



**Academia de Studii Economice, Bucureşti**

**Facultatea de Cibernetică, Statistică și Informatică Economică**

**Specializarea: Statistică și Previziunea Economică**

# **Proiect de practică**

**Domeniu: Tehnologia informației (IT), cu specializare pe DevOps, Observabilitate și Analiza de Date.**

**Numele studentului:** Scarlat Anda-Maria-Cristina

## **Introducere**

Tehnologia ELK (Elasticsearch, Logstash, Kibana) reprezintă o platformă modernă și puternică pentru colectarea, analiza și vizualizarea datelor, în special a logurilor generate de aplicații, servere sau sisteme IT.

Elasticsearch este motorul de căutare și analiză, responsabil cu stocarea și interogarea rapidă a datelor.

Logstash este componenta de procesare, care preia date din diverse surse, le transformă și le trimită către Elasticsearch.

Kibana oferă o interfață vizuală intuitivă, unde utilizatorii pot explora datele prin grafice, tabele, hărți și dashboard-uri interactive.

Împreună, aceste trei componente formează o platformă scalabilă, flexibilă și open-source, folosită pe scară largă pentru monitorizare, audit, securitate cibernetică, business intelligence și multe altele. ELK Stack face ca datele brute să devină ușor de înțeles și acționabile.

## **Secțiunea 1- Pregatirea mașinii virtuale**

Pentru a putea realiza o platformă pentru analiză și vizualizare date folosind tehnologia ELK, avem nevoie de o mașină virtuală create de la zero.

Eu am ales să folosesc Ubuntu cu hypervisor de Linux.

### **Pașii:**

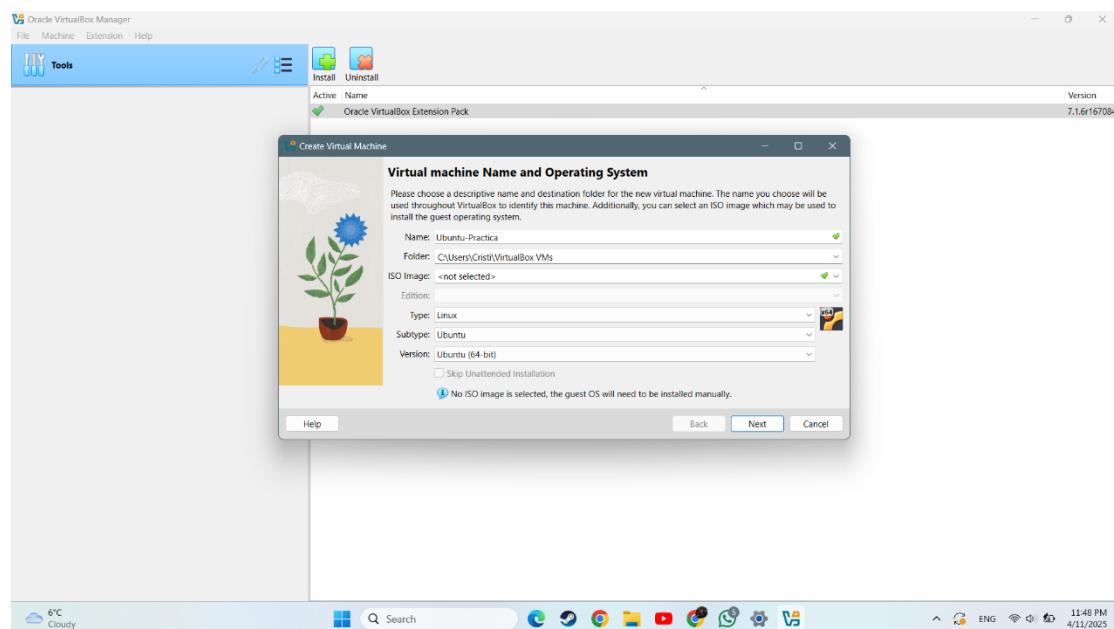
1. Instalarea VirtualBox și Extension Pack
2. Crearea mașinii virtuale

Name: UbuntuPractica

Type: Linux

Version: Ubuntu (64-bit)

Specificații: 2 CPU, 4 GB RAM, 20 GB spațiu.

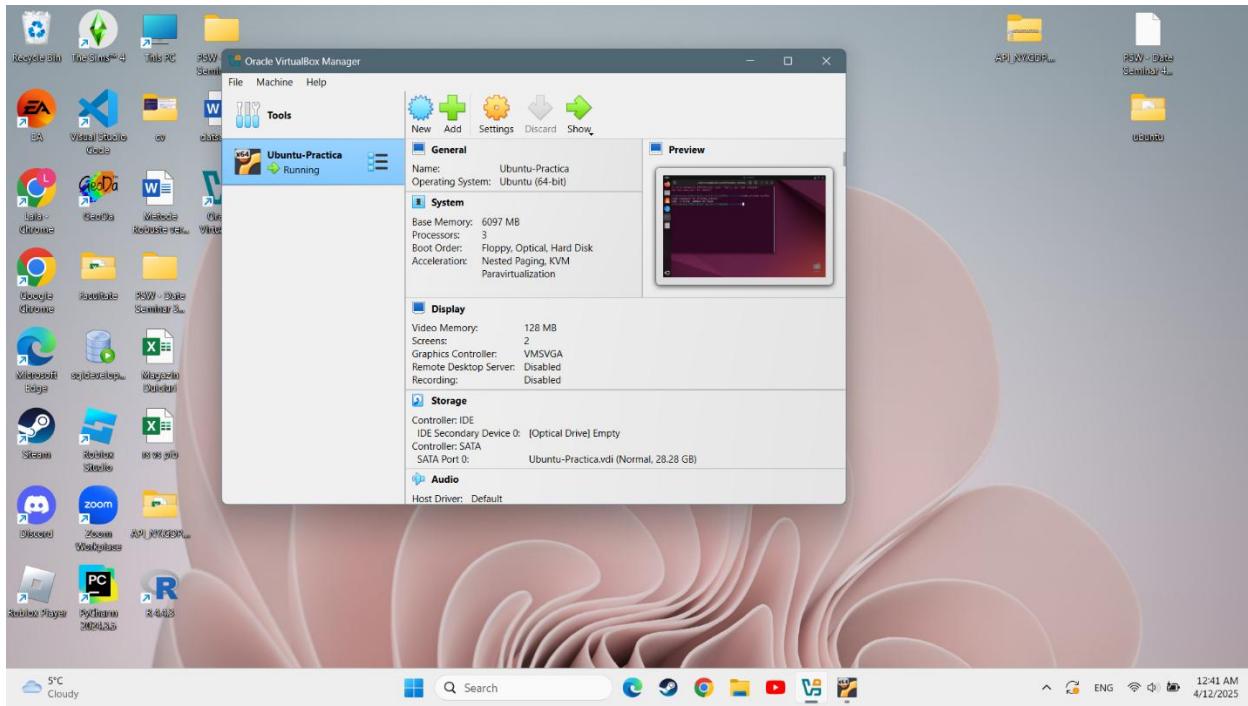


3. Setări suplimentare pentru VM

4. Instalarea Ubuntu

5. Finalizarea

După instalare, am restartat VM-ul



## **Secțiunea 2-Instalare Docker și Docker compose cu instanță de Elasticsearch și Kibana**

### **Instalare Docker și Docker Compose:**

#### Pasul 1:

Înainte de a instala Dockerul, a fost nevoie să verificăm dacă mașina noastră virtuală se află cu toate pachetele updatate.

```
sudo apt update
```

#### Pasul 2:

După ce am verificat și updatat unde era cazul, a trebuit să instalăm pachetele preliminare necesare sistemului.

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

#### Pasul 3:

Am adăugat cheia GPG a dockerului.

Ce este o cheie GPG?

- GPG vine de la GNU Privacy Guard, și e un sistem de criptare care ajută la:
- Verificarea autenticității pachetelor pe care le descarci (adică, să știi că nu sunt modificate de cineva rău-intenționat).
- Asigurarea integrității (că fișierul nu a fost corrupt sau modificat pe drum).
- Semnături digitale – un fel de semnătură virtuală a unei companii care zice „da, asta e versiunea oficială a pachetului nostru”.

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o  
/usr/share/keyrings/docker-archive-keyring.gpg
```

Pasul 4:

Am adăugat un docker repository.

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-  
keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

Pasul 5:

Rulăm comanda update încă o dată.

Pasul 6:

Folosim repository-ul oficial pentru docker.

```
apt-cache policy docker-ce
```

Pasul 7:

Instalăm docker.

```
sudo apt install docker-ce
```

Pasul 8:

Verificăm statusul serviciului de docker.

```
sudo systemctl status docker
```

Pasul 9:

Rulăm hello world docker.

```
sudo docker run hello-world
```

Pasul 10:

Instalăm docker compose.

```
sudo apt install docker-compose
```

**Instanță Elasticsearch și Kibana:**

Pentru a realiza aceste două instanțe, am fost nevoită să configerez un fisier **docker-compose.yml**.

Conținutul acestuia l-am editat folosind comanda fie nano, fie vi.

Sintaxa de cod folosită:

---

```
version: '3.8'
```

```
services:
```

```
  elasticsearch:
```

```
    image: docker.elastic.co/elasticsearch/elasticsearch:8.12.2
```

```
    container_name: elasticsearch
```

```
    environment:
```

```
      - discovery.type=single-node
```

```
      - xpack.security.enabled=false
```

- bootstrap.memory\_lock=true
- ES\_JAVA\_OPTS=-Xms512m -Xmx512m

ports:

- "9200:9200"

**volumes:**

- esdata:/usr/share/elasticsearch/data

ulimits:

**memlock:**

soft: -1

hard: -1

kibana:

image: docker.elastic.co/kibana/kibana:8.12.2

container\_name: kibana

environment:

- ELASTICSEARCH\_HOSTS=http://elasticsearch:9200
- xpack.security.enabled=false

ports:

- "5601:5601"

depends\_on:

- elasticsearch

**volumes:**

esdata:

---

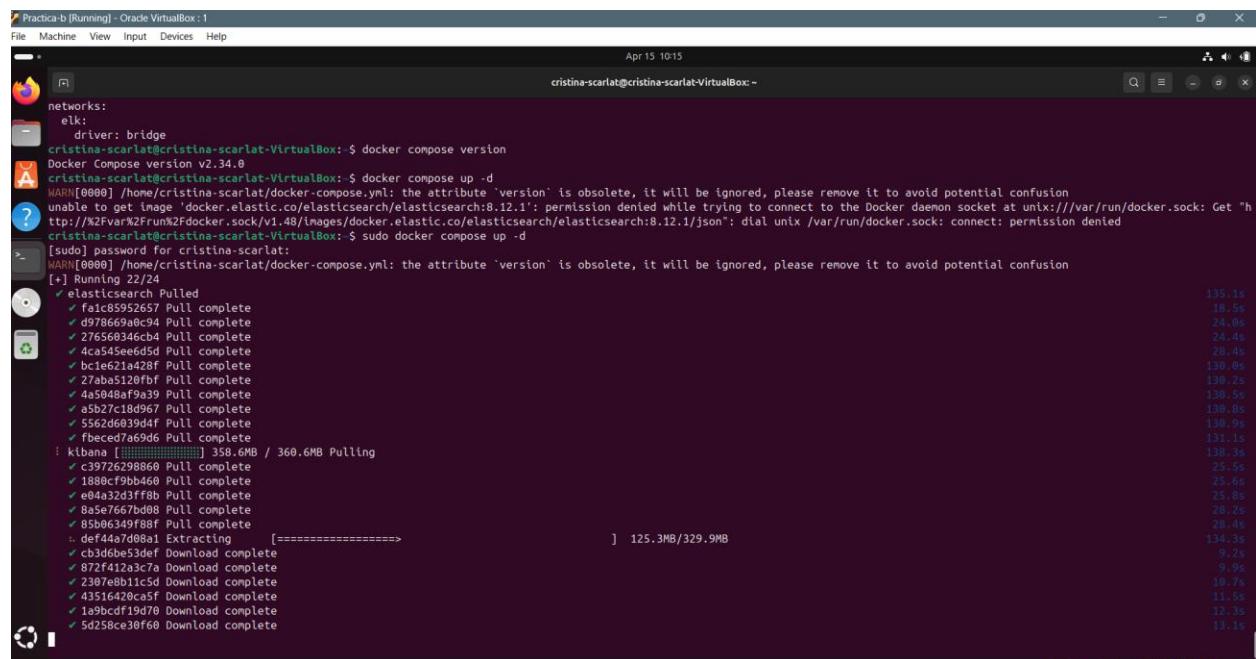
Instruție pornire docker: sudo docker compose up -d

**Instrucțiune pentru afișarea instanțelor din docker: sudo docker compose ps**

**Instrucțiune oprire docker: sudo docker compose down**

Inițial sintaxa codului nu conținea comenziile legate de volum, fapt ce a condus la folosirea unor zone de memorie temporare. Ca și consecință, deși eu îmi creasem 2 indecsi și aceștia erau funcționali și populați cu date, odată ce am oprit și repornit container-ul, ei s-au șters. Astfel, cu ajutorul volumului, datele sunt salvate în sistem și rămân intacte.

Mai exact, un volum Docker e un loc în afara containerului unde Docker-ul salvează datele.



The screenshot shows a terminal window titled "Practica-b [Running] - Oracle VirtualBox : 1". The command "docker compose version" is run, showing Docker Compose version v2.34.0. Then "docker compose up -d" is run, which fails due to permission denied errors when trying to connect to the Docker daemon socket. A password is entered for the user "cristina-scarlat". Finally, "docker compose up -d" is run again, showing a list of services being pulled, including Elasticsearch and Kibana. The logs indicate that Elasticsearch is pulling 360.6MB and Kibana is pulling 329.9MB.

```
Practica-b [Running] - Oracle VirtualBox : 1
File Machine View Input Devices Help
Apr 15 10:15
cristina-scarlat@cristina-scarlat-VirtualBox: ~

networks:
  elasticsearch:
    driver: bridge
cristina-scarlat@cristina-scarlat-VirtualBox: $ docker compose version
Docker Compose version v2.34.0
cristina-scarlat@cristina-scarlat-VirtualBox: $ docker compose up -d
WARN[08080] /home/cristina-scarlat/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
unable to get image 'docker.elastic.co/elasticsearch/elasticsearch:8.12.1': permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "https://index.docker.io/v1/images/docker.elastic.co/elasticsearch/elasticsearch:8.12.1/json": dial unix /var/run/docker.sock: connect: permission denied
cristina-scarlat@cristina-scarlat-VirtualBox: $ sudo docker compose up -d
[sudo] password for cristina-scarlat:
WARN[08080] /home/cristina-scarlat/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
[-] Running 2/24
  ✓ elasticsearch Pulled
    ✓ fa1c8595265f Pull complete
    ✓ d978669a0c94 Pull complete
    ✓ 276569346cb0 Pull complete
    ✓ 4ca545eed45d Pull complete
    ✓ bc1e621a0428f Pull complete
    ✓ 27ab5120fbf1 Pull complete
    ✓ 4a5048af939 Pull complete
    ✓ a5b27c18d967 Pull complete
    ✓ 5562d68394af Pull complete
    ✓ fbeced7a69de Pull complete
  ! kibana [=====[ 358.6MB / 360.6MB Pulling
    ✓ c39726298860 Pull complete
    ✓ 1880cf9bb460 Pull complete
    ✓ e04a32d3ff8b Pull complete
    ✓ 8a5e7667b008 Pull complete
    ✓ 85b06349f08f Pull complete
    ✓ def4447d08a1 Extracting [=====] 125.3MB/329.9MB
      ✓ cb3d6be53def Download complete
      ✓ 872f412a3c7a Download complete
      ✓ 2307e8b11c5d Download complete
      ✓ 43516420ca5f Download complete
      ✓ 1a9bcdf19d70 Download complete
      ✓ 5d258ce30f60 Download complete
      ✓ 13.1s
```

În final, am verificat dacă cele 2 host-uri funcționează în browser:

- <http://localhost:9200> pentru Elasticsearch
- <http://localhost:5601> pentru Kibana

### Secțiunea 3- Crearea unui index și popularea cu date a acestuia. Comenzi de navigare și manipulare a datelor.

- Crearea unui index și popularea lui cu date

Ce este un index?

În general, un index este o structură de date care îți permite să găsești rapid informațiile căutate, fără să parcurgi tot conținutul, ceea ce îl face ca în Elasticsearch să reprezinte un „container” pentru documente care au o structură similară.

Pentru a înțelege mai bine am făcut o analogie cu SQL Developer și elementele sale:

#### SQL vs ELK

- Tabelele = Indexuri (indices).
- Liniile = Documente (documents).
- Coloanele = Câmpuri (fields) în documente.

Pentru a crea un index putem folosi atât terminalul, cât și Dev Tools.

Comandă terminal:

---

