Spring Data Elastic Search

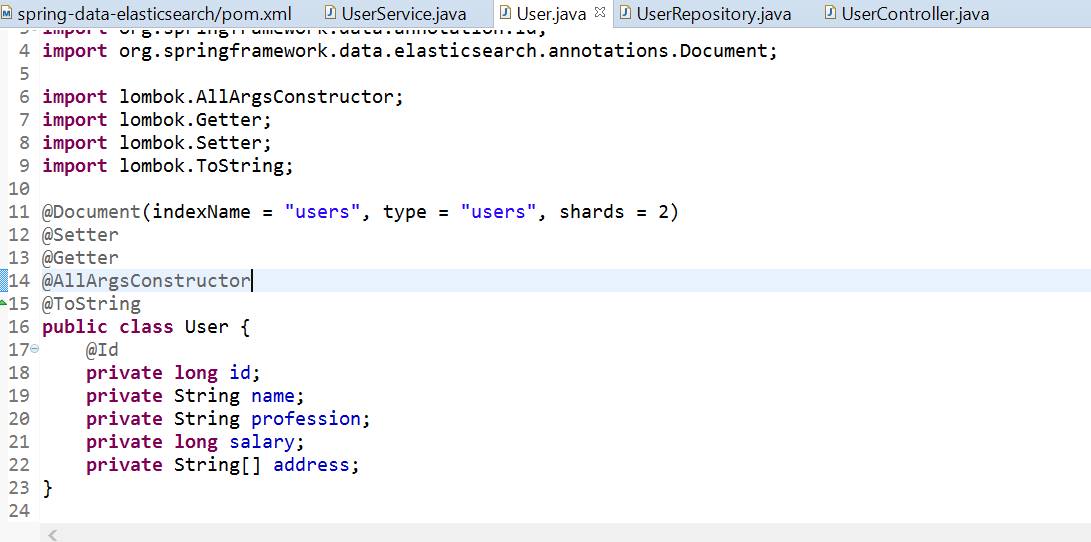
[BASANT KUMAR HOTA](https://www.facebook.com/dugulu)·[TUESDAY, DECEMBER 26, 2017](https://www.facebook.com/notes/sriman_java_gyan_mantra/spring-data-elastic-search/1534886083263094/)

**What is spring data?**

Spring Data helps avoid boilerplate code. For example, if we define a repository interface that extends the CRUDRepository or JpaRepository or ElasticSearchRepository interface provided by spring*,* CRUD operations for the corresponding model class will be made available by default.Additionally, there is no need to write an implementation of the repository interface.

Assume for Example:

Model class:



User Model which is my Business object BO

So how can I achieve spring data inbuilt crud features so for that I need to create one Repository interface which should be extends from spring provided predefined repository map with Model entity and serialize id .as below

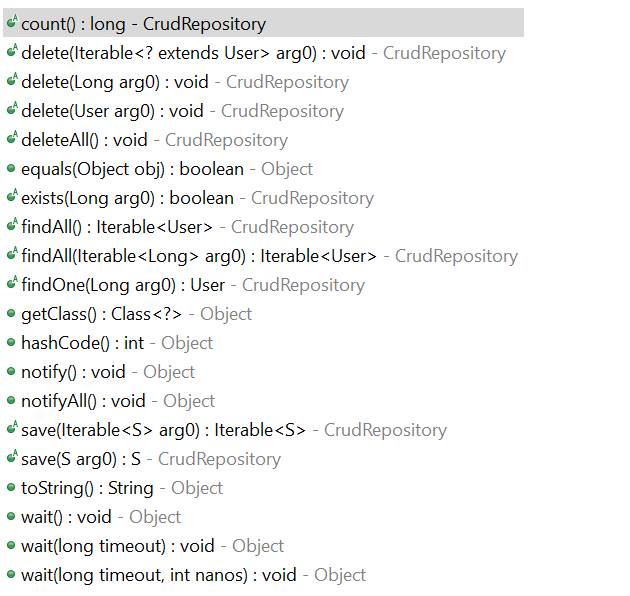
1. Public interface UserRepository extends CrudRepository<User, Long> {}

2. Public interface UserRepository extends JpaRepository<User, Long> {}

3. Public interface UserRepository extends ElasticsearchRepository<User, Long> {}

If you mark properly in generic I passed my model class and id data type of my model class then, this model will register with spring data then we can get all inbuilt method to perform CRUD operation

All operations:



inbuilt methods for CRUD Operation

Some more features are there suppose you want to perform own search operation like you want to search a User object based on name or any field from your entity

You can do like below

Public User findByName (String name);

Internally spring data helps us to generate query internally for above method like below

SELECT \* FROM USER WHERE NAME=?

