text"
18        },
19        "servicio": {
20          "type": "text"
21        }
22      }
23    }
24  }
25 }

1 {
2   "acknowledged": true
3 }
```

En **template** se define la configuración de este template. En este caso solo se definió el mapping del índice que creamos anteriormente.

4. Una vez definido tu template cargaras una serie de documentos en tu índice utilizando el archivo que se encuentra en el escritorio: **log\_consultas.json** . Para esto utiliza el **API (BULK)**.

Para cargar los datos en el nuevo índice, se intentó conectar la consola de Dev Tools con el archivo 'Registros.json', sin embargo no se encontró ninguna manera de realizarlo. Como el

archivo no es grande se decidió pegar todos los json a la consola de Dev Tools. Resultando en el siguiente código para la carga de los datos.

```
History Settings Help
1 POST_bulk
2 {"index":{"_index":"log_consultas","_id":1}}
3 {"@timestamp":"2010-05-15T22:00:54","estado_consulta":"consumo",
  "servicio":"consulta","administrador":"Juan Carlos",
  "consultas_realizadas":52}
4 {"index":{"_index":"log_consultas","_id":2}}
5 {"@timestamp":"2010-05-15T12:55:04","estado_consulta":"consumo",
  "servicio":"modificacion","administrador":"Juan Lara",
  "consultas_realizadas":10}
6 {"index":{"_index":"log_consultas","_id":3}}
7 {"@timestamp":"2010-05-15T14:56:48","estado_consulta":"consumo",
  "servicio":"consulta","administrador":"Juan Lara",
  "consultas_realizadas":20}
8 {"index":{"_index":"log_consultas","_id":4}}
9 {"@timestamp":"2010-05-15T22:33:34","estado_consulta":"error",
  "servicio":"modificacion","administrador":"Juan Carlos",
  "consultas_realizadas":65}
10 {"index":{"_index":"log_consultas","_id":5}}
11 {"@timestamp":"2010-05-15T18:36:57","estado_consulta":"consumo",
  "servicio":"consulta","administrador":"Carlos Lara",
  "consultas_realizadas":5}
12 {"index":{"_index":"log_consultas","_id":6}}
13 {"@timestamp":"2010-05-15T11:21:05","estado_consulta":"informativo",
  "servicio":"borrado","administrador":"Juan Carlos"}

Click to send request
200 - 323 ms
1 {
  "took": 25,
  "errors": false,
  "items": [
    {
      "index": {
        "_index": "log_consultas",
        "_id": "1",
        "_version": 1,
        "result": "created",
        "_shards": {
          "total": 2,
          "successful": 2,
          "failed": 0
        },
        "_seq_no": 0,
        "_primary_term": 1,
        "status": 201
      }
    },
    {
      "index": {
        "_index": "log_consultas",
        "_id": "2",

```

## Realizar búsquedas sobre el índice

1. Obtener el número de registros con **estado\_consulta** igual a error y consumo.

En el primer intento se utilizó el siguiente código. Este código se compone de **query**, **size** y **aggregation**.

1. **query** obtiene los documentos que cumplen ciertos criterios, en este caso, el campo **estado\_consulta** debe tener la palabra error.
2. **size** nos permite indicar cuantos documentos es necesario mostrar. En el código indicamos que ningún documento debe ser mostrado.
3. **aggregation** nos permite resumir el query que se obtuvo. En este caso se utilizó **value\_count** el cual calcula el número de registros.

```
History Settings Help
1 GET log_consultas/_search
2 {
3   "query": {
4     "match": {
5       "estado_consulta": {
6         "query": "error"
7       }
8     }
9   },
10  "size": 0,
11
12  "aggs": {
13    "consultas": {
14      "value_count": {
15        "field": "estado_consulta"
16      }
17    }
18  }
19 }