```
curl -X PUT "localhost:9200/index_studenti" -H 'Content-Type: application/json' -d'  
{  
  "mappings": {  
    "properties": {  
      "Nr_matricol": {  
        "type": "integer"  
      },  
      "Nume": {  
        "type": "text"  
      },  
      "Varsta": {  
        "type": "integer"  
      },  
      "Media": {  
        "type": "float"  
      }  
    }  
  }  
}
```

```
 },
  "tara_natala": {
    "type": "text"
  }
}
}'
```

---

Comandă Dev tools:

---

```
PUT /index_studenti_bursieri
{
  "mappings": {
    "properties": {
      "Nr_matricol": {
        "type": "integer"
      },
      "Nume": {
        "type": "text"
      },
      "Varsta": {
        "type": "integer"
      },
      "Media": {
        "type": "float"
      }
    }
  }
}
```

```
},  
  "tara_natala": {  
    "type": "text"  
  }  
}  
}  
}
```

---

Putem observa că cele două sintaxe sunt similare, ele diferențându-se prin modul în care se începe. În plus, apare comanda *mapping* care oferă informații cu privire la schema indexului..

Practica-b [Running] - Oracle VirtualBox : 1

File Machine View Input Devices Help

Apr 15 10:46

cristina-scarlat@cristina-scarlat-VirtualBox:~

```
>      }
>      }
>      }
> }
> ^C
cristina-scarlat@cristina-scarlat-VirtualBox:~$ curl -X PUT "localhost:9200/index_stu
denti_bursieri" -H 'Content-Type: application/json' -d'
{
  "mappings": {
    "properties": {
      "Nr_matricol": {
        "type": "integer"
      },
      "Nume": {
        "type": "text"
      },
      "Varsta": {
        "type": "integer"
      },
      "Media": {
        "type": "float"
      },
      "tara_natala": {
        "type": "text"
      }
    }
  }
}
{"acknowledged":true,"shards_acknowledged":true,"index":"index_studenti_bursieri"}cri
cristina-scarlat@cristina-scarlat-VirtualBox:~$
```

The screenshot shows the Elasticsearch Dev Tools Console interface. In the left sidebar, there are icons for History, Settings, Variables, and Help. The main area has tabs for Console, Search Profiler, Grok Debugger, and Painless Lab (BETA). The Console tab is active. The history panel shows a single PUT request:

```
PUT /index_studentl
{
  "mappings": {
    "properties": {
      "Nr_matricol": {
        "type": "integer"
      },
      "Nume": {
        "type": "text"
      },
      "Varsta": {
        "type": "integer"
      },
      "Media": {
        "type": "float"
      }
    }
  }
}
```

The response panel shows the successful response:

```
{"acknowledged": true,
"shards_acknowledged": true,
"index": "index_studentl"}
```

The screenshot shows the Elasticsearch Dev Tools Settings view for the index `index_studenti_bursieri`. The interface includes tabs for JSON, Raw Data, and Headers, with the JSON tab selected. It features a toolbar with Save, Copy, Collapse All, Expand All, and Filter JSON buttons.

The settings for the index are displayed as follows:

```
index_studenti_bursieri:
  aliases: {}
  mappings:
    properties:
      Media:
        type: float
      Nr_matricol:
        type: integer
      Nume:
        type: text
      Varsta:
        type: integer
      tara_natala:
        type: text
  settings:
    index:
      routing:
        allocation:
          include:
            _tier_preference: "data_content"
        number_of_shards: "1"
        provided_name: "index_studenti_bursieri"
        creation_date: "1744703044655"
        number_of_replicas: "1"
        uuid: "rdAvCTAkQqiNamNEbfaalWQ"
    version:
      created: "8500010"
```

Cum populăm indexul cu date?

La fel ca în cazul creării, putem folosi atât terminalul, cât și Elastic Dev Tools.

```
cristina-scarlat@cristina-scarlat-VirtualBox:~$ curl -X POST "localhost:9200/index_studenti/_doc" -H 'Content-Type: application/json' -d'
{
    "Nr_matricol": 212,
    "Nume": "Albert Nicuale",
    "Varsta": 22,
    "Media": 8.75,
    "tara_natala": "Romania"
}
'

{"_index":"index_studenti","_id":"okp50JYBkARzw_M-FYdj","_version":1,"result":"created","_shards":{"total":2,"successful":1,"failed":0},"_seq_no":1,"_primary_term":2}cris
```

Mai mult de atât, am mai creat 3 indecsăi: pentru profesori, pentru cursuri și pentru departamente.

---

PUT /index\_profesori

```
{
    "mappings": {
        "properties": {
            "Nume": {
                "type": "text"
            },
            "Birou": {
                "type": "integer"
            },
            "Id_profesor": {
                "type": "integer"
            }
        }
    }
}
```

```
    }  
}  
  
}
```

POST /index\_profesori/\_doc

```
{  
  "Nume": "Selena Morningson",  
  "Birou": 6000,  
  "Id_prefesor": 3010  
}
```

---

//cream un index al cursurilor

PUT /index\_cursuri

```
{  
  "mappings": {  
    "properties": {  
      "Denumire": {  
        "type": "text"  
      },  
      "Nr_credite": {  
        "type": "integer"  
      },  
    }  
  }  
}
```

```
"Reprezentant_curs": {
    "type": "integer"
}
}
}
}
```

//il populam cu date

```
POST /index_cursuri/_doc
{
    "Denumire": "Optional Matematica Aplicata",
    "Nr_credite": 0,
    "Reprezentat_curs": 3002
}
```

---

//cream un index cu departamente

PUT /index\_departamente

```
{
    "mappings": {
        "properties": {
            "Criteriu_birou": {
                "type": "text"
            },
            "Id_Dep": {
                "type": "integer"
            }
        }
    }
}
```

"Denumire":

```
{  
  "type": "text"  
}  
}  
}
```

---

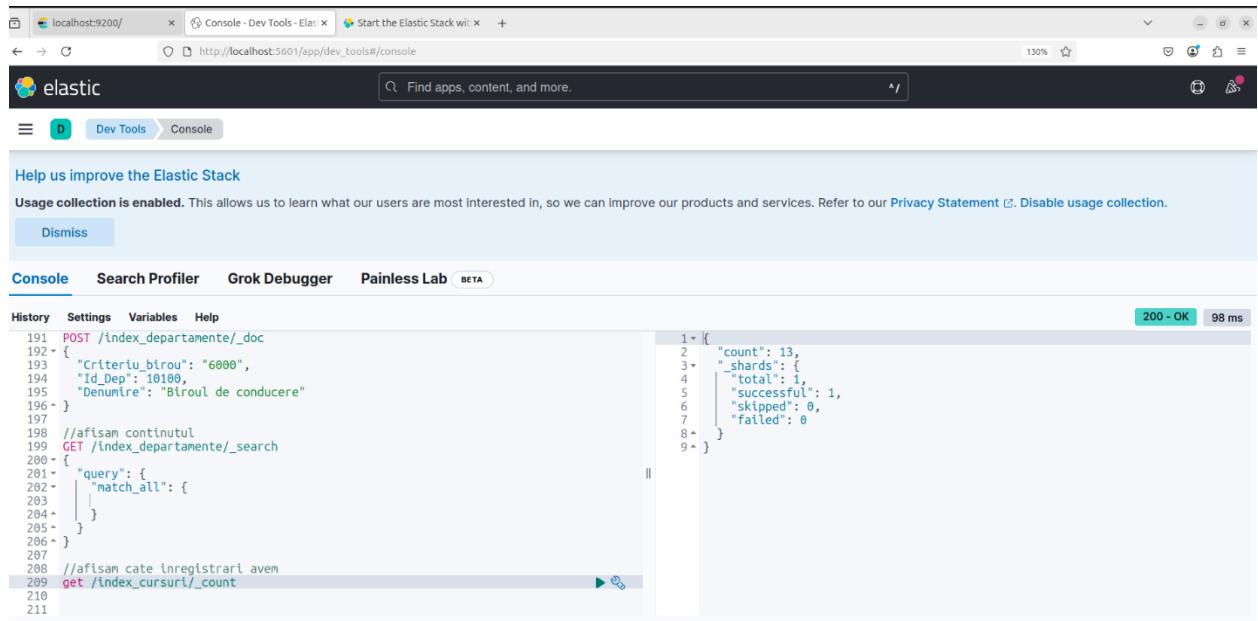
- **Comenzi**

Dev Tools oferă o multitudine de comenzi atât pentru navigare, definirea sau manipularea datelor. Însă, acestea nu sunt la fel de intuitive ca în alte limbaje de gestionare a bazelor de date.

Pentru început am folosit comenzi simple și ușor de înțeles.

- **Comanda get**  
-se folosește pentru a prelua un document după ID-ul lui exact
- **Comanda count**  
`get /index_cursuri/_count`

-oferă numărul de documente dintr-un index.



The screenshot shows a browser window with the URL `http://localhost:5601/app/dev_tools#/console`. The page title is "Console - Dev Tools - Elastic". The main area displays a command-line interface with the following session:

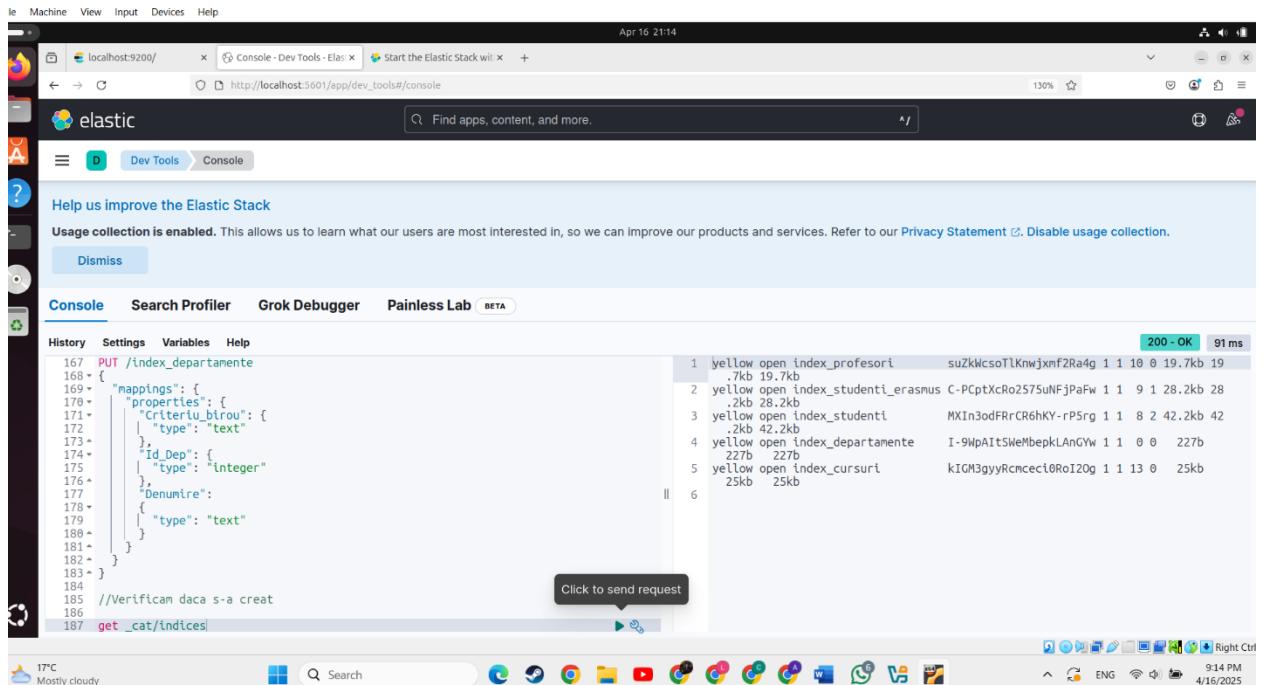
```
1 POST /index_departamente/_doc  
2 {  
3   "Criteriu_birou": "6000",  
4   "Id_Dep": 10189,  
5   "Denumire": "Biroul de conducere"  
6 }  
7  
8 //afisam continutul  
9 GET /Index_departamente/_search  
10 {  
11   "query": {  
12     "match_all": {  
13     }  
14   }  
15 }  
16  
17 //afisam cate inregistrari avem  
18 get /index_cursuri/_count  
19  
20 211
```

To the right of the command history, the response is shown in a JSON format:

```
1+ {  
2   "count": 13,  
3+   "_shards": {  
4     "total": 1,  
5     "successful": 1,  
6     "skipped": 0,  
7     "failed": 0  
8   }  
9 }
```

The status bar at the bottom right indicates "200 - OK" and "98 ms".

- Comanda `_cat/indices`  
-afisează indecșii creați de noi  
`get _cat/indices`



The screenshot shows a Firefox browser window with the URL `http://localhost:5601/app/dev_tools#/console`. The page title is "elastic". The main area is the "Console" tab, which displays the following code and its execution results:

```

167 PUT /index_departamente
168 {
169   "mappings": {
170     "properties": {
171       "criteriu_birou": {
172         "type": "text"
173       },
174       "id_dep": {
175         "type": "integer"
176       },
177       "Denumire": {
178         "type": "text"
179       }
180     }
181   }
182 }
183 //Verificam daca s-a creat
184
185 //Afiseaza tot continutul
186
187 get _cat/indices

```

A tooltip "Click to send request" points to the button at the bottom of the code editor.

The results section shows a table with the following data:

Index	Type	Name	Size	Score
1	yellow	open index_profesori	suZkWcs0TlKnwJxmF2Ra4g	1 1 10 0 19.7kb 19
2	yellow	open index_studenti_erasmus	C-PCptXcRo2575NFjPaFw	1 1 9 1 28.2kb 28
3	yellow	open index_studenti	MXIn3odFRrCR6hKY-rP5rg	1 1 8 2 42.2kb 42
4	yellow	open index_departamente	I-9WpAIt5WeMbepkLAnGYw	1 1 0 0 227b
5	yellow	open index_cursuri	kIGM3gyyRcmceci0RoI20g	1 1 13 0 25kb
6				

At the bottom of the browser window, the status bar shows: 17°C Mostly cloudy, 9:14 PM, 4/16/2025.

- Comanda `search`  
Caută și afisează documentele dintr-un index (sau mai multe), în funcție de o condiție (query).

GET /index\_profesori/\_search

```
{
  "query": {
    "match_all": {} //afisează tot conținutul
  }
}
```

The screenshot shows the Elasticsearch Dev Tools Console interface. In the left panel, there is a code editor with a JSON query:

```

17+     }
18+
19+ }
20
21 GET index_studenti_erasmus/_search
22 {
23   "query": {
24     "match_all": {}
25   }
26 }
27
28
29 get _cat/indices
30 GET index_studenti/_count
31
32 POST /index_studenti/_doc
33 {
34   "Nr_matricol": 407,
35   "Nume": "Laur Ion",
36   "Varsta": 22,
37   "Materiale": "A"

```

In the right panel, the results of the search are displayed:

```

1+ [
2   "took": 7,
3   "timed_out": false,
4   "_shards": {
5     "total": 1,
6     "successful": 1,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": [
11    {
12      "total": {
13        "value": 9,
14        "relation": "eq"
15      },
16      "max_score": 1,
17      "hits": [
18        {
19          "_index": "index_studenti_erasmus",
20          "_id": "gledoJYBku0IB_HByuVx",
21          "_score": 1,
22          "_source": {

```

At the bottom of the interface, there is a toolbar with various icons and a status bar showing "19°C", "8:32 PM", and other system information.

De asemenea, acest lucru se poate face și în bash

```
curl -X GET "http://localhost:9200/index_cursuri/_search?pretty" -H 'Content-Type: application/json'
```

```
cristina-scarlat@cristina-scarlat-VirtualBox:~$ curl -X GET "http://localhost:9200/index_cursuri/_search?pretty" -H 'Content-Type: application/json'
{
  "took" : 9,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : [
    {
      "total" : {
        "value" : 13,
        "relation" : "eq"
      },
      "max_score" : 1.0,
      "hits" : [
        {
          "_index" : "index_cursuri",
          "_id" : "UGS-P5YBu0IVGB9OYE6V",
          "_score" : 1.0,
          "_source" : {
            "Denumire" : "Algebra Liniara",
            "Nr_credite" : 6,
            "Reprezentat_curs" : 3001
          }
        },
        {
          "_index" : "index_cursuri",
          "_id" : "UWS-P5YBu0IVGB90k07t",
          "_score" : 1.0,
          "_source" : {
            "Denumire" : "Analiza matematica",
            "Nr_credite" : 6,
            "Reprezentat_curs" : 3002
          }
        }
      ]
}
```

- Comanda **delete**

-șterge documentele din index pe baza id-ului

```
DELETE index_studenti_erasmus/_doc/iVeiOJYBku0IB_HByuVx
```

```
DELETE index_studenti/_doc/fleaOJYBku0IB_HBjeUm
```

The screenshot shows the Elasticsearch Dev Tools Console interface. On the left, there is a code editor window displaying a series of JSON objects (lines 53 to 72). Lines 63 and 72 show a POST and a DELETE request respectively to the '/index\_studenti\_erasmus/\_doc' endpoint. On the right, the results of the DELETE request are displayed in a results panel. A button labeled 'Click to send request' is visible between the code editor and the results panel.

