# Ejercicio 5 – Índice Autocomplete

|  |  |
| --- | --- |
| Apellidos, Nombre: | Campos de Padua, Alfonso |
| Puntuación Final: |  |

Ejercicio 5 - Creación de índice y subida de datos con BulkAPI + Sense o Curl

Partiendo de este conjunto de documentos bien definidos:

PUT /indice/tipo/\_bulk

{"index":{"\_id":1}}

{"DocID":"ABCD-A-1234-JZ3", "precio":10, "marca":"mi marca primera"}

{"index":{"\_id":2}}

{"DocID":"ABCD-A-1234-LZ5", "precio":20, "marca":"mi marca privilegiada"}

{"index":{"\_id":3}}

{"DocID":"ABDC-B-5678-LP5", "precio":30, "marca":"mi marca pringada"}

{"index":{"\_id":4}}

{"DocID":"ABDC-B-5678-LP5", "precio":20, "marca":"mi marca prima"}

{"index":{"\_id":5}}

{"DocID":"DCAB-C-4321-LP5", "precio":10, "marca":"mi marca primaria es mejor"}

{"index":{"\_id":6}}

{"DocID":"DCAB-C-4321-LP5", "precio":20, "marca":"mi marca priorizada"}

{"index":{"\_id":7}}

{"DocID":"DACB-D-8765-VT7", "precio":30, "marca":"mi marca previsora"}

{"index":{"\_id":8}}

{"DocID":"DACB-D-8765-VT7", "precio":20, "marca":"mi marca precargada"}

{"index":{"\_id":9}}

{"DocID":"DBAC-E-1111-DO8", "precio":50, "marca":"mi segunda marca primera"}

Y sabiendo que serán utilizados en una aplicación que tiene un campo de búsqueda que necesita funcionalidad autocomplete, define los settings del índice, el analizador necesario así como el mapping de documentos para permitir que una búsqueda de este tipo:

GET /indice/tipo/\_search

{

"query": {

"match": {

"marca": "mi marca prim"

}

}

}

Retorne todos los valores que comienzan por "mi marca prim"

Pista - los edge\_ngram filters permiten hacer cosas que podrían ser útiles para este caso.

El índice creado:

PUT documentos

{

"settings": {

"analysis": {

"filter": {

"marca\_ngram\_filter": {

"type": "edge\_ngram",

"min\_gram": 2,

"max\_gram": 15

}

},

"analyzer": {

"marca\_custom\_analyzer": {

"type": "custom",

"tokenizer": "standard",

"filter": [

"lowercase",

"marca\_ngram\_filter"

]

}

}

}

},

"mappings": {

"documento": {

"properties": {

"DocID": {

"type": "string"

},

"precio": {

"type": "integer"

},

"marca": {

"type": "string",

"index\_analyzer": "marca\_custom\_analyzer",

"search\_analyzer": "standard"

}

}

}

}

}

Introducimos los documentos:

PUT documentos/documento/\_bulk

{"index":{"\_id":1}}

{"DocID":"ABCD-A-1234-JZ3", "precio":10, "marca":"mi marca primera"}

{"index":{"\_id":2}}

{"DocID":"ABCD-A-1234-LZ5", "precio":20, "marca":"mi marca privilegiada"}

{"index":{"\_id":3}}

{"DocID":"ABDC-B-5678-LP5", "precio":30, "marca":"mi marca pringada"}

{"index":{"\_id":4}}

{"DocID":"ABDC-B-5678-LP5", "precio":20, "marca":"mi marca prima"}

{"index":{"\_id":5}}

{"DocID":"DCAB-C-4321-LP5", "precio":10, "marca":"mi marca primaria es mejor"}

{"index":{"\_id":6}}

{"DocID":"DCAB-C-4321-LP5", "precio":20, "marca":"mi marca priorizada"}

{"index":{"\_id":7}}

{"DocID":"DACB-D-8765-VT7", "precio":30, "marca":"mi marca previsora"}

{"index":{"\_id":8}}

{"DocID":"DACB-D-8765-VT7", "precio":20, "marca":"mi marca precargada"}

{"index":{"\_id":9}}

{"DocID":"DBAC-E-1111-DO8", "precio":50, "marca":"mi segunda marca primera"}

Comporbamos que todos los documentos e han introducido correctamente:

GET documentos/documento/\_search

Realizamos la búsqueda:

GET documentos/documento/\_search

{

"query": {

"match": {

"marca":{

"query":"mi marca prim",

"operator": "and"

}

}

}

}

Y nos retorna los que cumplen todos los edge ngrams:

{

"took": 3,

"timed\_out": false,

"\_shards": {

"total": 5,

"successful": 5,

"failed": 0

},

"hits": {

"total": 4,

"max\_score": 0.65325016,

"hits": [

{

"\_index": "documentos",

"\_type": "documento",

"\_id": "1",

"\_score": 0.65325016,

"\_source": {

"DocID": "ABCD-A-1234-JZ3",

"precio": 10,

"marca": "mi marca primera"

}

},

{

"\_index": "documentos",

"\_type": "documento",

"\_id": "4",

"\_score": 0.5148823,

"\_source": {

"DocID": "ABDC-B-5678-LP5",

"precio": 20,

"marca": "mi marca prima"

}

},

{

"\_index": "documentos",

"\_type": "documento",

"\_id": "9",

"\_score": 0.5148823,

"\_source": {

"DocID": "DBAC-E-1111-DO8",

"precio": 50,

"marca": "mi segunda marca primera"

}

},

{

"\_index": "documentos",

"\_type": "documento",

"\_id": "5",

"\_score": 0.23252454,

"\_source": {

"DocID": "DCAB-C-4321-LP5",

"precio": 10,

"marca": "mi marca primaria es mejor"

}

}

]

}

}

NOTA: El resultado “mi segunda marca primera” tiene la palabra segunda que no incumple ningún edge ngram, por eso se retorna. Esto es más parecido a lo que ocurre cuando buscamos en google, donde lo importante es devolver las búsquedas que contienen, no las que tienen en el orden exacto las palabras. Entiendo que la “gracia” de usar edge ngrams es obtener este comportamiento, aunque el enunciado pida “todos los valores que comienzan por”.