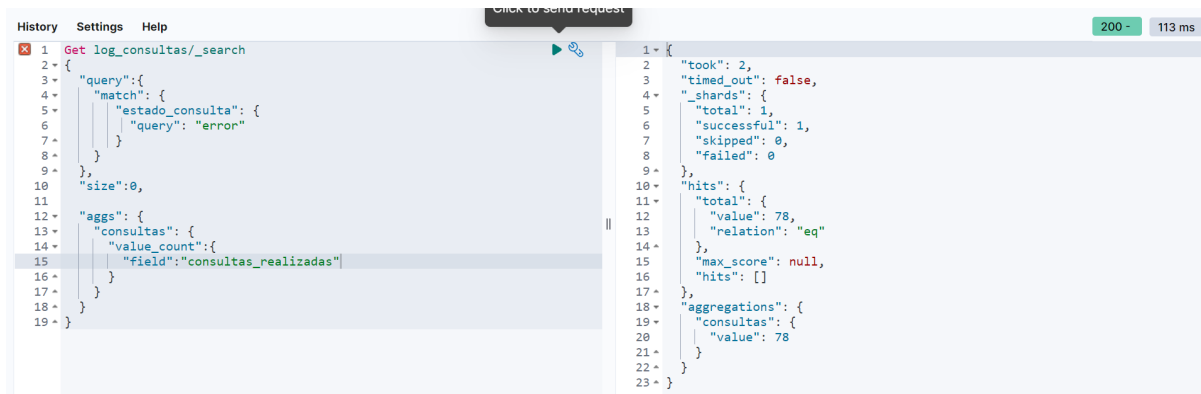
Click to send request
400 - 550 ms
1 {
  "error": {
    "root_cause": [
      {
        "type": "illegal_argument_exception",
        "reason": "Text fields are not optimised for operations that require per-document field data like aggregations and sorting, so these operations are disabled by default. Please use a keyword field instead. Alternatively, set fielddata=true on [estado_consulta] in order to load field data by uninverting the inverted index. Note that this can use significant memory."
      }
    ],
    "type": "search_phase_execution_exception",
    "reason": "all shards failed",
    "phase": "query",
    "grouped": true,
    "failed_shards": [
      {
        "shard": 0,
        "index": "log_consultas",
        "node": "xuxag-wtQXusP9YdGYWfRA",
        "reason": {

```

Sin embargo se obtuvo el siguiente error. En cual se indica que los campos de **text** no están optimizados para operaciones como **aggregations**. El error sugiere dos posibles soluciones. Cambiar el tipo de dato a **keyword** o cambiar la configuración del campo para que estas operaciones sean posibles, sin embargo esta última solución implica el uso de mayor cantidad de memoria.

Se encontró una mejor solución, simplemente al usar otro campo donde las **aggregations** fueran posibles. Debido a que contar registros en el campo **estado\_consulta** es

equivalente a contar registros en **consultas\_realizadas** y que este campo si se pueden realizar **aggregations**. Se decidió usar el siguiente codigo.



```
1 Get log_consultas/_search
2 {
3   "query": {
4     "match": {
5       "estado_consulta": {
6         "query": "error"
7       }
8     }
9   },
10  "size": 0,
11  "aggs": {
12    "consultas": {
13      "value_count": {
14        "field": "consultas_realizadas"
15      }
16    }
17  }
18 }
19 }
```

```
1 {
2   "took": 2,
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": 78,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 78
21    }
22  }
23 }
```

Donde se obtuvo que hay un total de 78 registros donde el campo **estado\_consulta** es error.

Otra posible solución fue utilizar un **bucket aggregations**, los cuales son un tipo especial de **aggregations** que agrupa los documentos. Se utilizó **categorize\_text** la cual agrupa los documentos y nos da algunas estadísticas sobre estos, como el número de documentos en el **bucket** (grupo). Para utilizar **categorize\_text** solo es necesario indicar sobre qué campo se debe agrupar los documentos, en este caso **estado\_consulta**. Obteniendo la misma información que en el problema anterior.



```
1 Get log_consultas/_search
2 {
3   "size": 0,
4   "aggs": {
5     "EstadoConsultas": {
6       "categorize_text": {
7         "field": "estado_consulta"
8       }
9     }
10  }
11 }
```

```
16 "hits": []
17 },
18 "aggregations": {
19   "EstadoConsultas": {
20     "buckets": [
21       {
22         "doc_count": 117,
23         "key": "informativo",
24         "max_matching_length": 12
25       },
26       {
27         "doc_count": 104,
28         "key": "consumo",
29         "max_matching_length": 7
30       },
31       {
32         "doc_count": 78,
33         "key": "error",
34         "max_matching_length": 5
35       }
36     ]
37   }
38 }
39 }
```

## 2. Obtener el número de registros realizados por el administrador Juan Lara.

Utilizando un razonamiento similar al primer problema se utilizó las dos soluciones anteriores. Sin embargo estas soluciones dieron resultados distintos. En la primera solución usando el **match** se obtuvo 299 registros, en la segunda solución utilizando **categorize\_text** se obtuvo un total de 98 registros.