```

53 "Media": {
54   "type": "float"
55 },
56 "Tara_natala": {
57   "type": "text"
58 }
59 }
60 }
61 }
62
63 POST /index_studenti_erasmus/_doc
64 {
65   "Nr_matricol": 674,
66   "Nume": "Daria Mla",
67   "Varsta": 23,
68   "Media": 8.8,
69   "Tara_natala": "America"
70 }
71
72 DELETE /index_studenti_erasmus/_doc/_id/1VeI0JYBku0IB_HByuVx
    
```

```

1+ {
2   "index": "index_studenti_erasmus",
3   "_id": "1VeI0JYBku0IB_HByuVx",
4   "_version": 2,
5   "result": "deleted",
6   "_shards": {
7     "total": 2,
8     "successful": 1,
9     "failed": 0
10  },
11  "_seq_no": 12,
12  "_primary_term": 2
13 }
    
```

## Secțiunea 4- Backup și Restore

### Backup

ElasticSearch nu își creează backup-ul manual, aşa că este nevoie să-l creăm noi. În plus, acesta poartă denumirea de snapshot.

Pentru a putea realiza procesul de backup a fost nevoie să creez un director în /home special pentru această acțiune, dar și să modific yml specific dockerului.

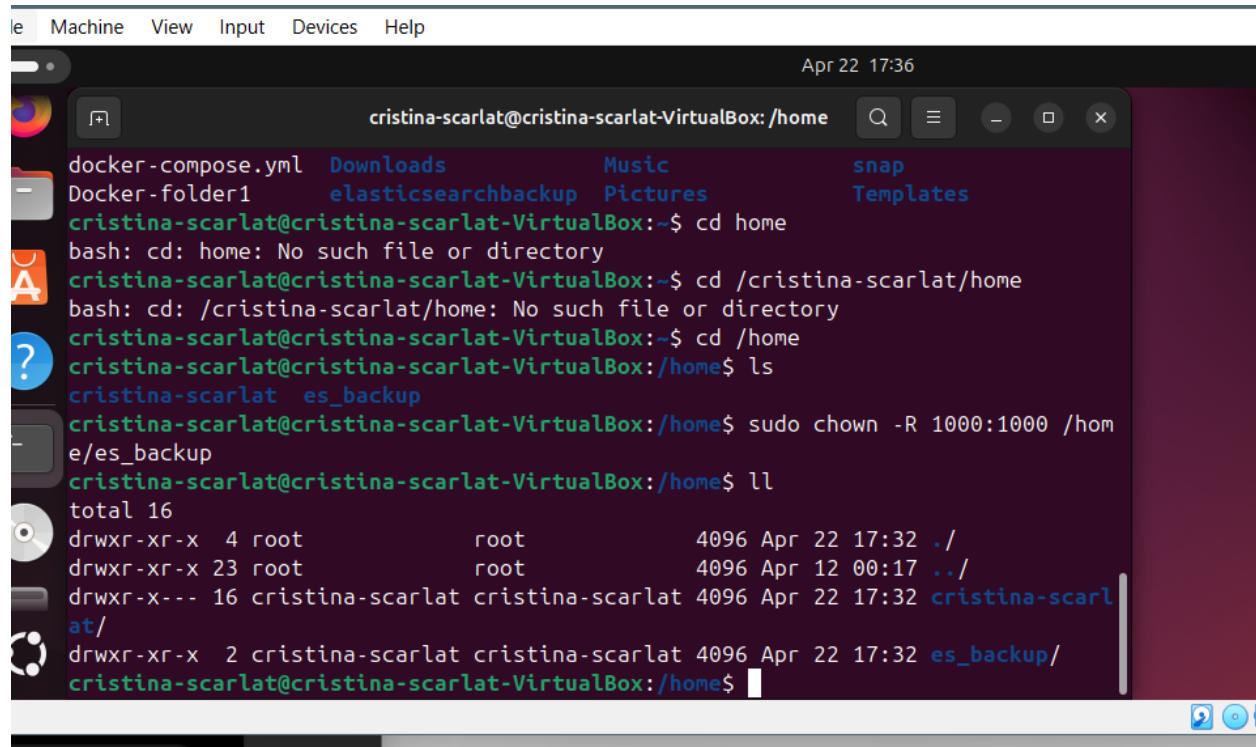
În yml a trebuit să adaug în **path repo** calea fișierului unde vreau să se realizeze snapshotul, altfel realizarea acestuia ar fi fost doar temporară, backup-ul stergându-se la primul restart al mașinii.

În plus, un aspect foarte important este cel legat de **mount**, mai exact, modificarea adusă la linia de volum. Aceasta funcție de mounting se referă la lipirea unui fișier/director/locații de foloderul nostru principal(host), astfel montarea face sistemele de fișiere, fișierele, directoarele, dispozitivele și fișierele speciale disponibile pentru utilizare într-o locație anume. Este singura

modalitate prin care un sistem de fișiere devine accesibile. Ex. Un USB există fizic, dar până când nu este inserat în device, acesta este nefolositor.

Așadar, putem spune:

- Montarea este esențială pentru accesul la date pe orice sistem Unix/Linux.
- Fără montare, fișierele există fizic, dar nu pot fi accesate.
- În Docker, montările sunt vitale pentru păstrarea datelor în afara containerelor, mai ales când vorbim despre backup-uri sau fișiere persistente.



```
cristina-scarlat@cristina-scarlat-VirtualBox:~/home$ cd home
bash: cd: home: No such file or directory
cristina-scarlat@cristina-scarlat-VirtualBox:~/home$ cd /cristina-scarlat/home
bash: cd: /cristina-scarlat/home: No such file or directory
cristina-scarlat@cristina-scarlat-VirtualBox:~/home$ cd /home
cristina-scarlat@cristina-scarlat-VirtualBox:/home$ ls
cristina-scarlat es_backup
cristina-scarlat@cristina-scarlat-VirtualBox:/home$ sudo chown -R 1000:1000 /home/es_backup
cristina-scarlat@cristina-scarlat-VirtualBox:/home$ ll
total 16
drwxr-xr-x 4 root          root          4096 Apr 22 17:32 .
drwxr-xr-x 23 root          root          4096 Apr 12 00:17 ..
drwxr-x--- 16 cristina-scarlat cristina-scarlat 4096 Apr 22 17:32 cristina-scarlat/
drwxr-xr-x  2 cristina-scarlat cristina-scarlat 4096 Apr 22 17:32 es_backup/
cristina-scarlat@cristina-scarlat-VirtualBox:/home$
```

version: '3.8'

services:

elasticsearch:

image: docker.elastic.co/elasticsearch/elasticsearch:8.12.2

container\_name: elasticsearch2

environment:

- discovery.type=single-node
- path.repo=/usr/share/elasticsearch/backup
- xpack.security.enabled=false

```
- bootstrap.memory_lock=true
- ES_JAVA_OPTS=-Xms512m -Xmx512m
- xpack.security.transport.ssl.enabled=false

ports:
- "9200:9200"
- "9300:9300"

volumes:
- esdata:/usr/share/elasticsearch/data
- /home/es_backup:/usr/share/elasticsearch/backup

networks:
- elasticnet

ulimits:
memlock:
soft: -1
hard: -1

kibana:
image: docker.elastic.co/kibana/kibana:8.12.2
container_name: kibana
environment:
- ELASTICSEARCH_HOSTS=http://elasticsearch2:9200
- xpack.security.enabled=false
ports:
- "5601:5601"
networks:
- elasticnet
depends_on:
- elasticsearch

volumes:
esdata:
external: true
name: cristina-scarlat_esdata

networks:
elasticnet:
driver: bridge
```

---

Apoi am creat un repository pentru a putea realiza snapshot-ul pe baza lui.

```
PUT _snapshot/backup_es
{
  "type": "fs",
  "settings": {
    "location": "backup",
    "compress": true
  }
}
```

După am creat o politică pe baza acestui repository care să creeze backup automat odata în fiecare marți de la ora 8.

**weekly-backup-f** 

[Summary](#) [History](#)

Snapshots taken	Failures	Deleted	Deletion failures
<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>

## General

Version	Last modified
<b>1</b>	<b>Apr 22, 2025 7:09 PM GMT+3</b>
Snapshot name	Repository
<a href="#">snapshot-f</a>	<a href="#">backup_es</a>
Schedule	Next snapshot
<b>0 0 8 ? * 3</b>	<b>Apr 29, 2025 11:00 AM</b>
	<b>GMT+3</b>
Data streams and indices	Ignore unavailable indices

[Close](#) [Manage policy](#)

Am dat run și s-a creat primul nostru snapshot.

The screenshot shows the Elasticsearch Stack Management interface. The left sidebar has sections for Management, Lifecycle Policies, and Snapshot and Restore, with Snapshot and Restore selected. The main header is "Snapshot and Restore". The page title is "Snapshot and Restore". A search bar at the top right says "Find apps, content, and more." Below the title, a sub-header says "Use repositories to store and recover backups of your Elasticsearch indices and clusters." There are tabs for "Snapshots", "Repositories", "Policies", and "Restore Status", with "Snapshots" selected. A search bar and a "Repository" dropdown are above the table. The table lists two snapshots: "snapshot-f-nsc5125brekzmmjt7-xiyg" and "snapshot\_test", both in the "backup\_es" repository. The table columns are: Snapshot, Repository, Indices, Shards, Failed shards, Date created, Duration, and Actions. At the bottom, there's a "Rows per page" dropdown set to 20, and a navigation bar with a left arrow, page number 1, and a right arrow.

Snapshot	Repository	Indices	Shards	Failed shards	Date created	Duration	Actions
<a href="#">snapshot-f-nsc5125brekzmmjt7-xiyg</a>	<a href="#">backup_es</a>	24	33	0	Apr 22, 2025 7:09 PM GMT+3	3s	<a href="#"></a> <a href="#"></a>
<a href="#">snapshot_test</a>	<a href="#">backup_es</a>	24	33	0	Apr 22, 2025 7:01 PM GMT+3	5s	<a href="#"></a> <a href="#"></a>

Aici putem observa faptul că backupul a avut loc cu succes.

Index		Actions	
Category	Sub-Category	Value	Duration
000/index_stx	00 Snapshots - Snapshot and ...	+	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup/backup/indices	
		WARN[0000] /home/cristina-scarlat/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion	
		✓ Network cristina-scarlat_elasticnet Created	0.25
		✓ Container elasticsearch2 Started	1.25
		✓ Container kibana Started	1.65
		cristina-scarlat@cristina-scarlat-VirtualBox: \$ ls	
		ls: /z: command not found, but can be installed with:	
		sudo apt install mtools	
		cristina-scarlat@cristina-scarlat-VirtualBox: \$ ls	
		Docker Desktop Docker-Folder1 Downloads elasticseachBackup elasticsearch.yml Pictures snap Videos	
BS		docker-compose.yml Documents Music Public Templates	
		cristina-scarlat@cristina-scarlat-VirtualBox: \$ cd /home	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home\$ ls	
		. .bashrc .bashrc.bak	
ement		cristina-scarlat@cristina-scarlat-VirtualBox: /home\$ cd es_backup	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup\$ ls	
		backup	
e Policies		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup\$ cd backup	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup/backup\$ ls	
		index-1 indices meta-s_31VL3s045cabqhwJsbWBew.dat snap-s_31VL3s045cabqhwJsbWBew.dat	
		index_latest.meta-31VL3s045cabqhwJsbWBew.dat snap-31VL3s045cabqhwJsbWBew.dat	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup/backup\$ cd indices	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup/backup/indices\$ ls	
		.0L0F72H7y0tG_Zpn2Ca DxtTOOCtYutjwzyZ0y0wv 1rDART0TmEnbymh0215wv sd01h-X0nG0taFe7YuUcsA wjciUuaKv026s4rc8ED6Pwg	
		/K9MfC7oN0Hari1UsTaXb0 FPRw4pGL75SS0vdBmZPGdA luADW-MFQsi1F5i1xL2v SWFRTd0uX06x6lnfepht0 lJ_WjAc0y-n-e23445sIlt10	
		er0P12CSRp-0YEE0nqg fHeZ2Lc1856vxxstLAtPbw -l_X0P0FS9p0nkw-sHhC0 _sgAVON-R5qID-ChMACnqA	
		oJ372HP -SLM12r-0PTonr Fvgv3yh2Ty0loLjVnPrg m66Gz0QY0Q8C10huM4GcpRA TMCjvTAHSVlyIHKhulgRq xJLwyd0ITGH7d9MPL16N10	
		bgbJlJxqP-EfhokHIIHath ha7n0J6L72VyzzzEnhpuq of_b-DPXTN-CQ63H7t7i_v VLJda971QGz2ngXkFtvA xjnhis1jS62dNa0RacBoEA	
		LvKvHmC8LUs0gq23sLkv -zebecM8Rllqzbjgxbozu po2zL9qfjPfRWhch78Mlk8w	
sights	②	dv_K4N4QK0C4EgyjZQ2X KAT7-ykR_LSoqgkEzJdo QRP0355GSTM50CMgLK1ca WdnInluulQXKhryfJuju6EQ	
		cristina-scarlat@cristina-scarlat-VirtualBox: /home/es_backup/backup/indices\$	
		Rows per page: 20 ▾	

## Restore

Restore = procesul prin care reîncarci datele dintr-un snapshot într-un cluster Elasticsearch.

Snapshot-ul poate contine unul sau mai mulți indici (index-uri), iar restore-ul îți permite să:

- readuci indici pierduți sau șterși accidental
  - clonezi date între clustere
  - reversi la o versiune anterioară a datelor

Însă pentru a putea realiza operația de restore este necesar să ștergem din cluster indeșii existenței sau să-i restaurăm cu alt nume.

Pentru a testa acest proces, am ales să sterg index cursuri și să-l restaurez cu aceeași denumire.

//verificam daca avem indecsii in snapshot

GET /snapshot/backup es/snapshot-f-nsc5125brekzmmjt7-xlyg/

Apr 23 09:09

Console - Dev Tools - Elasticsearch - localhost:9200/ http://localhost:5601/app/dev\_tools#/console 130% Find apps, content, and more. elastic Dev Tools Console

History Settings Variables Help 200 - OK 59 ms

```
6 * }
7 *
8 GET _cat/indices
9
10 PUT _snapshot/es_backup_repo
11 {
12   "type": "fs"
13   "settings": {
14     "location": "backup",
15     "compress": true
16   }
17 }
18
19 PUT _snapshot/es_backup_repo/snapshot_test
20
21 PUT _snapshot/backup_es
22 {
23   "type": "fs"
24   "settings": {
25     "location": "backup",
26     "compress": true
27   }
28 }
29
30 //verificam daca avem indecsit in snapshot
31 GET _snapshot/backup_es/snapshot-f-nsc5125brekzmmjt7-xlyg/
32
33
34
```

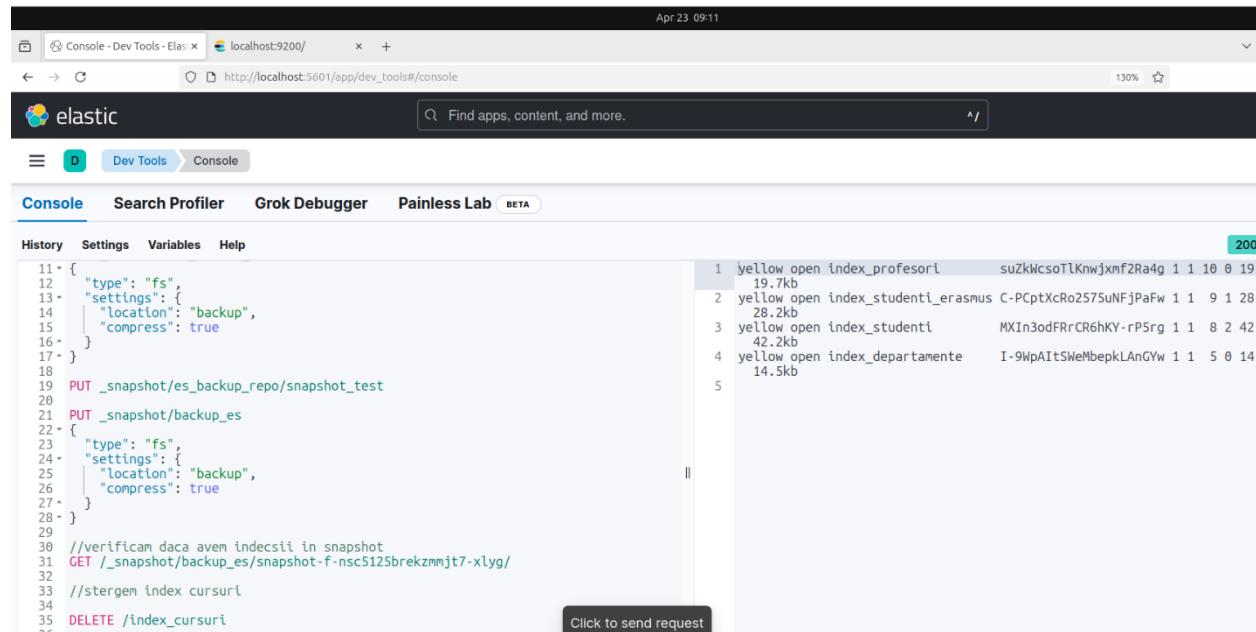
1 { "snapshots": [
2 {
3 "snapshot": "snapshot-f-nsc5125brekzmmjt7-xlyg",
4 "uid": "3IVL3s04ScabqJwlsbWBew",
5 "repository": "backup\_es",
6 "version\_id": 8500010,
7 "version": "8500010",
8 "indices": [
9 {
10 "tasks": [
11 ".kibana-event-log-ds",
12 ".internal.alerts-observability.metrics.alerts-default-000001",
13 ".apm-agent-configuration",
14 ".kibana ingest\_8.12.1\_001",
15 ".kibana\_alerting\_cases\_8.12.1\_001",
16 ".kibana\_analytics\_8.12.1\_001",
17 ".index\_departamento",
18 ".index\_profesor",
19 ".ds\_slm-history-5-2025.04.22-000001",
20 ".internal.alerts-stack.alerts-default-000001",
21 ".ds\_logs-deprecation.elasticsearch-default-2025.04.22-000001",
22 ".slo-observability.sli-v3",
23 ".internal.alerts-security.alerts-default-000001",
24 ".index\_cutsuri",
25 ".internal.alerts-observability.threshold.alerts-default-000001",
26 ".index\_studenti\_erasmus",
27 ".index\_studenti",
28 ".internal.alerts-observability.uptime.alerts-default-000001",
29 ".slo-observability.summary-v3.temp"
30 ]
31 }
32 ]
33 }
34 ]}

//stergem index cursuri

**DELETE /index cursuri**

//verificam daca mai exista

GET \_cat/indices



The screenshot shows the Elasticsearch Dev Tools Console interface. The URL is http://localhost:5601/app/dev\_tools#/console. The results of a GET \_cat/indices request are displayed, showing five indices: index\_profesori, index\_studenti\_erasmus, index\_studenti, index\_departamente, and index\_cursuri. Each entry includes details like status, type, size, and shard counts.

Index	Type	Status	Size	Shards	Replicas
index_profesori	fs	yellow	19.7kb	1	10
index_studenti_erasmus	fs	yellow	28.2kb	1	9
index_studenti	fs	yellow	42.2kb	1	8
index_departamente	fs	yellow	14.5kb	1	5
index_cursuri	fs	yellow	0	0	0

//Resaturam acest index

POST /\_snapshot/backup\_es/snapshot-f-nsc5125brekzmmjt7-xlyg/\_restore