Here findBy is common syntax next is your field name of your entity

Same like assume I want to fetch list of user based on salary

Public List<User> findBySalary (long salary);

Internal query: SELECT \* FROM USER WHERE SALARY=?

If you want to manually write your query for this particular requirement then you can use @Query Annotation as below

@Query (“SELECT \* FROM USER WHERE SALARY=?”)

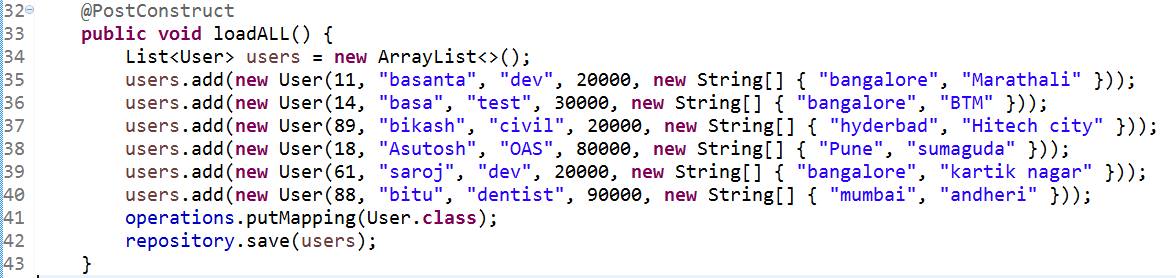
Public List<User> findBySalary (long salary);

**ELASTIC-SEARCH:**

**So think now if we have already multiple approaches to search Field using spring data then why we should go for new concept Elastic Search ?**

In single word Elasticsearch is a highly scalable open-source which can be used for data store, text search and analytics engine

My dummy data which am storing in Elastic search index



Data set which am storing when my application start

***Enhance features:***

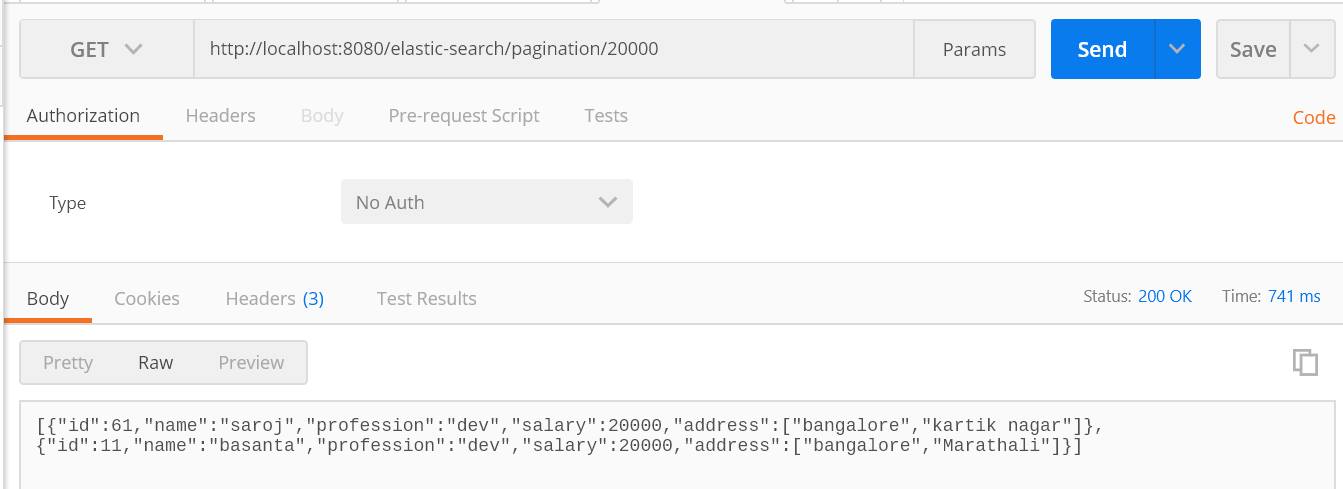
1. Provides built-in support for pagination and sorting. As ElasticsearchRepository extends PagingAndSortingRepository