```
History Settings Help
1 Get log_consultas/_search
2 {
3   "query": {
4     "match": {
5       "administrador": {
6         "query": "Juan Lara"
7       }
8     }
9   },
10  "size": 0,
11
12  "aggs": {
13    "consultas": {
14      "value_count": {
15        "field": "consultas_realizadas"
16      }
17    }
18  }
19 }

1 {
2   "took": 2,
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": 299,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 299
21    }
22  }
23 }
```

Primera solución 299 registros

```
History Settings Help
1 Get log_consultas/_search
2 {
3   "size": 0,
4   "aggs": {
5     "Estado_Consultas": {
6       "categorize_text": {
7         "field": "administrador"
8       }
9     }
10  }
11 }

16 "hits": []
17 },
18 "aggregations": {
19   "Estado_Consultas": {
20     "buckets": [
21       {
22         "doc_count": 110,
23         "key": "Carlos Lara",
24         "max_matching_length": 12
25       },
26       {
27         "doc_count": 98,
28         "key": "Juan Lara",
29         "max_matching_length": 9
30       },
31       {
32         "doc_count": 91,
33         "key": "Juan Carlos",
34         "max_matching_length": 12
35       }
36     ]
37   }
38 }
39 }
```

Segunda solución 98 registros

Esta diferencia ocurre debido a que en el cuerpo del **query** se utiliza **match**, el cual utiliza tokens para analizar el texto en lugar de buscar un término exacto. En el índice actual solo hay 3 administradores, Carlos Lara, Juan Lara y Juan Carlos, al utilizar **match**, la palabra Juan Lara, hace match con Carlos Lara, con Juan Lara y con Juan Carlos, debido a que todas tienen alguna palabra en común con Juan Lara, por lo tanto el query regresa todos los documentos.

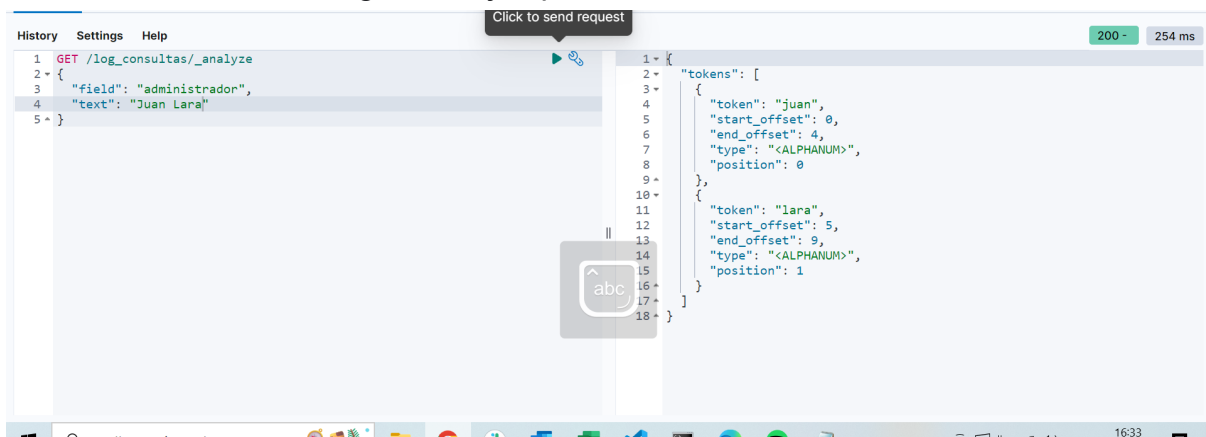
Existe otra alternativa para **match** y es **term** el cual realiza una búsqueda exacta sin embargo se encontró el siguiente error.

```
History Settings Help
1 Get log_consultas/_search
2 {
3   "query": {
4     "term": {
5       "administrador": {
6         "value": "Juan Lara"
7       }
8     }
9   },
10  "size": 0,
11
12  "aggs": {
13    "consultas": {
14      "value_count": {
15        "field": "consultas_realizadas"
16      }
17    }
18  }
19 }

1 {
2   "took": 8,
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": 0,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 0
21    }
22  }
23 }
```

Ninguno de los administradores hizo match con la palabra de búsqueda Juan Lara.

Este error ocurre debido a que al cargar los datos, los campos que son **text**, son “analizados” por Elasticsearch, es decir el texto se separa en pequeños tokens, como se muestra en el siguiente ejemplo.