```
{  
  "indices": "index_cursuri"  
}
```

//verificam daca s-a restaurat

GET \_cat/indices

The screenshot shows the Elasticsearch Dev Tools Console interface. In the left panel, there is a code editor with a series of Elasticsearch API requests. The right panel displays the results of these requests, including index statistics and document counts. A tooltip 'Click to send request' points to the execute button at the bottom of the code editor.

```

21 PUT _snapshot/backup_es
22 {
23   "type": "fs",
24   "settings": {
25     "location": "backup",
26     "compress": true
27   }
28 }
29
30 //verificam daca avem indecsil in snapshot
31 GET _snapshot/backup_es/snapshot-f-nsc5125brekzmnjt7-xlyg/
32
33 //stergem index cursuri
34
35 DELETE /index_cursuri
36
37 //verificam daca mal exista
38 GET _cat/indices
39
40 //Resaturam acest index
41 POST _snapshot/backup_es/snapshot-f-nsc5125brekzmnjt7-xlyg/_restore
42 {
43   "indices": "index_cursuri"
44 }
45
46 //verificam daca s-a restaurat
47
48 GET _cat/indices
49

```

Index	Type	Name	Primary Shards	Replica Shards	Docs	Deleted	Size	Primary Size	Replica Size
1	yellow	open_index_profesori	1	1	10	0	19.7kb	19.7kb	19.7kb
2	yellow	open_index_studenti_erasmus	1	1	9	1	28.2kb	28.2kb	28.2kb
3	yellow	open_index_studenti	1	1	8	2	42.2kb	42.2kb	42.2kb
4	yellow	open_index_departamente	1	1	5	0	14.5kb	14.5kb	14.5kb
5	yellow	open_index_cursuri	1	1	13	0	25kb	25kb	25kb
6									

Am verificat și în bash dacă acesta s-a resaturat cu succes.

```

$ docker COMMAND --help' for more information on a command.

For more help on how to use Docker, head to https://docs.docker.com/go/guides/
cristina-scarlat@cristina-scarlat-VirtualBox:~$ sudo docker compose up -d
[REDACTED] /home/cristina-scarlat/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid warnings
| Running 2/2
Container elasticsearch2 Started
Container kibana Started
cristina-scarlat@cristina-scarlat-VirtualBox:~$ curl -X GET "localhost:9200/_cat/indices?v"
{
  "status": "index",
  "uuid": "suZkWcsoTlKnwJxmF2Ra4g",
  "pri": 1,
  "rep": 1,
  "docs": {
    "count": 10,
    "deleted": 0
  },
  "store": {
    "size": "19.7kb",
    "pri_size": "19.7kb",
    "rep_size": "19.7kb"
  },
  "pri_size": "19.7kb",
  "rep_size": "19.7kb"
}
{
  "status": "index",
  "uuid": "C-PCptXcRo2575uNFjPaFw",
  "pri": 1,
  "rep": 1,
  "docs": {
    "count": 9,
    "deleted": 1
  },
  "store": {
    "size": "28.2kb",
    "pri_size": "28.2kb",
    "rep_size": "28.2kb"
  },
  "pri_size": "28.2kb",
  "rep_size": "28.2kb"
}
{
  "status": "index",
  "uuid": "MXIn3odFRrCR6hKY-rP5rg",
  "pri": 1,
  "rep": 1,
  "docs": {
    "count": 8,
    "deleted": 2
  },
  "store": {
    "size": "42.2kb",
    "pri_size": "42.2kb",
    "rep_size": "42.2kb"
  },
  "pri_size": "42.2kb",
  "rep_size": "42.2kb"
}
{
  "status": "index",
  "uuid": "I-9WpAITSWemBepkLAnGYw",
  "pri": 1,
  "rep": 1,
  "docs": {
    "count": 5,
    "deleted": 0
  },
  "store": {
    "size": "14.5kb",
    "pri_size": "14.5kb",
    "rep_size": "14.5kb"
  },
  "pri_size": "14.5kb",
  "rep_size": "14.5kb"
}
{
  "status": "index",
  "uuid": "cfX8EfL0TRiKlwY8og-HDQ",
  "pri": 1,
  "rep": 1,
  "docs": {
    "count": 13,
    "deleted": 0
  },
  "store": {
    "size": "25kb",
    "pri_size": "25kb",
    "rep_size": "25kb"
  },
  "pri_size": "25kb",
  "rep_size": "25kb"
}
cristina-scarlat@cristina-scarlat-VirtualBox:~$ 

```

//verificam daca si-a pastrat continutul

GET /index\_cursuri/\_search

Console - Dev Tools - Elasticsearch - localhost:5601

Find apps, content, and more.

elastic

Dev Tools Console

Console Search Profiler Grok Debugger Painless Lab BETA

History Settings Variables Help

200 - OK 90 ms

```
24- "settings": { 25   "location": "backup", 26   "compress": true 27- } 28- } 29- 30 //verificam daca avem indecsi in snapshot 31 GET /_snapshot/backup_es/snapshot-f-nsc5125brekzmmjt7-xlyg/ 32- 33 //stergem index cursuri 34- 35 DELETE /index_cursuri 36- 37 //verificam daca mai exista 38 GET _cat/indices 39- 40 //Resaturam acest index 41 POST /_snapshot/backup_es/snapshot-f-nsc5125brekzmmjt7-xlyg/_restore 42- { 43   "indices": "index_cursuri" 44- } 45- 46 //verificam daca s-a restaurat 47- 48 GET _cat/indices 49- 50- 51 //verificam daca si-a pastrat continutul 52 GET /index_cursuri/_search
```

Click to send request

14°C 9:14 AM Right Ctrl

## Sectiunea 5- Reindexare

Reindexarea este procesul prin care copiez documente dintr-un index sursă într-un alt index destinație. Poți folosi ocazia să modifici datele, să adaugi câmpuri noi sau să folosești un alt mapping.

Când folosim reindexare?

- Pentru a modifica mapping-ul, mai exact schimbăm type-ul unei variabile;
- Pentru a adăuga/modifica câmpuri
- Pentru curățare, arhivare sau transformări
- Pentru a crea un alias swap
- 

Observații: Indexul vechi nu este șters, el trebuie șters manual.

Dacă nu creăm o structură pentru indexul destinație în prealabil, acesta va primi câmpurile modificate de tipul text, cu tipul keyword.

Sintaxă generală:

---

### **POST \_reindex**

```
{  
  "source": {  
    "index": "index_vechi"  
  },  
  "dest": {  
    "index": "index_nou"  
  }  
}
```

Observații: Index\_nou va fi creat cu mapping alocat dinamic, ceea ce înseamnă că este posibil ca anumite câmpuri să primească tipul greșite de dată.

---

### **Reindexare cu script (pentru modificarea documentelor)**

---

POST \_reindex

```
{  
  "source": {  
    "index": "index_vechi"  
  },  
  "dest": {  
    "index": "index_nou"  
  },  
  "script": {  
    "lang": "painless",  
    "source": """"  
      ctx._source.status = 'active';  
      ctx._source.age = ctx._source.age != null ? ctx._source.age : 0;  
    """"  
  }  
}
```

Explicații: 1. `ctx._source.status = 'active';`

Ce face?

Adaugă (sau înlocuiește) câmpul status din fiecare document și îi dă valoarea "active".

Dacă documentul nu avea acel câmp, îl creează. Dacă îl avea, îl suprascrie.

2. `ctx._source.age = ctx._source.age != null ? ctx._source.age : 0;`

Ce face?

Verifică dacă documentul are deja un câmp age și dacă nu e null.

Dacă are o valoare (adică e diferit de null), o păstrează.

Dacă NU are valoare (e null sau lipsește), setează age la 0.

---

### **Reindexare cu filtrare (ex: doar anumiți useri)**

---

`POST _reindex`

```
{  
  "source": {  
    "index": "index_vechi",  
    "query": {  
      "term": {  
        "status": "inactive"  
      }  
    }  
  },  
  "dest": {  
    "index": "index_nou"
```

```
}
```

---

```
}
```

## **Alias swap**

---

```
POST _aliases
{
  "actions": [
    { "remove": { "index": "index_vechi", "alias": "my_index" }},
    { "add": { "index": "index_nou", "alias": "my_index" }}
  ]
}
```

---

## **Secțiunea 6-Aplicații**

### **1. Modificarea indexului: index\_cursuri astfel încât:**

- “Denumire” să nu mai fie “text”, ci “keyword”, pentru a putea face sortare și filtrare exactă,
- Să se adauge un câmp nou “An\_studiu” pentru fiecare curs

Rezolvare: Este nevoie sa folosim reindexare, deoarece in Elastic nu putem modifica direct structura unui index.

**-Cream un nou index care sa contine modificarile cerute, insa nepopulat cu date**

```
PUT /index_cursuri2
{
  "mappings": {
    "properties": {
```

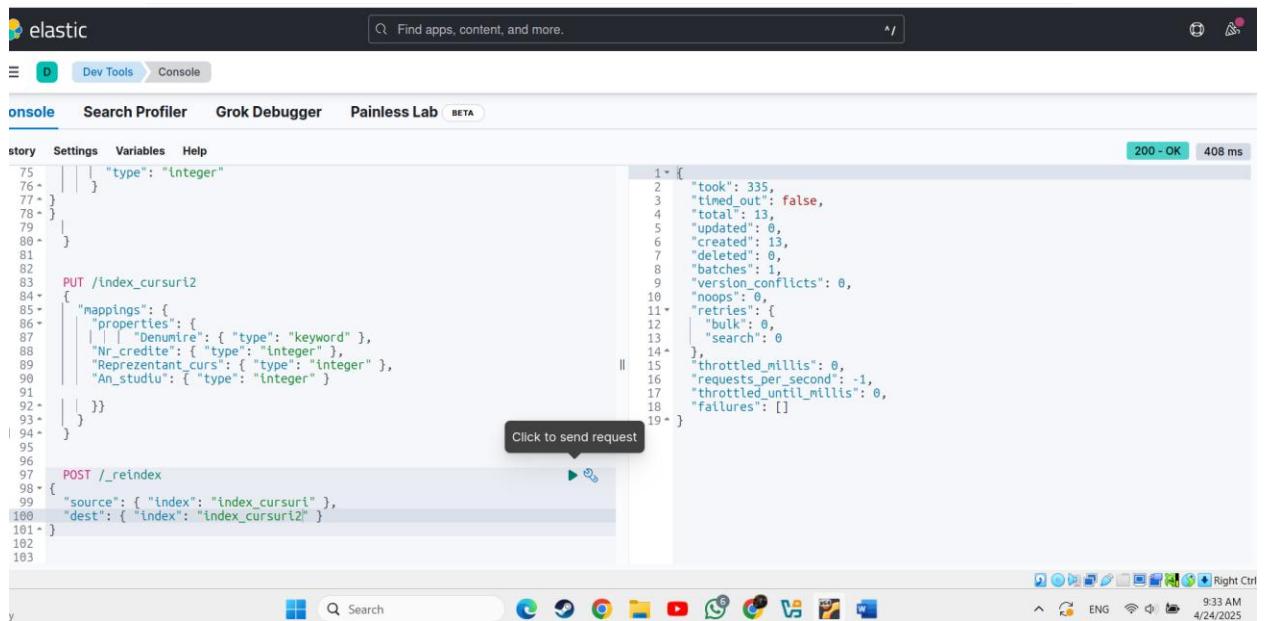
```
        "Denumire": { "type": "keyword" },
        "Nr_credite": { "type": "integer" },
        "Reprezentant_curs": { "type": "integer" },
        "An_studiu": { "type": "integer" }

    }
}
```

## **-reindexam**

POST / reindex

```
{  
  "source": { "index": "index_cursuri" },  
  "dest": { "index": "index_cursuri2" }  
}
```



-adaugam campul An studiu

```
POST /index_cursuri2/_update_by_query  
{  
  "script":{  
    "source" : "ctx._source.AN_studiu = 1"  
    "lang": "painless"  
  },  
}
```

```

    "query": {
      "match_all": {}
    }
  }

```

The screenshot shows the Elasticsearch Dev Tools interface with the 'Console' tab selected. A search query is entered in the text area:

```

99  "source": { "index": "index_cursuri" },
100 "dest": { "index": "index_cursuri2" }
101 }
102
103
104 POST /index_cursuri_v2/_update_by_query
105 {
106   "script": {
107     "source": "ctx._source.An_studiu = 1",
108     "lang": "painless"
109   },
110   "query": {
111     "match_all": {}
112   }
113 }
114
115 POST /index_cursuri2/_update_by_query
116 {
117   "script": {
118     "source": "ctx._source.An_studiu = 1",
119     "lang": "painless"
120   },
121   "query": {
122     "match_all": {}
123   }
124 }
125
126 GET /index_cursuri2/_search

```

The results pane displays three documents from the 'index\_cursuri2' index:

```

17 {
18   "_index": "index_cursuri2",
19   "_id": "UGS-PSYBÜoIVGB90YE6V",
20   "_score": 1,
21   "_source": {
22     "Nr_credite": 6,
23     "Denumire": "Algebra Liniara",
24     "AN_studiu": 1,
25     "Reprezentat_curs": 3001
26   }
27 }
28 {
29   "_index": "index_cursuri2",
30   "_id": "UMS-PSYBÜoIVGB90k07t",
31   "_score": 1,
32   "_source": {
33     "Nr_credite": 6,
34     "Denumire": "Analiza matematica",
35     "AN_studiu": 1,
36     "Reprezentat_curs": 3002
37   }
38 }
39 {
40   "_index": "index_cursuri2",
41   "_id": "UmS-PSYBÜoIVGB90ME4A",
42   "_score": 1,
43   "_source": {
44     "Nr_credite": 4,
45     "Denumire": "Microeconomia cantitativa"
46   }
47 }

```

At the top right, the status bar shows '200 - OK' and '41'.

## -comparam structura indexului index\_cursuri2 cu index\_cursuri

The screenshot shows the Elasticsearch Settings API with the 'JSON' tab selected. The 'index\_cursuri' index mapping is displayed:

```

{
  "index_cursuri": {
    "aliases": {},
    "mappings": {
      "properties": {
        "Denumire": {
          "type": "text"
        },
        "Nr_credite": {
          "type": "integer"
        },
        "Reprezentant_curs": {
          "type": "integer"
        },
        "Reprezentat_curs": {
          "type": "long"
        }
      }
    },
    "settings": {
      "index": {
        "routing": {
          "allocation": {
            "include": {
              "_tier_preference": "data_content"
            },
            "number_of_shards": "1",
            "provided_name": "index_cursuri",
            "creation_date": "1744826058492",
            "number_of_replicas": "1",
            "uuid": "cfx8EfL0TRiKlWygog-HDQ"
          }
        }
      }
    }
  }
}

```