Ex: if you will mark dummy data set I have 3 records having salary 20000 but I want to apply pagination here means I want to display only 2 records. To show u that pagination is working properly .u can change it based on your business // using inbuilt pagination

Public List<User> getPaginationData (long salary) {

return repository.findBySalary (salary, new PageRequest (0, 2)).getContent();}

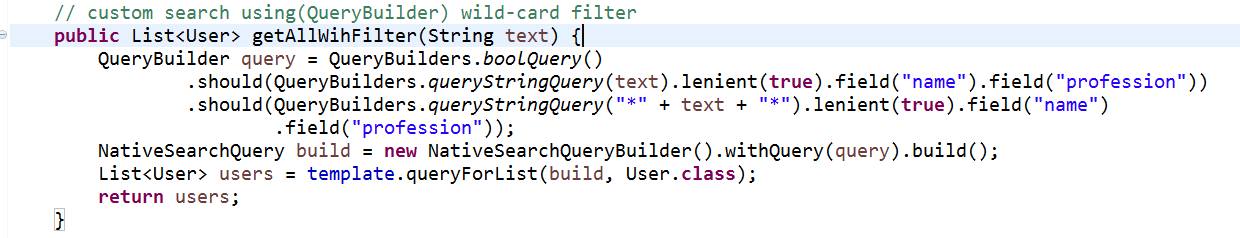
Test: see the record count (it gives me only 2 records)



Pagination Response

2. It provide custom filter to search record from database like assume I have 1lakh records in DB and I want to fetch users whose name is starts from “b” then it will gives list of record for my search criteria using wild card character

For example : Check in my dummy dataset I have 4 records whose name starts with “b” so I want to display those records, same you can do using Query part but spring support us to do it in service end using Elastic search

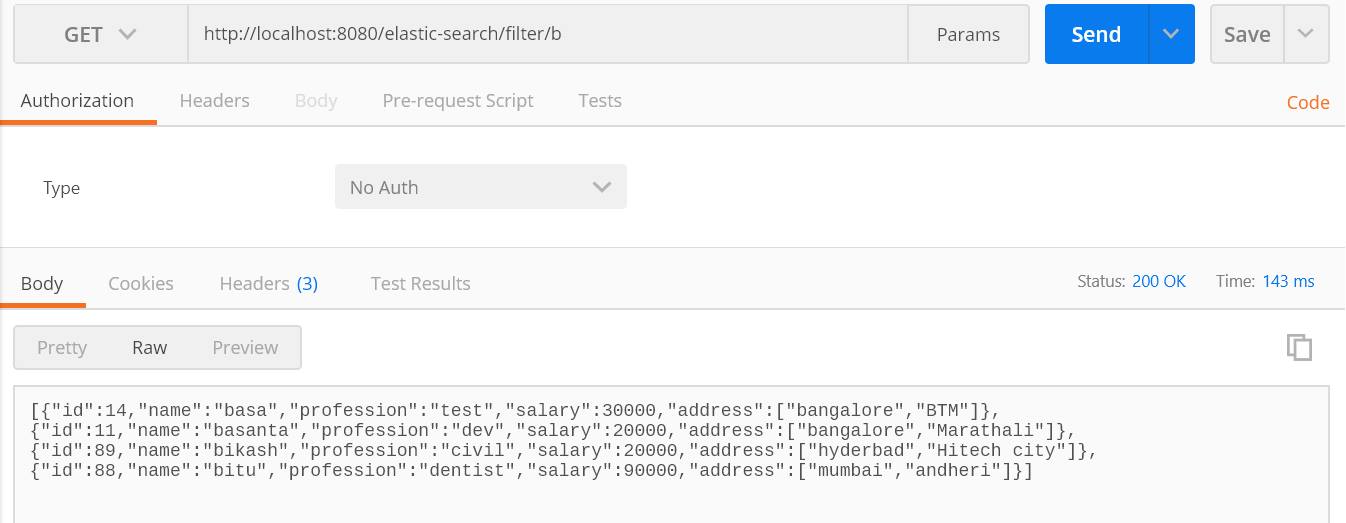


Search Query using QueryBuilder

Not only name we can add multiple field for search as per above snippet of code I use 2 field name and profession