```
1 GET /log_consultas/_analyze
2 {
3   "field": "administrador",
4   "text": "Juan Lara"
5 }

1 {
2   "tokens": [
3     {
4       "token": "juan",
5       "start_offset": 0,
6       "end_offset": 4,
7       "type": "<ALPHANUM>",
8       "position": 0
9     },
10    {
11      "token": "lara",
12      "start_offset": 5,
13      "end_offset": 9,
14      "type": "<ALPHANUM>",
15      "position": 1
16    }
17  ]
18 }
```

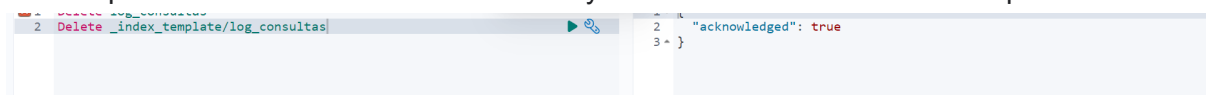
Por lo cual **term** no considera que el texto ‘Juan Carlos’ sea igual algún administrador en el índice con el cual trabajamos

Más información puede ser encontrada en el siguiente link

[https://www.elastic.co/guide/en/elasticsearch/guide/current/finding\\_exact\\_values.html#term\\_filter\\_with\\_text](https://www.elastic.co/guide/en/elasticsearch/guide/current/finding_exact_values.html#term_filter_with_text)

Una solución a este problema es cambiar el tipo de datos de **text** a **keyword**. Debido a que los campos que son **text** parecen no ser la mejor elección para los campos con los se trabaja, se decido cambiar su tipo de dato por **keyword**, el cual parece estar diseñado para manejar ‘categorías’.

Hay varias maneras de cambiar el tipo de dato de un campo sin embargo, la solución más sencilla para este caso fue eliminar el índice y volverlo a crear con el nuevo tipo de dato.



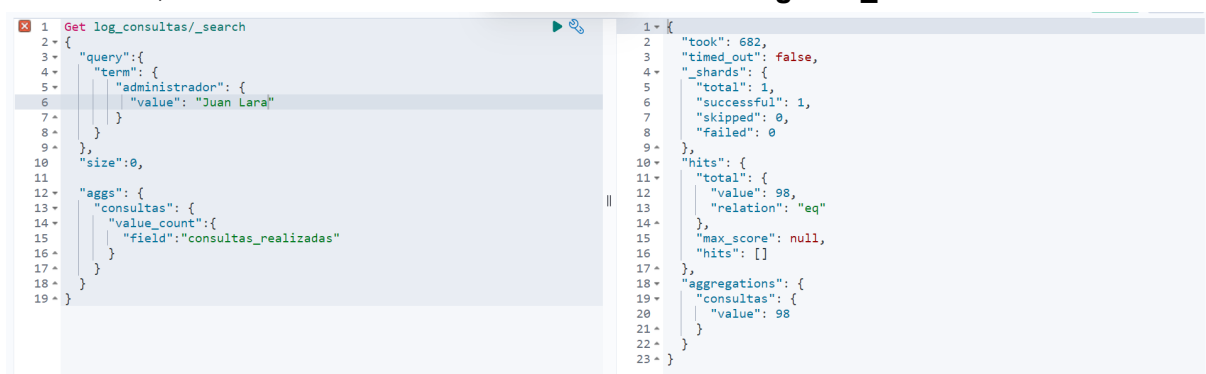
```
2 Delete _index_template/log_consultas

