Missing fields:

All fields in elastic search are opitional

Searches automatically handle missing fields

Mapping parameters: (overview)

Format parameter:

Used to customise the format for data fields

“strict\_data\_opitional\_time “//”epoch\_mills”

Using java’s date format syntax

Eg:”dd//mm/yyyy”

Using built in formats.

Eg:”epoch\_second”

“type”: “date”,

“format”:”dd//MM//yyyy

Properties parameter:

Detines nested fields for object and nested fields

Coerce parameter

Used to enable or disable coercion of value

“type”: “float” “settings”: {

“corece”: false (or) “index.mapping.coerce”:false

}

At index leaves

Doc\_values:

Used for aggregation, sorting or scrpitics

Disabiling it may save disk space(whrn not required)

Norms parameter:

Normalizarion factors used for relevel scoring

Norms can be disabled to save disk space (for fields that not require scoring)

Index parameter:

Disables indexing for field vals are still stores in \_source

Null\_value parameter:

NULL values cannot be indexd or searched .

Can be used to replace NULL values with other NULL vals can be replaced with same data type

“type”.keyword “

“null\_values”:”NULL”

Copy\_to parameter:

Used to copy multiple field values into group fied

Values are copied not terminated

Updating existing mappings:

We cannot change the mapping

We cannot also remove field mappings

The updated by query API can be used to reclaim disk space

Reindex API:

Post /\_reindex

{

“source”:{

“index” : “source\_index”

}

“dest”:{

“index”:”destination\_index”

}

}

\_source data types

The data types doesnot reflect how the values are indexed

Modify documents during reindexing is to supply a script that is run for each document

“script”:{

“source”: “ “ “

If(ctx.\_source.product.\_id!= null){

Ctx.\_source.product\_id=ctx.\_source.product\_id.to string();

}

“ “ “

}

Noe deleting the documents with the new index and reindex the documents again

We can delete\_by\_query API

To delete the documents and index them again

-----------------------------------------------------------------------------------------

Now

Filed mapping cannot be deleted

Fields can be left out while indexing documents

Source fitering (removing fields)

Post /\_reindex

{

“source”:{

“index”:”review”,

“\_source:”[content , created\_at ,”rating”]

}

“dest”:{

“index”:”review\_new”

}

}

Changing a fields name:

“script”:{

“source”: ” ” ”

Ctx.source.comment=ctx.\_source.remove(“content”);

“ “ “

}

“ “ “

If(ctx.sourcce.rating<4.0){

Ctx.op=”noop”:#can also be set to “delete”

}

“ “ “

Handing version contilts

A snapshort is created before reindexing doc

Batching & throtting:

The reindex API performs operations inbus

It uses scroll API internally

Field Aliases:

PUT /review/\_mappig

{

“properties”:{

“comment”:{

“type”:”alias”

“path”:”content”

}

}

Similar to field aliases,Elastic search also support index aliases(when dealing with large data values)

Multi-field-mapping to run aggregation on text values we need to use the keyword” data type and we also need “text” data type

PUT/ multi-field-test

{

“mappings”:{

“properties”:{

“description”:{

“type”:”text”

“fields”:{

Anything-----🡪”keyword”:{

“type”:”keyword”

}

}

}

}

}