So here my search criteria is to fetch List of user whose name and profession matches with my input

Test: (Multi field search criteria)

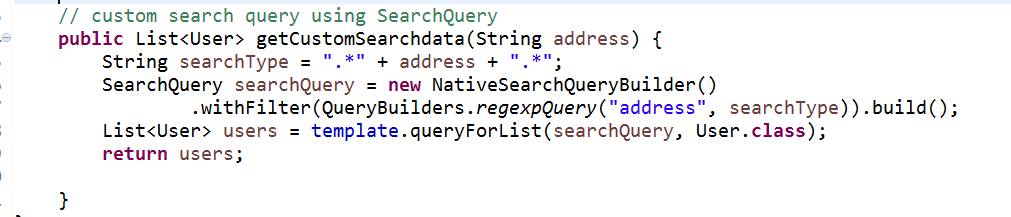


All records whose name starts from b

3. We can do the custom Search using Search Query Regular expression

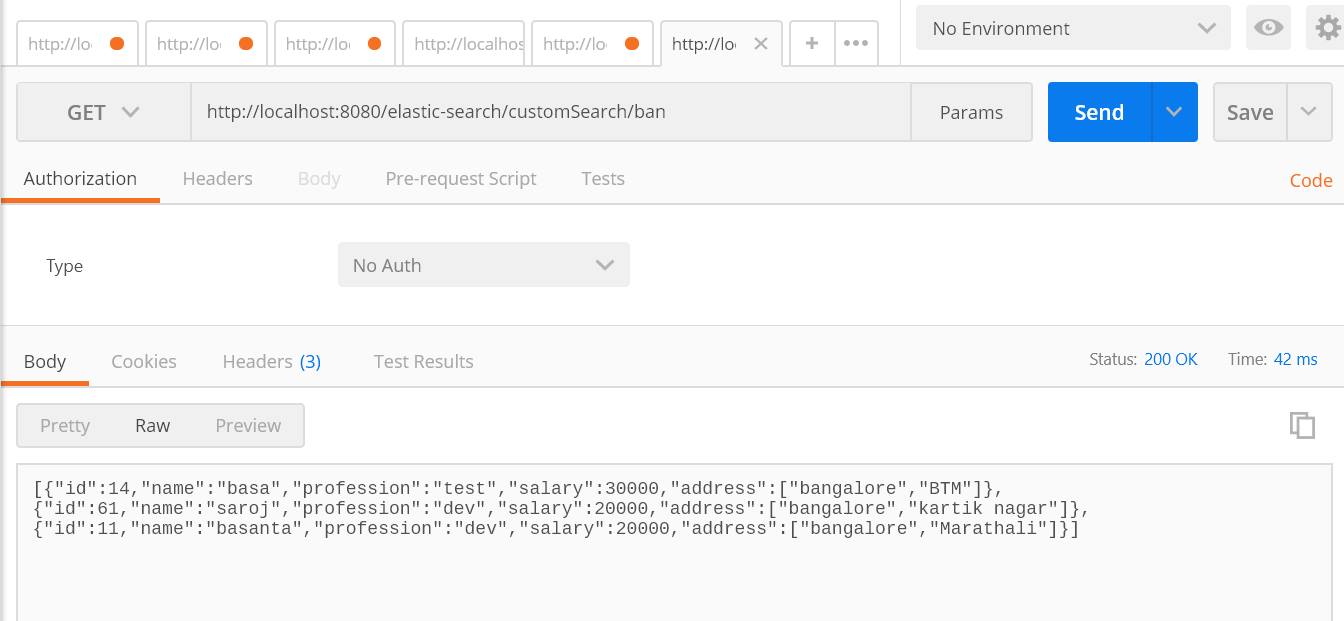
For Example assume I want to display those records whose address matches with my input.