2 {
3   "acknowledged": true
4 }
```

La eliminación del índice y su template

Se replicó los pasos del primer ejercicio, solo cambiando la palabra **text** por **keyword**.

Realizando estos cambios, al usar **term**, el número de registros con el administrador Juan Lara son 98, como se había notado anteriormente con **categorize\_text**.

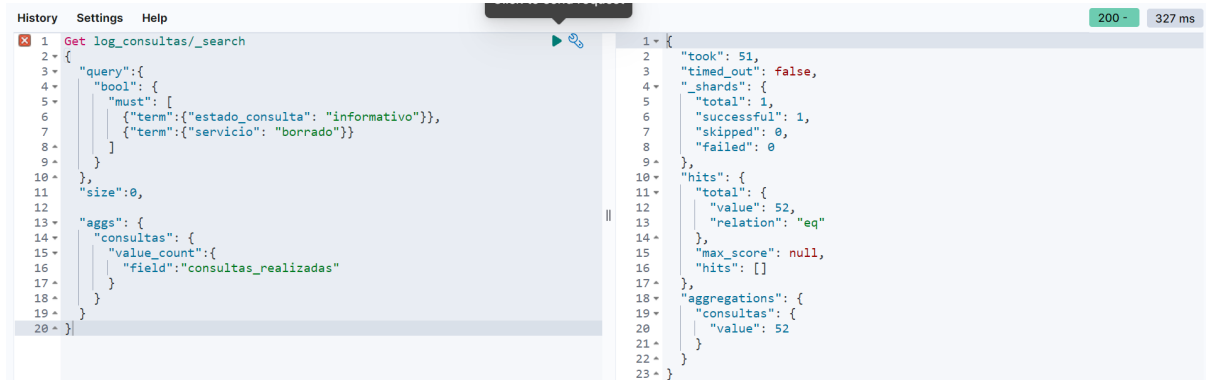


```
1 Get log_consultas/_search
2 {
3   "query": {
4     "term": {
5       "administrador": {
6         "value": "Juan Lara"
7       }
8     }
9   },
10  "size": 0,
11  "aggs": {
12    "consultas": {
13      "value_count": {
14        "field": "consultas_realizadas"
15      }
16    }
17  }
18 }

1 {
2   "took": 682,
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": 98,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 98
21    }
22  }
23 }
```

3.- Obtener el número de registros con **estado\_consulta** igual a informativo y **servicio** igual a borrado

Para este problema se utilizó **bool**, el cual nos permite determinar criterios que se deben cumplir, como establecer las dos condiciones que pide el problema.

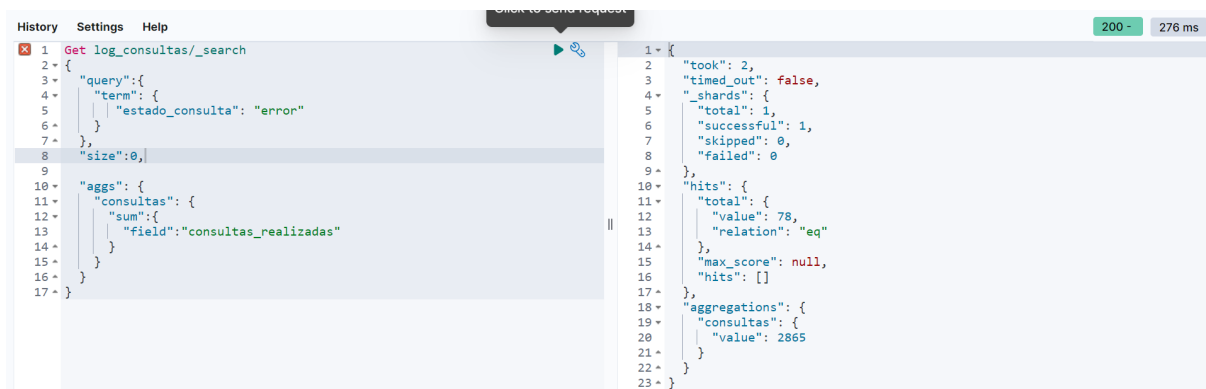


```
History Settings Help
1 Get log_consultas/_search
2 {
3   "query": {
4     "bool": {
5       "must": [
6         { "term": { "estado_consulta": "informativo" } },
7         { "term": { "servicio": "borrado" } }
8       ]
9     }
10  },
11  "size": 0,
12  "aggs": {
13    "consultas": {
14      "value_count": {
15        "field": "consultas_realizadas"
16      }
17    }
18  }
19 }
20 }
```

```
1 {
2   "took": 51,
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": 52,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 52
21    }
22  }
23 }
```

4.- Obtener la suma de los valores en **consultas\_realizadas** con **estado\_consulta** igual a error.

Para este problema solo es necesario cambiar la **aggregation** en vez de usar **value\_count**, se utilizara **sum**.