```

Save Copy Collapse All Expand All | Filter JSON
▼ index_cursuri2:
  ▼ aliases: {}
  ▼ mappings:
    ▼ properties:
      ▼ An_studiu:
        type: "integer"
      ▼ Denumire:
        type: "keyword"
      ▼ Nr_credite:
        type: "integer"
      ▼ Reprezentant_curs:
        type: "integer"
      ▼ Reprezentat_curs:
        type: "long"
  ▼ settings:
    ▼ index:
      ▼ routing:
        ▼ allocation:
          ▼ include:
            _tier_preference: "data_content"
        number_of_shards: "1"
        provided_name: "index_cursuri2"
        creation_date: "1745476244435"
        number_of_replicas: "1"
        uuid: "y60iKsLUTyeD9enz9MVnUw"
    ▼ version:
      created: "8500010"

```

## -verificam continutul

```

← → ⌂ http://localhost:9200/index_cursuri2/_search
JSON Raw Data Headers
Save Copy Collapse All Expand All | Filter JSON
Reprezentat_curs: 3001
▼ 5:
  _index: "index_cursuri2"
  _id: "VWS_P5YBu0IVGB904E5r"
  _score: 1.0 [S:1]
  ▼ _source:
    Nr_credite: 6
    Denumire: "Statistica"
    AN_studiu: 1
    Reprezentat_curs: 3005
▼ 6:
  _index: "index_cursuri2"
  _id: "VmTAP5YBu0IVGB90KE7i"
  _score: 1.0 [S:1]
  ▼ _source:
    Nr_credite: 3
    Denumire: "POO"
    AN_studiu: 1
    Reprezentat_curs: 3006
▼ 7:
  _index: "index_cursuri2"
  _id: "V2TAP5YBu0IVGB90ZE47"
  _score: 1.0 [S:1]
  ▼ _source:
    Nr_credite: 2
    Denumire: "Cercetari Operationale"
    AN_studiu: 1
    Reprezentat_curs: 3007

```

- sortare după “Denumire” în noul index.

Deoarece am schimbat tipul din text in keyboard, sortarea a fost posibila.

GET index\_cursuri2/\_search

```
{  
  "sort": [  
    {  
      "Denumire": {  
        "order": "asc"  
      }  
    }  
  ]  
}
```

The screenshot shows the Elasticsearch Dev Tools Console interface. The top navigation bar includes 'elastic' (logo), 'Find apps, content, and more.', and a search bar. Below the bar are tabs: 'Dev Tools' (selected), 'Console', 'Search Profiler', 'Grok Debugger', and 'Painless Lab (BETA)'. The main area has tabs: 'History', 'Settings', 'Variables', and 'Help'. The code editor contains the following Elasticsearch search query:

```
182  
183  
184  
185  
186 POST /index_cursuri2/_update_by_query  
187 {  
188   "script": {  
189     "source" : "ctx._source.AN_studiu = 1",  
190     "lang": "painless"  
191   },  
192   "query": {  
193     "match_all": {}  
194   }  
195 }  
196 GET /index_cursuri2/_search  
197  
198 [ 120 |  
199 121 {  
200 122 "sort": [  
201 123 {  
202 124 "Denumire": {  
203 125   "order": "asc"  
204 126 }  
205 127 }  
206 128 ]  
207 129 ]  
208 130 ]
```

The results pane shows the response from the server:

```
200 - OK 51 ms  
{  
  "hits": {  
    "total": {  
      "value": 13,  
      "relation": "eq"  
    },  
    "max_score": null,  
    "hits": [  
      {  
        "index": "index_cursuri2",  
        "id": "WnTBPSYBu0IVGB90Mk4A",  
        "score": null,  
        "source": {  
          "Nr_credite": 5,  
          "Denumire": "Administrarea Bazelor de Date",  
          "AN_studiu": 1,  
          "Reprezentat_curs": 3010  
        },  
        "sort": [  
          "Administrarea Bazelor de Date"  
        ]  
      },  
      {  
        "index": "index_cursuri2",  
        "id": "UGS-P5BYBu0IVGB90YE6V",  
        "score": null,  
        "source": {  
          "Nr_credite": 6,  
          "Denumire": "Algebra Liniara"  
        }  
      }  
    ]  
  }  
}
```

At the bottom right of the results pane, there are several small icons: magnifying glass, gear, refresh, etc., followed by the text 'Right Ctrl'.

- căutare după "AN\_studiu"

The screenshot shows the Elasticsearch Dev Tools Console interface. On the left, there is a code editor with a series of numbered lines of JSON code. Lines 111-139 show a search query for 'AN\_studiu'. Lines 15-43 show the resulting hits, which are two documents from the 'index\_cursuri2' index. Each hit includes fields like '\_index', '\_id', '\_score', and '\_source'.

```
111 },  
112 "query": {  
113     "match_all": {}  
114 }  
115 }  
116 }  
117  
118 GET /index_cursuri2/_search  
119  
120 GET index_cursuri2/_search  
121 {  
122     "sort": [  
123         {  
124             "Denumire": {  
125                 "order": "asc"  
126             }  
127         }  
128     ]  
129 }  
130  
131 GET index_cursuri2/_search  
132 {  
133     "query": {  
134         "term": {  
135             "AN_studiu": 1  
136         }  
137     }  
138 }  
139
```

```
15     "max_score": 1,  
16     "hits": [  
17         {  
18             "_index": "index_cursuri2",  
19             "_id": "UGS-PSYBu0IVG890YE6V",  
20             "_score": 1,  
21             "_source": {  
22                 "Nr_credite": 6,  
23                 "Denumire": "Algebra Liniara",  
24                 "AN_studiu": 1,  
25                 "Reprezentat_curs": 3001  
26             }  
27         },  
28         {  
29             "_index": "index_cursuri2",  
30             "_id": "UNs-PSYBu0IVG890k07t",  
31             "_score": 1,  
32             "_source": {  
33                 "Nr_credite": 6,  
34                 "Denumire": "Analiza matematica",  
35                 "AN_studiu": 1,  
36                 "Reprezentat_curs": 3002  
37             }  
38         },  
39         {  
40             "_index": "index_cursuri2",  
41             "_id": "Um5-PSYBu0IVG890ME4A",  
42             "_score": 1,  
43             "_source": {  
44         }
```

## Ce am învățat din exercițiul?

-Pentru a sorta e nevoie de un câmp de tip keyword.

-Dacă reindeindexăm într-un index\_nou , care are un câmp nou (fără valori), câmpul nu va primi o valoare default. Mai mult de atât câmpul nu va apărea în structura indexului nou în momentul în care folosim search. Așa că e novoie de o nouă reindeindexare în care să atribuim o valoare default pentru noul câmp.

## 2. 1. Reindexare cu adăugare câmp nou 'Tip\_curs'

Creează un nou index `index\_cursuri\_tip` în care adăugă câmpul 'Tip\_curs' (ex: obligatoriu, optional). Copiază toate datele din 'index\_cursuri' și adaugă acest câmp cu valoare implicită 'obligatoriu'.

-creare index\_cursuri\_tip

```
PUT /index_cursuri_tip  
{  
    "mappings": {  
        "properties": {  
            "Denumire": { "type": "keyword" },  
            "Nr_credite": { "type": "integer" },  
            "Reprezentant_curs": { "type": "integer" },  
            "Tip_curs": { "type": "keyword" }  
        }  
    }  
}
```

```

    }
}
}
```

-reindexare

POST /\_reindex

```
{
  "source": { "index": "index_cursuri" },
  "dest": { "index": "index_cursuri_tip" }
}
```

- adaugam valoarea default 'obligatoriu'

POST /index\_cursuri\_tip/\_update\_by\_query

```
{
  "script": {
    "source": "ctx._source.Tip_curs= 'obligatoriu' ",
    "lang": "painless"
  }
}
```

The screenshot shows the Elasticsearch Dev Tools interface with the following details:

- Header:** File, Search Profiler, Grok Debugger, Painless Lab (BETA)
- Left Panel:**
  - Settings, Variables, Help
  - Scripted Fields (with code snippets for Reprezentant\_curs and Tip\_curs)
  - POST /\_reindex (with source and dest indices specified)
  - get /index\_cursuri\_tip/\_search
  - POST /index\_cursuri\_tip/\_update\_by\_query (with script: ctx.\_source.Tip\_curs= 'obligatoriu' and lang: painless)
  - get /index\_cursuri\_tip/\_search
- Right Panel:**
  - 200 - OK, 32 ms
  - Scripted Fields (with code snippets for Reprezentant\_curs and Tip\_curs)
  - POST /\_reindex (with source and dest indices specified)
  - get /index\_cursuri\_tip/\_search (with results):
    - Index: index\_cursuri\_tip, ID: UGS-P5YBUoIVGB90YE6V, Score: 1, Source: {"Nr\_credite": 6, "Denumire": "Algebra Liniara", "Reprezentat\_curs": 3001, "Tip\_curs": "obligatoriu"}
    - Index: index\_cursuri\_tip, ID: UMS-P5YBUoIVGB90k07t, Score: 1, Source: {"Nr\_credite": 6, "Denumire": "Analiza matematica", "Reprezentat\_curs": 3002, "Tip\_curs": "obligatoriu"}
    - Index: index\_cursuri\_tip, ID: UmS\_P5YBUoIVGB90ME4A, Score: 1, Source: {"Nr\_credite": 4, "Denumire": "Microeconomia cantitativa"}

A tooltip says "Click to send request" pointing to the search bar.

## 2.2 Reindexare cu modificare de mapping pentru 'Nr\_credite'

Creează un nou index 'index\_cursuri\_numeric' în care 'Nr\_credite' este definit ca 'long' în loc de 'integer'. Copiază datele din 'index\_cursuri' și testează sortarea după acest câmp.

PUT /index\_cursuri\_numeric

```
{  
  "mappings": {  
    "properties": {  
      "Denumire": { "type": "keyword" },  
      "Nr_credite": { "type": "long" },  
      "Reprezentant_curs": { "type": "integer" }  
    }  
  }  
}
```

POST /\_reindex

```
{  
  "source": { "index": "index_cursuri" },  
  "dest": { "index": "index_cursuri_numeric" }  
}
```

-sortare

The screenshot shows the Elasticsearch Dev Tools Console interface. The top navigation bar includes 'elastic', 'Find apps, content, and more.', and a search bar. Below the bar, tabs for 'Console', 'Search Profiler', 'Grok Debugger', and 'Painless Lab' are visible, with 'Console' being the active tab. The main area has tabs for 'History', 'Settings', 'Variables', and 'Help'. A code editor on the left contains the following log entries:

```
157  
158  
159 POST /index_cursuri_tip/_update_by_query  
160 {  
161   "script": {  
162     "source": "ctx._source.Tip_curs= 'obligatoriu' ",  
163     "lang": "painless"  
164   }  
165 }  
166  
167  
168 get /index_cursuri_tip/_search  
169  
170   POST /_reindex  
171 {  
172   "source": { "index": "index_cursuri" },  
173   "dest": { "index": "index_cursuri_numeric" }  
174 }  
175  
176 GET index_cursuri_numeric/_search  
177 {  
178   "sort": [  
179     {  
180       "Nr_credite": {  
181         "order": "asc"  
182       }  
183     }  
184   ]  
185 }
```

A tooltip 'Click to send request' points to the 'dest' field in the reindex command. To the right, the results of a search query are displayed in two columns. The first column shows the raw log entries from lines 157 to 185. The second column shows the corresponding Elasticsearch search results:

```
51   {  
52     "_index": "index_cursuri_numeric",  
53     "_id": "XGTBPSYBu0IVGB90yE40",  
54     "score": null,  
55     "_source": {  
56       "Denumire": "Optional Matematica Aplicata ",  
57       "Nr_credite": 0,  
58       "Reprezentant_curs": 3002  
59     },  
60     "sort": [  
61       0  
62     ]  
63   },  
64   {  
65     "_index": "index_cursuri_numeric",  
66     "_id": "V2TAPSYBu0IVGB90ZE47",  
67     "score": null,  
68     "_source": {  
69       "Denumire": "Cercetari Operationale",  
70       "Nr_credite": 2,  
71       "Reprezentant_curs": 3007  
72     },  
73     "sort": [  
74       2  
75     ]  
76   },  
77   {  
78     "_index": "index_cursuri_numeric",  
79     "_id": "WGTAPSYBu0IVGB90M040",  
80     "score": null,  
81   }
```

The results show three documents with their respective scores and source details. The 'Nr\_credite' field is explicitly mentioned in the '\_source' field of each document.

```

GET index_cursuri_numeric/_search
{
  "sort": [
    {
      "Nr_credite": {
        "order": "asc"
      }
    }
  ]
}

```

### 2.3. Reindexare cu renumire de câmp (de la 'Denumire' la 'Titlu')

Creează un nou index `index\_cursuri\_rename` în care 'Denumire' este redenumit 'Titlu'. Folosește un script în `reindex` pentru a copia datele cu numele câmpului schimbat.

```

PUT /index_cursuri_rename
{
  "mappings": {
    "properties": {
      "Titlu": { "type": "keyword" },
      "Nr_credite": { "type": "integer" },
      "Reprezentant_curs": { "type": "integer" }
    }
  }
}

```

```

POST /_reindex
{
  "source": {
    "index": "index_cursuri"
  },
  "dest": {
    "index": "index_cursuri_rename"
  },
  "script": {
    "lang": "painless",

```

```

    "source": """""
    ctx._source.Titlu = ctx._source.remove("Denumire");
    """
}

{
}

```

The screenshot shows the Elasticsearch Dev Tools Console interface. On the left, there is a code editor with a scrollable list of log entries. The entries include:

- Line 171: A mapping update for index `index\_cursuri`.
- Line 172: A search request for `index\_cursuri/\_search`.
- Line 173: A get request for `\_cat/indices`.
- Line 174: A post request for `/\_reindex`.
- Line 175: A mapping update for index `index\_cursuri\_rename`.
- Line 176: A search request for `index\_cursuri\_rename/\_search`.
- Line 177: A get request for `\_cat/indices`.
- Line 178: A post request for `/\_reindex`.
- Line 179: A mapping update for index `index\_cursuri`.
- Line 180: A search request for `index\_cursuri/\_search`.
- Line 181: A post request for `/\_reindex`.
- Line 182: A mapping update for index `index\_cursuri\_rename`.
- Line 183: A search request for `index\_cursuri\_rename/\_search`.
- Line 184: A get request for `\_cat/indices`.
- Line 185: A post request for `/\_reindex`.
- Line 186: A mapping update for index `index\_cursuri`.
- Line 187: A search request for `index\_cursuri/\_search`.
- Line 188: A get request for `\_cat/indices`.
- Line 189: A post request for `/\_reindex`.
- Line 190: A mapping update for index `index\_cursuri\_rename`.
- Line 191: A search request for `index\_cursuri\_rename/\_search`.
- Line 192: A get request for `\_cat/indices`.
- Line 193: A post request for `/\_reindex`.
- Line 194: A mapping update for index `index\_cursuri`.
- Line 195: A search request for `index\_cursuri/\_search`.
- Line 196: A get request for `\_cat/indices`.
- Line 197: A post request for `/\_reindex`.
- Line 198: A mapping update for index `index\_cursuri\_rename`.
- Line 199: A search request for `index\_cursuri\_rename/\_search`.

On the right side of the interface, the results of the last search request (line 199) are displayed. The response status is 200 OK and it took 92 ms. The results show a single hit with a total score of 1, where the value is 13 and the relation is eq. The hit details include the index, \_id, \_score, source, and specific fields like Nr\_credite, Titlu, and Reprezentant\_curs.

2.//

#### 4. Reindexare cu filtrare - doar cursuri cu peste 3 credite

Creează un index nou `index\_cursuri\_filtrat` și copiază doar documentele din `index\_cursuri` cu `Nr\_credite > 3`.

PUT /index\_cursuri\_filtrat

```

{
  "mappings": {
    "properties": {
      "Denumire": { "type": "keyword" },
      "Nr_credite": { "type": "integer" },
      "Reprezentant_curs": { "type": "integer" }
    }
  }
}

```

```

POST _reindex
{
  "source": {
    "index": "index_cursuri",
    "query": {
      "range": {
        "Nr_credite": {
          "gt": 3
        }
      }
    },
    "dest": {
      "index": "index_cursuri_filtrat"
    }
  }
}

```

**2.5. Reindexare din `index\_profesori` cu câmp nou `Grad\_didactic`**  
 Creează un nou index `index\_profesori\_grad` care conține un câmp nou `Grad\_didactic`, setat implicit la 'lector'. Fă reindex din `index\_profesori` și folosește un script pentru a adăuga acest câmp.

**.5. Reindexare din `index\_profesori` cu câmp nou `Grad\_didactic`**  
 Creează un nou index `index\_profesori\_grad` care conține un câmp nou `Grad\_didactic`, setat implicit la 'lector'. Fă reindex din `index\_profesori` și folosește un script pentru a adăuga acest câmp.