For testing am searching for ‘Bangalore’ location but I can give input like either “b” or “ba” or “ban” like this I want to give and expecting result using regular expression



Search Query using SaerchQuery

Test: (it should give me 3 records as per my dummy data set)



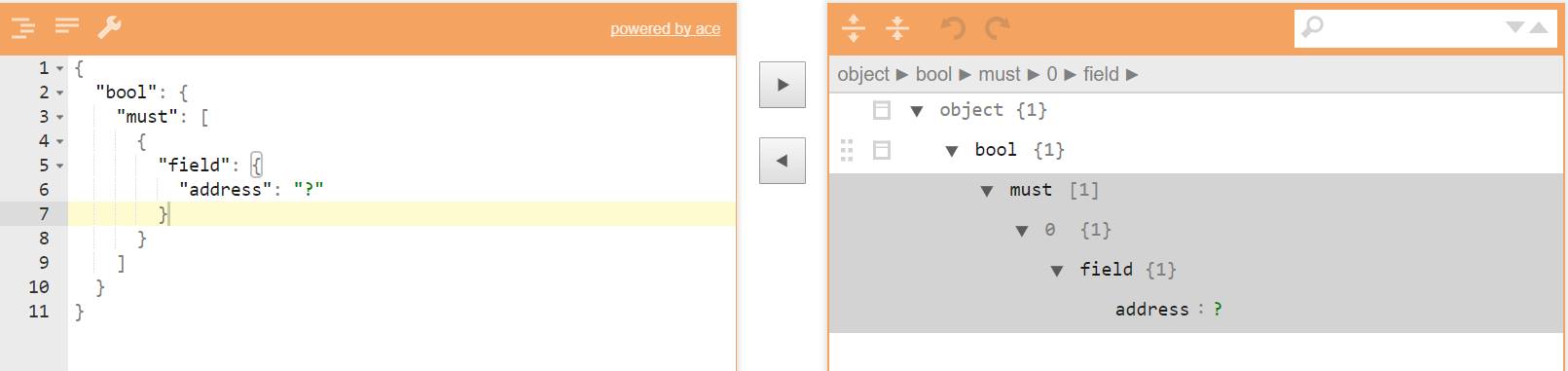
All records whose address starts from ban

4. The main advantages the data stored in elastic search index in the format of JSON and the query which generated by Elastic search is also in Same format as below

Assume am fetching list of records based on address

Internal SQL: Select \* from User where address=?

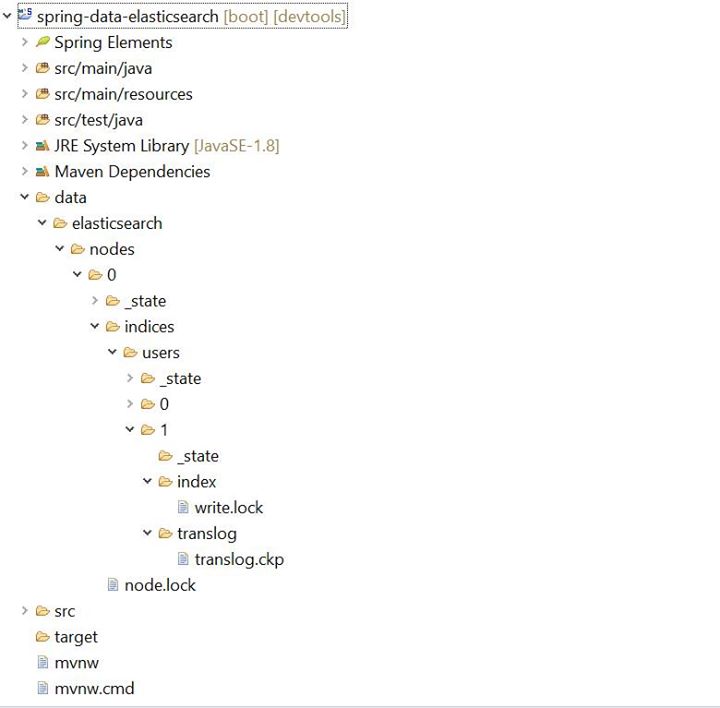
Internal JSON Query:



Internal JSON Query

5. Using @Document annotation, we indicate that instances of this Model class should be stored in Elasticsearch in an index called user, and with a document type of user. Documents with many different types can be stored in the same index. (see model class above screen shot)

Please see below screen shot, it confirm u that data not storing in any in memory DB like H2, Derby or redies it internally creating index structure



Generate folder in runtime

This is all about Spring Elastic Search API features and internal flow and process, some more predefined methods and classes are provided in this API which I will update soon, now I just pick out main features.

Thanks,

Basant hota