```
History Settings Help
1 Get log_consultas/_search
2 {
3   "query": {
4     "term": {
5       "estado_consulta": "error"
6     }
7   },
8   "size": 0,
9   "aggs": {
10    "consultas": {
11      "sum": {
12        "field": "consultas_realizadas"
13      }
14    }
15  }
16 }
17 }
```

```
1 {
2   "took": 2,
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": 78,
13      "relation": "eq"
14    },
15    "max_score": null,
16    "hits": []
17  },
18  "aggregations": {
19    "consultas": {
20      "value": 2865
21    }
22  }
23 }
```

# Realizar un tablero para visualizar información de empleados

Los **Index patterns** han sido renombrados como **data views**



elastic

Platform▼

Use cases▼

Pricing

Customers▼

Resources▼

Company▼

[Elastic Docs](#) > [Kibana Guide \[8.3\]](#) > [Deleted pages](#)

## Index patterns has been renamed to data views.



This content has moved. Refer to [Create a data view](#).

Al igual que vizualize, ahora hace referencia a **Dashboard**,



elastic

Platform▼

Use cases▼

Pricing

Customers▼

Resources▼

Company▼

[Elastic Docs](#) > [Kibana Guide \[8.3\]](#) > [Deleted pages](#)

## Visualize



This content has moved. Refer to [Dashboard](#).

This content has moved. Refer to [Dashboard](#).

Los **data view** pueden crearse en la hoja de **Dashboard**,

### Create data view

Name

Enter an index pattern that matches one or more data sources. Use an asterisk (\*) to match multiple characters. Spaces and the characters , / ? " , < , > , | are not allowed.

Timestamp field

Select a timestamp field for use with the global time filter.

[Show advanced settings](#)

✕ Close

Create data view

✓ Your index pattern matches 1 source.

log\_consultas

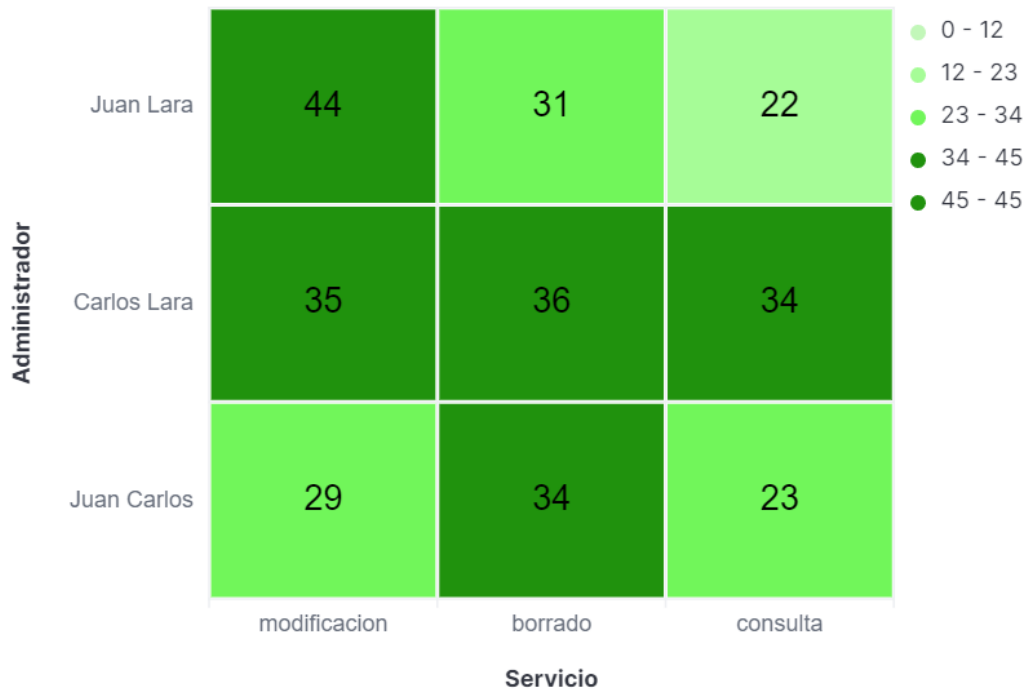
Index

Rows per page: 10 ▼

Para la creación del dashboard simplemente me familiarice con los controles, observe que hacía cada cosa, hasta terminar las gráficas.



## examen



## examen2