```

PUT /index_profesori_grad
{
  "mappings": {
    "properties": {
      "Nume": {
        "type": "text"
      },
      "Birou": {
        "type": "integer"
      }
    }
  }
}

```

```
        },
        "Id_profesor": {
            "type": "integer"
        },
        "Grad_didactic": {
            "type": "keyword"
        }
    }
}
```

```
POST _reindex
{
  "source": {"index": "index_profesori"},
  "dest": {"index": "index_profesori_grad"}
}
```

```
POST index_profesori_grad/_update_by_query
{
  "script": {
    "source": "ctx._source.Grad_didactic = 'lector' ",
    "lang": "painless"
  }
}
```

Dev Tools Console

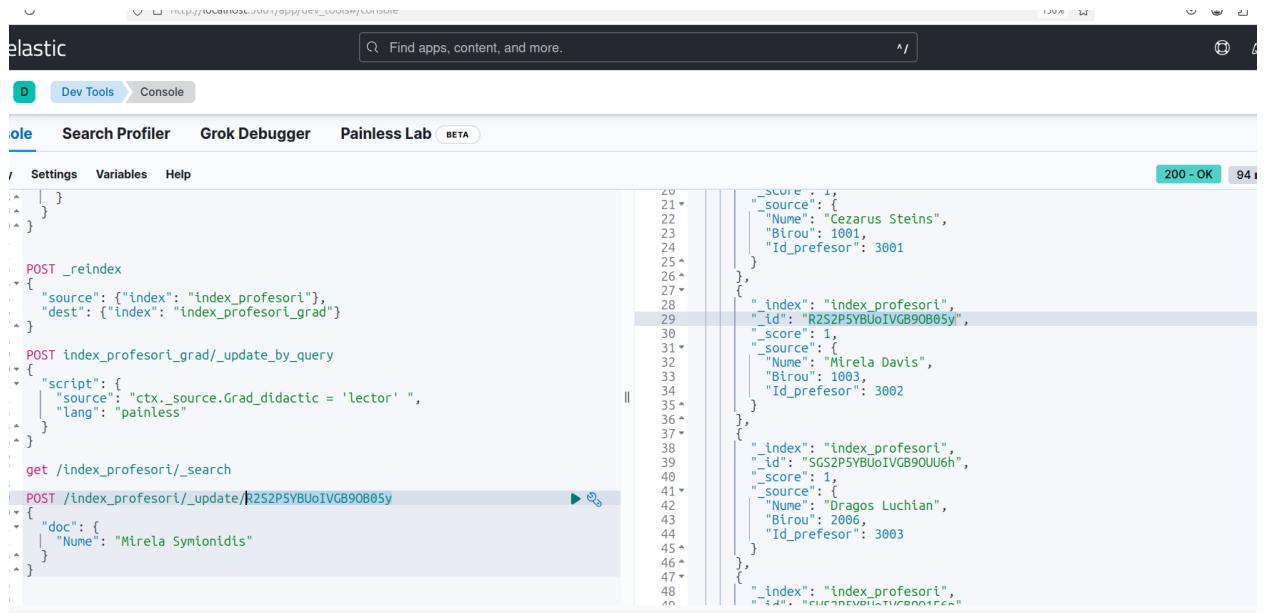
Console Search Profiler Grok Debugger Painless Lab BETA 200 - OK

```
tory Settings Variables Help
241 },
242 | "Id_profesor": {
243 | | "type": "integer"
244 },
245 | "Grad_didactic": {
246 | | "type": "keyword"
247 }
248 }
249 }
250 }
251
252
253 POST _reindex
254 {
255   "source": {"index": "index_profesori"},
256   "dest": {"index": "index_profesori_grad"}
257 }
258
259 POST index_profesori_grad/_update_by_query
260 {
261   "script": {
262     "source": "ctx._source.Grad_didactic = 'lector' ",
263     "lang": "painless"
264   }
265 }
266
267 get /index_profesori_grad/_search|▶🔗
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337 }
```

## Exerciții pentru Managing Documents

### Updating documents

- Actualizează un câmp dintr-un document existent (hint: POST către /\_update cu `doc').



The screenshot shows the Elasticsearch Dev Tools Console interface. A POST request is being executed against the `/index_profesori/_update/R2S2P5YBUoIVGB9OB05y` endpoint. The request body contains a `"doc"` object with the key `"Nume"` set to `"Mirela Symionidis"`. The response shows the updated document with the new value for `"Nume"`.

```
POST /index_profesori/_update/R2S2P5YBUoIVGB9OB05y
{
  "doc": {
    "Nume": "Mirela Symionidis"
  }
}

{
  "_index": "index_profesori",
  "_id": "R2S2P5YBUoIVGB9OB05y",
  "_score": 1,
  "_source": {
    "Nume": "Mirela Symionidis",
    "Birou": 1003,
    "Id_prefesor": 3002
  }
}
```

POST /index\_profesori/\_update/R2S2P5YBUoIVGB9OB05y

```
{
  "doc": {
    "Nume": "Mirela Symionidis"
  }
}
```

```
{
  "_index": "index_profesori",
  "_id": "R2S2P5YBUoIVGB9OB05y",
  "_score": 1,
  "_source": {
    "Nume": "Mirela Symionidis",
    "Birou": 1003,
    "Id_prefesor": 3002
  }
}
```

- Schimbă numărul de credite pentru un curs la 7.

```

Settings Variables Help
"dest": {"index": "index_profesori_grad"}
}
POST index_profesori_grad/_update_by_query
{
  "script": {
    "source": "ctx._source.Grad_didactic = 'lector' ",
    "lang": "painless"
  }
}
get /index_profesori/_search
POST /index_profesori/_update/R2S2P5YBUoIVGB90B05y
{
  "doc": {
    "Nume": "Mirela Symionidis"
  }
}
get /index_cursuri/_search
POST /index_cursuri/VGS_P5YBUoIVGB90qk4_
{
  "doc": {
    "Nr_credite": 7
  }
}

48   "index": "index_cursuri",
49   "_id": "U2S_PSYBUoIVGB90eU6I",
50   "score": 1,
51   "_source": {
52     "Denumire": "Macroeconomie cantitativa",
53     "Nr_credite": 4,
54     "Reprezentat_curs": 3004
55   }
56   },
57   {
58     "index": "index_cursuri",
59     "_id": "VGS_PSYBUoIVGB90qk4_",
60     "score": 1,
61     "_source": {
62       "Denumire": "Statistica Spatiala",
63       "Nr_credite": 4,
64       "Reprezentat_curs": 3001
65     }
66   },
67   {
68     "index": "index_cursuri",
69     "_id": "VM5_PSYBUoIVGB904E5r",
70     "score": 1,
71     "_source": {
72       "Denumire": "Statistica",
73       "Nr_credite": 6,
74       "Reprezentat_curs": 3005
75     }
76   }
}

```

POST /index\_cursuri/\_update/VGS\_P5YBUoIVGB90qk4\_

```
{

```

```
"doc": {
```

```
  "Nr_credite": 7
```

```
}
```

```
}
```

```
{
  "_index": "index_cursuri",
  "_id": "VGS_P5YBUoIVGB90qk4_",
  "_version": 2,
  "_seq_no": 13,
  "_primary_term": 11,
  "found": true,
  "_source": {
    "Denumire": "Statistica Spatiala",
    "Nr_credite": 7,
    "Reprezentat_curs": 3001
  }
}
```

3. Actualizează numele unui profesor.

POST /index\_profesori/\_update/R2S2P5YBUoIVGB90B05y

```
{
```

```
"doc": {
```

```
  "Nume": "Mirela Symionidis"
```

```
}
```

```
}
```

```

        },
        {
            "_index": "index_profesori",
            "_id": "R2S2P5YBUoIVGB90B05y",
            "_score": 1,
            "_source": {
                "Nume": "Mirela Symionidis",
                "Birou": 1003,
                "Id_prefesor": 3002
            }
        }
    ]
}

```

4. Adaugă un câmp nou `Observatii` într-un document din `index\_cursuri`.

POST /index\_cursuri/\_update/VGS\_P5YBUoIVGB9Oqk4\_

```
{
  "doc": {
    "Observatii" : "Materie noua"
  }
}
```

```
Settings Variables Help
get /index_profesori/_search
POST /index_profesori/_update/R2S2P5YBUoIVGB90B05y
{
  "doc": {
    | "Nume": "Mirela Symionidis"
  }
}
get /index_cursuri/_search
POST /index_cursuri/_update/VGS_P5YBUoIVGB90qk4_
{
  "doc": {
    | "Nr_credite": 7
  }
}
GET /index_cursuri/_doc/VGS_P5YBUoIVGB90qk4_
```

```
1 [
2   {
3     "_index": "index_cursuri",
4     "_id": "VGS_P5YBUoIVGB90qk4_",
5     "_version": 3,
6     "_seq_no": 14,
7     "_primary_term": 11,
8     "found": true,
9     "_source": {
10       "Denumire": "Statistica Spatiala",
11       "Nr_credite": 7,
12       "Reprezentat_curs": 3001,
13       "Observatii": "Materie noua"
14     }
15   }
16 ]
```

5. Crește valoarea `Birou` cu 100 folosind un script update (hint: `script.source`).

POST /index\_profesori/\_update\_by\_query

```
{
  "query": {
    "match_all": {}
  },
  "script": {
    "source": "ctx._source.Birou += 100",
    "lang": "painless"
  }
}
```

The screenshot shows the Elasticsearch Dev Tools Console interface. In the left panel, there is a code editor with a history of requests. The current request is a POST to /index\_cursuri/\_update/VGS\_P5YBUoIVGB90qk4\_. The right panel displays the search results for the query, showing three documents with their \_id, \_score, and \_source fields.

```

279+ {
280+   "doc": {
281+     | "Nr_credite": 7
282+   }
283+ }
284+
285 GET /index_cursuri/_doc/VGS_P5YBUoIVGB90qk4_
286
287 POST /index_cursuri/_update/VGS_P5YBUoIVGB90qk4_
288+ {
289+   "doc": {
290+     | "Observatii" : "Materie noua"
291+   }
292+ }
293
294 POST /index_profesori/_update_by_query
295+ {
296+   "query": {
297+     | "match_all": {}
298+   },
299+   "script": {
300+     | "source": "ctx._source.Birou += 100",
301+     | "lang": "painless"
302+   }
303+ }
304
305 get /index_profesori/_search
306
307

```

```

17+ [
18  {
19    "_index": "index_profesori",
20    "_id": "RmS1PSYBUoIVGB90r07x",
21    "_score": 1,
22    "_source": {
23      "Birou": 1101,
24      "Id_profesor": 3001,
25      "Name": "Cezarus Steins"
26    }
27  },
28  {
29    "_index": "index_profesori",
30    "_id": "SGS2PSYBUoIVGB90UUh",
31    "_score": 1,
32    "_source": {
33      "Birou": 1106,
34      "Id_profesor": 3003,
35      "Name": "Dragos Luchian"
36    }
37  },
38  {
39    "_index": "index_profesori",
40    "_id": "SNS2PSYBUoIVGB901E6n",
41    "_score": 1,
42    "_source": {
43      "Birou": 2101,
44      "Id_profesor": 3004,
45      "Name": "Adela Barbu"
46    }
47]

```

## Deleting documents

1. Sterge un document cunoscând ID-ul (hint: DELETE /index/\_doc/{id}).

The screenshot shows the Elasticsearch Dev Tools Console interface. In the left panel, there is a code editor with a history of requests. The current request is a Delete to /index\_cursuri/\_doc/VGS\_P5YBUoIVGB90qk4\_. A tooltip says "Click to send request". The right panel displays the response, which indicates the document was deleted successfully.

```

283+ {
284+   "Nr_credite": 7
285+ }
286+
287 GET /index_cursuri/_doc/VGS_P5YBUoIVGB90qk4_
288
289 POST /index_cursuri/_update/VGS_P5YBUoIVGB90qk4_
290+ {
291+   "doc": {
292+     | "Observatii" : "Materie noua"
293+   }
294+ }
295
296 POST /index_profesori/_update_by_query
297+ {
298+   "query": {
299+     | "match_all": {}
300+   },
301+   "script": {
302+     | "source": "ctx._source.Birou += 100",
303+     | "lang": "painless"
304+   }
305+ }
306
307 get /index_profesori/_search
308
309 Delete /index_cursuri/_doc/VGS_P5YBUoIVGB90qk4_
310
311

```

```

1+ {
2  "_index": "index_cursuri",
3  "_id": "VGS_P5YBUoIVGB90qk4_",
4  "_version": 4,
5  "result": "deleted",
6  "_shards": {
7    | "total": 2,
8    | "successful": 1,
9    | "failed": 0
10+ },
11  "_seq_no": 15,
12  "_primary_term": 12
13+

```

2. Sterge un curs cu 'Nr\_credite = 0' după ce l-ai identificat.

```

POST /index_profesori/_update_by_query
{
  "query": {
    | "match_all": {}
  },
  "script": {
    | "source": "ctx._source.Birou += 100",
    | "lang": "painless"
  }
}

get /index_cursuri/_search
Delete /index_cursuri/_doc/VGS_PSYBUoIVGB90qk4_
//2. Sterge un curs cu `Nr_credite = 0` după ce l-am identificat. Click to send request
//identificare
get /index_cursuri/_search
{
  "query": {
    | "match": {
      | "Nr_credite" :| 0
    }
  }
}

```

10+     "hits": {  
11+     | "total": {  
12+     | | "value": 2,  
13+     | | "relation": "eq"  
14+ },  
15+     "max\_score": 1,  
16+     "hits": [  
17+     | {  
18+     | | "\_index": "index\_cursuri",  
19+     | | "\_id": "W2TBP5YBUoIVGB90m04M",  
20+     | | "\_score": 1,  
21+     | | "\_source": {  
22+     | | | "Denumire": "Optional Economie ",  
23+     | | | "Nr\_credite": 0,  
24+     | | | "Reprezentat\_curs": 3001  
25+     | }},  
26+     | {  
27+     | | "\_index": "index\_cursuri",  
28+     | | "\_id": "XGTBP5YBUoIVGB90yE40",  
29+     | | "\_score": 1,  
30+     | | "\_source": {  
31+     | | | "Denumire": "Optional Matematica Aplicata ",  
32+     | | | "Nr\_credite": 0,  
33+     | | | "Reprezentat\_curs": 3002  
34+     | }},  
35+     | {  
36+     | | "\_index": "index\_cursuri",  
37+     | | "\_id": "W2TBP5YBUoIVGB90m04M",  
| }

```

//identificare
get /index_cursuri/_search
{
  "query": {
    "match": {
      "Nr_credite" : 0
    }
  }
}

//stergere
Delete /index_cursuri/_doc/W2TBP5YBUoIVGB90m04M

```

3. Sterge un document din `index\_profesori` după nume (folosește întâi un search).

```

//identificare
get /index_profesori/_search
{
  "query": {
    "match": {
      "Nume": "Cezarus Steins"
    }
  }
}

```

- Delete /index\_profesori/\_doc/RmS1P5YBUoIVGB9Or07x
- Găsește și șterge toate documentele cu `Id\_Dep` sub 1000 (doar după ce le verifici).  
-nu sunt departamente cu Id\_dep sub 1000

get /index\_departamente/\_search

```
{
  "query": {
    "range": {
      "Id_Dep": {
        "lt": 1000
      }
    }
  }
}
```

delete /index\_departamente/\_doc/

- Șterge un document din `index\_departamente` și confirmă cu un GET.

The screenshot shows the Elasticsearch Dev Tools interface with the following details:

- Header:** Sole, Search Profiler, Grok Debugger, Painless Lab (BETA)
- Left Panel (Query Editor):**

```

{
  "query": {
    "match": {
      "Name": "Cezarust Stelns"
    }
  }
}

Delete /index_profesori/_doc/RmS1P5YBUoIVGB9Or07x
get /index_departamente/_search

get /index_departamente/_search
{
  "query": {
    "range": {
      "Id_Dep": {
        "lt": 1000
      }
    }
  }
}

delete /index_departamente/_doc/YWTTTP5YBUoIVGB90vE75
get /index_departamente/_search/YWTTTP5YBUoIVGB90vE75
  
```
- Right Panel (Results):**

```

1 [ { "error": "no handler found for uri [/index_departamente/_search /YWTTTP5YBUoIVGB90vE75?pretty=true] and method [GET]" }
3 ^ ]
  
```
- Bottom Status Bar:** Click to send request
- Bottom Note:** //5. Șterge un document din `index\_departamente` și confirmă cu un GET.

delete /index\_departamente/\_doc/YWTTTP5YBUoIVGB90vE75  
get /index\_departamente/\_search/YWTTTP5YBUoIVGB90vE75

## Searching for terms

- Caută documente care conțin un anumit cuvânt (hint: query de tip `match`).

Console Search Profiler Grok Debugger Painless Lab BETA

200 - OK 54 ms

Story Settings Variables Help

```

331
332     "lt": 1000
333 }
334 }
335 }
336 }
337 delete /index_departamente/_doc/YWTPSYBu0IVGB90vE75
338 get /index_departamente/_search/YWTPSYBu0IVGB90vE75
340
341 //5. Șterge un document din 'index_departamente' și confirmă cu un click pe butonul 'Click to send request'
342
343 //1. Caută documente care conțin un anumit cuvânt (hint: query de tip 'match')
344 get /index_cursuri/_search
345 {
346     "query": {
347         "match": {
348             "Nr_credite": 6
349         }
350     }
351 }
352 //2. Găsește toți profesorii care au în nume 'Ion'.
353 //3. Caută cursuri care conțin 'Aplicata'.
354 //4. Caută departamente cu criteriul 'Central'.
355 //5. Caută toate denumirile care includ cuvântul 'Optional'.
356
357
358
359

```

9 +
10 +
11 +
12 +
13 +
14 +
15 +
16 +
17 +
18 +
19 +
20 +
21 +
22 +
23 +
24 +
25 +
26 +
27 +
28 +
29 +
30 +
31 +
32 +
33 +
34 +
35 +
36 +
37 +

},  
"hits": {  
"total": {  
"value": 3,  
"relation": "eq"  
}},  
"max\_score": 1,  
"hits": [  
{  
"\_index": "index\_cursuri",  
"\_id": "UGS-PSYBu0IVGB90YE6V",  
"\_score": 1,  
"\_source": {  
"Denumire": "Algebra Liniara",  
"Nr\_credite": 6,  
"Reprezentat\_curs": 3001  
}},  
{  
"\_index": "index\_cursuri",  
"\_id": "UMS-PSYBu0IVGB90k07t",  
"\_score": 1,  
"\_source": {  
"Denumire": "Analiza matematica",  
"Nr\_credite": 6,  
"Reprezentat\_curs": 3002  
}},  
{

get /index\_cursuri/\_search

```
{
    "query": {
        "match": {
            "Nr_credite": 6
        }
    }
}
```

2. Găsește toți profesorii care au în nume 'Ion'.

3.

GET /index\_profesori/\_search

```
{
    "query": {
        "match_phrase": {
            "Nume": "Ion"
        }
    }
}
```

3.Caută cursuri care conțin 'Aplicata'.

History Settings Variables Help 200 - OK 5

```
346 ~ "query": {  
347 ~ | "match": {  
348 ~ | | "Nr_credite": 6  
349 ~ }  
350 ~ }  
351 ~ }  
352 GET /index_profesori/_search  
353 {  
354 ~ "query": {  
355 ~ | "match_phrase": {  
356 ~ | | "Nume": "Ion"  
357 ~ }  
358 ~ }  
359 }  
360 //3. Caută cursuri care conțin 'Aplicata'.  
361 Get /index_cursuri/_search  
362 {  
363 ~ "query": {  
364 ~ | "match": {  
365 ~ | | "Denumire": "Aplicata"  
366 ~ }  
367 }  
368 }  
369 //4. Caută departamente cu criteriul 'Central'.  
370 //5. Caută toate denumirile care includ cuvântul 'Optional'.  
  
Click to send request
```

1 ~ {  
2 "took": 20,  
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": 1.8899627,  
16 ~ "hits": [  
17 ~ | {  
18 ~ | | "index": "index\_cursuri",  
19 ~ | | "id": "XGTBP5YBU0lVG90yE40",  
20 ~ | | "score": 1.8899627,  
21 ~ | | "source": {  
22 ~ | | | "Denumire": "Optional Matematica Aplicata ",  
23 ~ | | | "Nr\_credite": 0,  
24 ~ | | | "Reprezentant\_curs": 3002  
25 ~ | | }  
26 ~ | }  
27 ~ ]  
28 ~ }  
29 ~ }

Get /index\_cursuri/\_search

```
{  
  "query": {  
    "match": {  
      "Denumire": "Aplicata"  
    }  
  }  
}
```

4. Caută departamente cu criteriul 'Central'.

Criteriu este de tip numeric, astfel nu este posibila aceasta instructiune.

5. Caută toate denumirile care includ cuvântul 'Optional'.

```
1 }  
2 GET /index_profesori/_search  
3 {  
4   "query": {  
5     "match_phrase": {  
6       "Nume": "Ion"  
7     }  
8   }  
9 }  
10 //3. Caută cursuri care conțin 'Aplicata'.  
11 Get /index_cursuri/_search  
12 {  
13   "query": {  
14     "match": {  
15       "Denumire": "Aplicata"  
16     }  
17   }  
18 }  
19 //4. Caută departamente cu criteriul 'Central'.  
20 //5. Caută toate denumirile care includ cuvântul 'Optional'.  
21 Get /index_cursuri/_search  
22 {  
23   "query": {  
24     "match": {  
25       "Denumire": "Optional"  
26     }  
27   }  
28 }  
29 }  
30 }  
31 {  
32   "took": 4,  
33   "timed_out": false,  
34   "shards": {  
35     "total": 1,  
36     "successful": 1,  
37     "skipped": 0,  
38     "failed": 0  
39   },  
40   "hits": {  
41     "total": {  
42       "value": 1,  
43       "relation": "eq"  
44     },  
45     "max_score": 1.4577258,  
46     "hits": [  
47       {  
48         "_index": "index_cursuri",  
49         "_id": "XGTBPSYBuIIVGB90yE40",  
50         "_score": 1.4577258,  
51         "_source": {  
52           "Denumire": "Optional Matematica Aplicata",  
53           "Nr_credite": 0,  
54           "Reprezentat_curs": 3002  
55         }  
56       }  
57     ]  
58   }  
59 }  
60 }
```

## Range searches

1. Găsește documente unde o valoare numerică este peste un prag (hint: `range` query).  
//1. Găsește documente unde o valoare numerică este peste un prag (hint: `range` query).

Get /index\_cursuri/\_search

```
{  
  "query": {  
    "range": {  
      "Nr_credite": {  
        "gte": 4  
  
      }  
    }  
  }  
}
```

2. Caută cursuri cu `Nr\_credite` între 1 și 5.

Get /index\_cursuri/\_search

```
{  
  "query": {  
    "range": {  
      "Nr_credite": {  
        "gte": 2,  
        "lte": 5  
  
      }  
    }  
  }  
}
```

3. Găsește profesori cu `Birou` sub 6000.

Get /index\_profesori/\_search

```
{  
  "query" : {  
    "range": {  
      "Birou": {  
  
        "lte": 6000  
      }  
    }  
  }  
}
```

```
        }
    }
}
}
```

4. Caută departamente cu `Id\_Dep` între 1000 și 2000.

```
get /index_departamente/_search
{
  "query": {
    "range": {
      "Id_Dep": {
        "gte": 1000,
        "lte": 2000
      }
    }
  }
}
```

5. Afisează cursuri unde `Reprezentant\_curs` este peste 3000.

```
get /index_cursuri/_search
{
  "query": {
    "range": {
      "Reprezentat_curs": {
        "gte": 3000
      }
    }
  }
}
```

### Querying by field existence

1. Găsește documente care conțin un câmp anume (hint: `exists` query).

```
GET /index_cursuri/_search
{
  "query": {
    "exists": {
      "field": "Reprezentat_curs"
    }
  }
}
```

- ```
}
```
2. Caută documente care au câmpul `Birou` definit.  
GET /index\_profesori/\_search  
{  
 "query": {  
 "exists": {  
 "field": "Birou"  
 }  
 }  
}
  3. Găsește cursuri care au `Reprezentant\_curs`.  
GET /index\_cursuri/\_search  
{  
 "query": {  
 "exists": {  
 "field": "Reprezentat\_curs"  
 }  
 }  
}
  4. Verifică dacă toate documentele din `index\_departamente` au `Denumire`.
  5.  
get /index\_departamente/\_search  
{  
 "query": {  
 "exists": {  
 "field": "Denumire"  
 }  
 }  
}
  6. Caută profesori unde există câmpul `Nume`.  
GET /index\_profesori/\_search  
{  
 "query": {  
 "exists": {  
 "field": "Nume"  
 }  
 }  
}

## The match query

1. Caută după text într-un câmp (hint: `match` + numele câmpului).

```
get /index_profesori/_search
```

```
{  
  "query": {  
    "match": {  
      "Birou": "1101"  
    }  
  }  
}
```

2. Caută în `index\_profesori` pentru numele 'Selena'.

```
get /index_profesori/_search
```

```
{  
  "query": {  
    "match": {  
      "Nume": "Selena"  
    }  
  }  
}
```

3. Caută în `index\_cursuri` cursuri care includ 'Informatică'.

```
get /index_cursuri/_search
```

```
{  
  "query": {  
    "match": {  
      "Denumire": "Informatica"  
    }  
  }  
}
```

4. Caută în `index\_departamente` pentru 'Informatica'.

```
get /index_cursuri/_search
```

```
{  
  "query": {  
    "match": {  
      "Denumire": "Informatica"  
    }  
  }  
}
```

5. Caută documente unde denumirea conține 'Bază'.

```
get /index_cursuri/_search
```

```
{
```

```

"query": {
  "match": {
    "Denumire": "Baza"
  }
}
}

```

## Searching multiple fields

- Caută o valoare în mai multe câmpuri simultan (hint: `multi\_match`).

```

get /index_departamente/_search
{
  "query": {
    "multi_match": {
      "query": "1000",
      "fields": ["Id_Dep", "Criteriu_birou"]
    }
  }
}

```

- Caută valoarea `3002` în câmpurile `Denumire` și `Reprezentant\_curs`.

```

get index_cursuri/_search
{
  "query": {
    "multi_match": {
      "query": 3002,
      "fields": ["Id_Dep", "Criteriu_birou"]
    }
  }
}

```

- Caută cuvântul 'Matematica' în `Denumire` și `Criteriu\_birou`.

```

get index_cursuri/_search
{
  "query": {
    "multi_match": {
      "query": "Matematica",
      "fields": ["Denumire", "Criteriu_birou"]
    }
  }
}

```

4. Caută profesorul cu ID sau nume egal cu '3010'.

```
get index_profesori/_search
{
  "query": {
    "multi_match": {
      "query": "3010",
      "fields": ["Id_prefesor", "Nume"]
    }
  }
}
```

5. Caută textul 'Optional' în toate câmpurile text din `index\_cursuri`.

```
GET index_cursuri/_search
{
  "query": {
    "multi_match": {
      "query": "Optional",
      "fields": ["Denumire"]
    }
  }
}
```

**Observatii:** Nu merge sa cautam intr-un camp de tip numeric un caracter de tip string.

## Secțiunea 7 – Fragmentare si Reaplicare

Shard = o bucătică din datele tale.

ElasticSearch NU stochează toate datele într-un singur loc, ci le împarte în mai multe bucăți mai mici, care se numesc sharduri.

### Avantaje:

Scalabilitate → Putem adăuga mai multe servere și datele se împărătie automat.

Performanță → Putem căuta în paralel în mai multe sharduri → răspunsuri mult mai rapide.

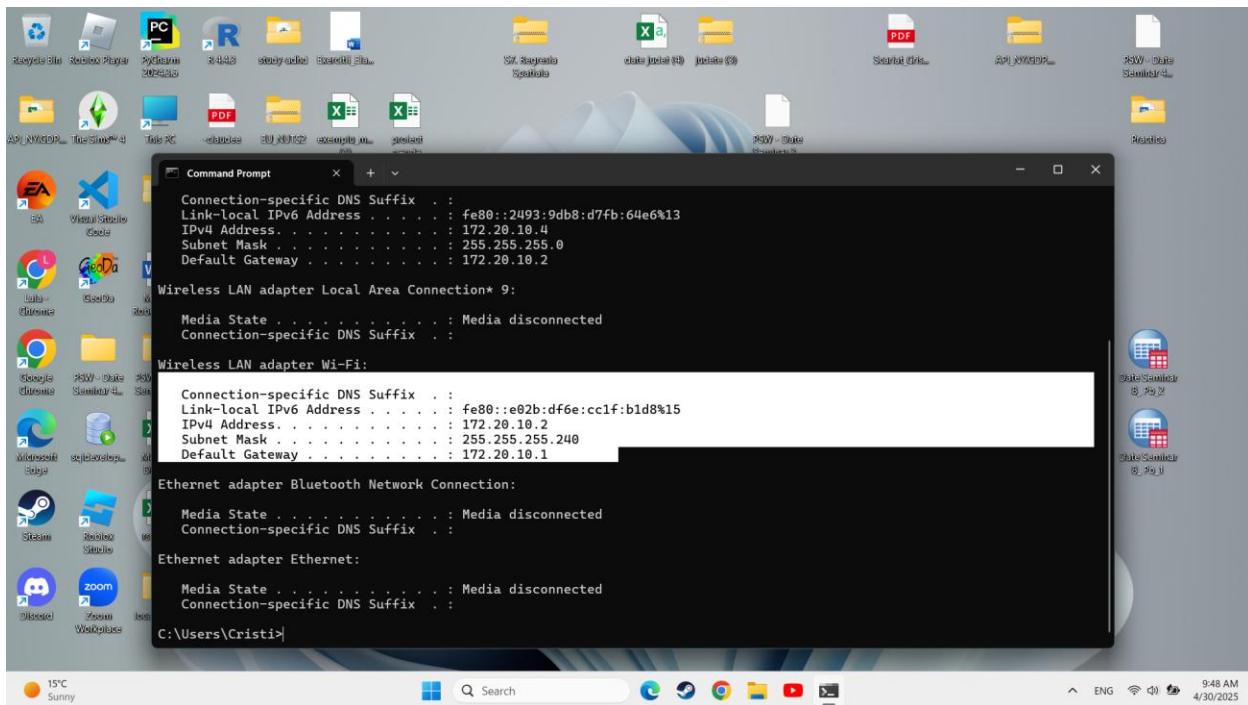
Rezistență la probleme (fault tolerance) → Dacă pică un server, nu pierdem datele, pentru că avem și replică ale shardurilor.

Sitory Settings Variables Help

```

500 ~ {
501   "query" : {
502     "multi_match": {
503       "query": "3010",
504       "fields": ["Id_prefesor", "Nume"]
505     }
506   }
507 }
508
509 GET /index_cursuri/_search
510 {
511   "query" : {
512     "multi_match": {
513       "query": "Optional",
514       "fields": ["Denumire"]
515     }
516   }
517 }
518
519 get /index_cursuri/_search
520 get _cat/indices?v
521
522 PUT /index_cursuri_1shard
523 {
524   "settings": {
525     "number_of_shards": 1,
526     "number_of_replicas": 0
527   },
528   "mappings": {
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
798
799
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
888
889
889
890
891
892
893
894
895
896
897
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
988
989
989
990
991
992
993
994
995
996
997
998
999
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1297
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1387
1388
1389
1389
1390
1391
1392
1393
1394
1395
1396
1397
1397
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1489
1490
1491
1492
1493
1494
1495
1496
1497
1497
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1588
1589
1589
1590
1591
1592
1593
1594
1595
1596
1597
1597
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1688
1689
1689
1690
1691
1692
1693
1694
1695
1696
1697
1697
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1797
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1888
1889
1889
1890
1891
1892
1893
1894
1895
1896
1897
1897
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1988
1989
1989
1990
1991
1992
1993
1994
1995
1996
1997
1997
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2088
2089
2089
2090
2091
2092
2093
2094
2095
2096
2097
2097
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2197
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2296
2297
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2388
2389
2389
2390
2391
2392
2393
2394
2395
2396
2397
2397
2398
2399
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2488
2489
2489
2490
2491
2492
2493
2494
2495
2496
2496
2497
2498
2499
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2588
2589
2589
2590
2591
2592
2593
2594
2595
2596
2597
2597
2598
2599
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
```

Mai întâi am verificat ip-uriile

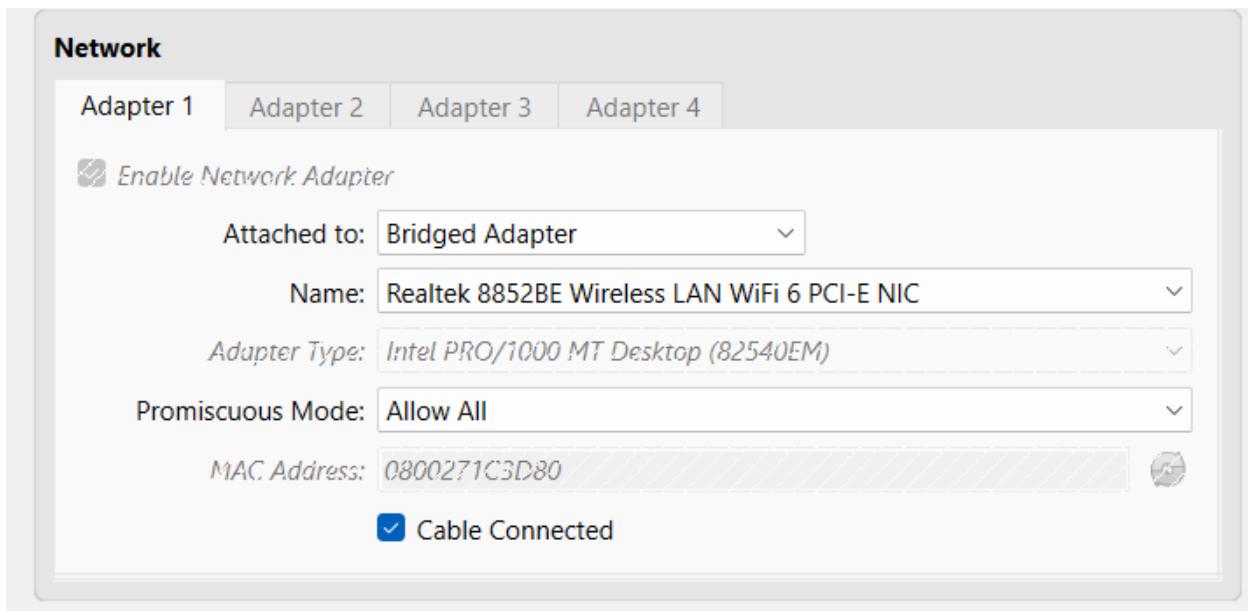


Cel de pe vm nu avea aceeasi clasa de ip, insa l-am modficat din control pannel si din setarile masinii virtuale.

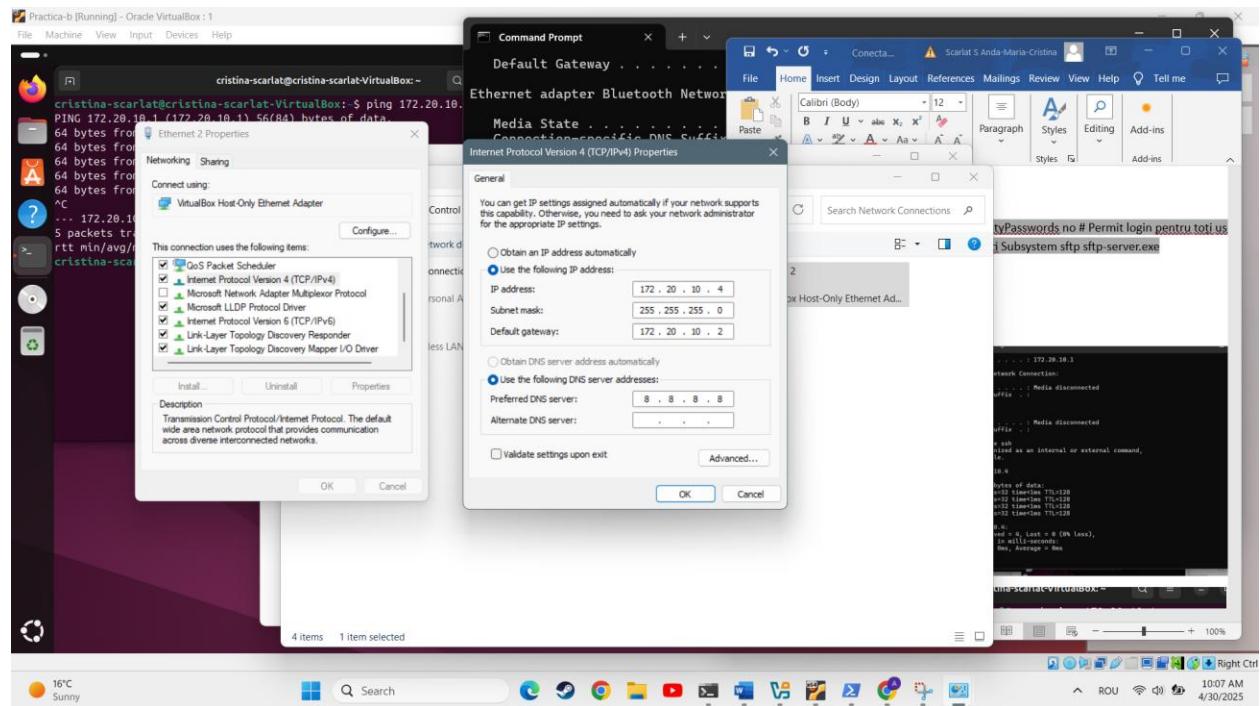
```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.20.10.5 netmask 255.255.255.240 broadcast 172.20.10.15
        inet6 fe80::a00:27ff:fe1c:3d80 prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:1c:3d:80 txqueuelen 1000 (Ethernet)
            RX packets 30 bytes 5438 (5.4 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 112 bytes 13191 (13.1 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 225 bytes 19200 (19.2 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 225 bytes 19200 (19.2 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

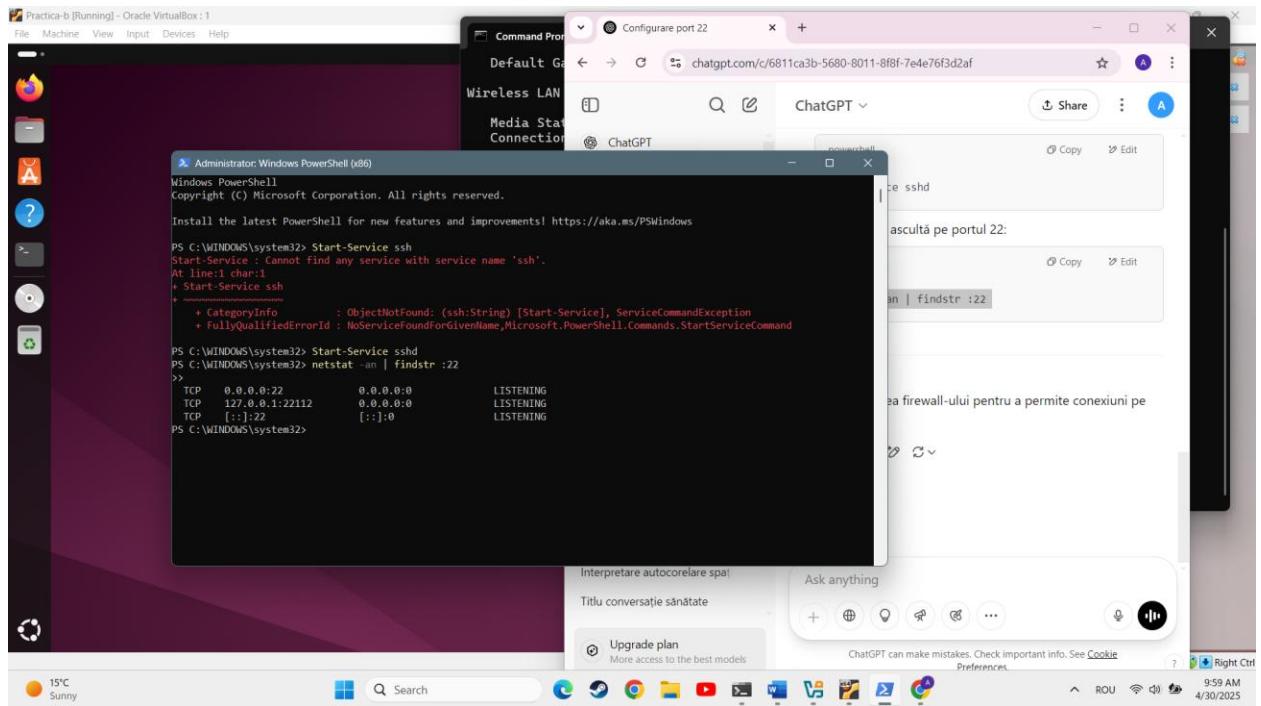
-setari vm



-setari control panel



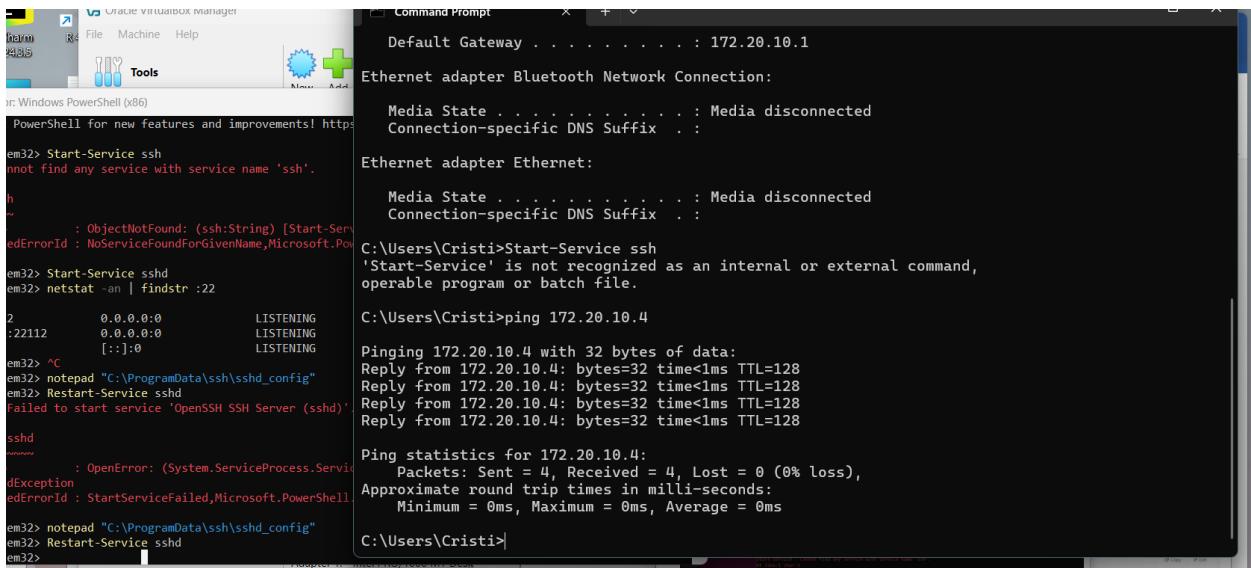
Am deschis powershell ca si administrator si am instalat OpenSSH. Am pornit serviciul si am verificat daca ruleaza



Asta am scris in notepad-ul pt configurarea sshd

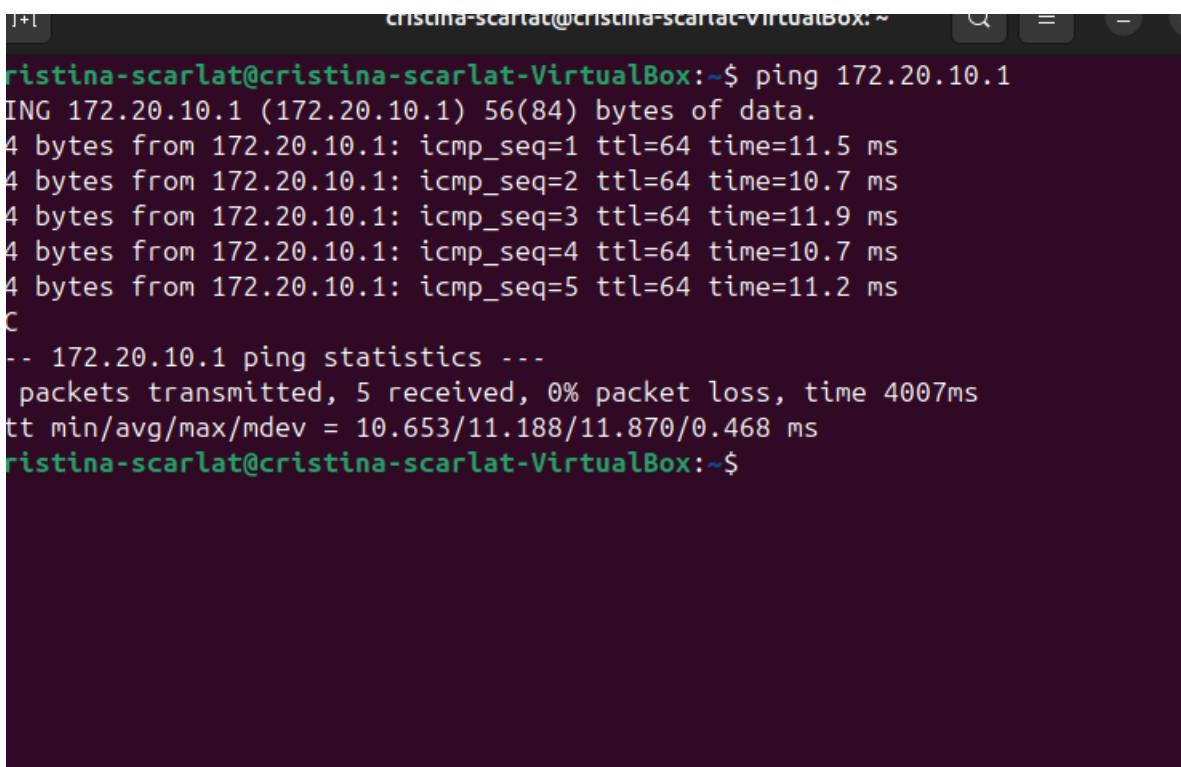
```
Port 22 AddressFamily any ListenAddress 0.0.0.0 ListenAddress :: HostKey
__PROGRAMDATA__/_ssh/ssh_host_ed25519_key HostKey
__PROGRAMDATA__/_ssh/ssh_host_rsa_key # Permit login cu parolă (opțional, în funcție de
setup) PasswordAuthentication yes PermitEmptyPasswords no # Permit login pentru toți userii
Windows PermitRootLogin yes AllowUsers cristi Subsystem sftp sftp-server.exe
```

Dupa am folosit ping pt a verifica conexiunea



Windows PowerShell (x86)  
PowerShell for new features and improvements! https://go.microsoft.com/fwlink/?linkid=845972  
em32> Start-Service sshh  
cmdlet 'Start-Service' not found. A cmdlet, function, or script file might have been moved, renamed, or deleted.  
em32> Start-Service sshd  
em32> netstat -an | findstr :22  
22112 0.0.0.0:0 LISTENING  
:22112 0.0.0.0:0 LISTENING  
[::]:0 LISTENING  
em32> ^C  
em32> notepad "C:\ProgramData\ssh\sshd\_config"  
em32> Restart-Service sshd  
Failed to start service 'OpenSSH SSH Server (sshd)'  
sshd  
----  
: OpenError: (System.ServiceProcess.ServiceController) dException  
dException : StartServiceFailed,Microsoft.PowerShell  
em32> notepad "C:\ProgramData\ssh\sshd\_config"  
em32> Restart-Service sshd  
em32>

Default Gateway . . . . . : 172.20.10.1  
Ethernet adapter Bluetooth Network Connection:  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . . .  
Ethernet adapter Ethernet:  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . . .  
C:\Users\Cristi>Start-Service sshh  
'Start-Service' is not recognized as an internal or external command,  
operable program or batch file.  
C:\Users\Cristi>ping 172.20.10.4  
Pinging 172.20.10.4 with 32 bytes of data:  
Reply from 172.20.10.4: bytes=32 time<1ms TTL=128  
Ping statistics for 172.20.10.4:  
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
Minimum = 0ms, Maximum = 0ms, Average = 0ms  
C:\Users\Cristi>



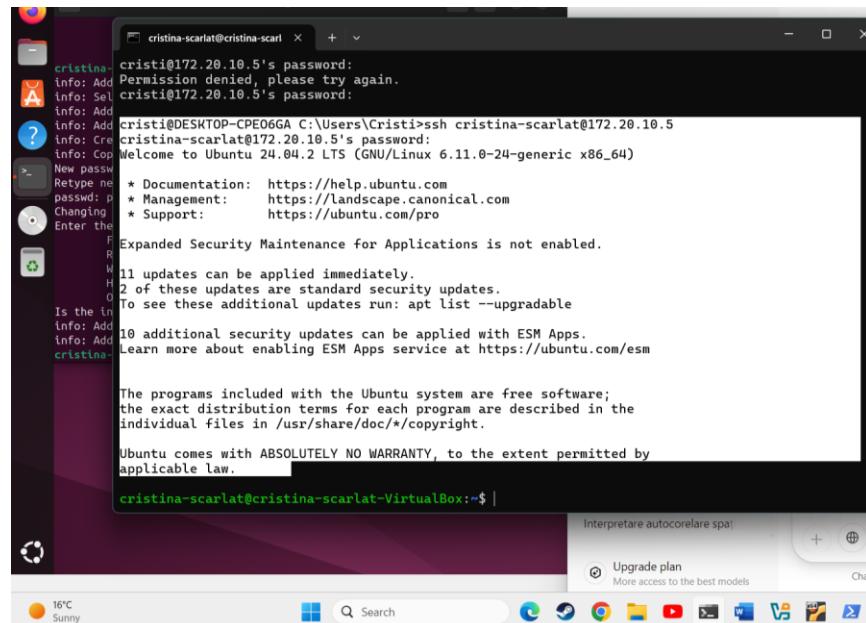
```
ristina-scarlat@cristina-scarlat-VirtualBox:~$ ping 172.20.10.1  
PING 172.20.10.1 (172.20.10.1) 56(84) bytes of data.  
4 bytes from 172.20.10.1: icmp_seq=1 ttl=64 time=11.5 ms  
4 bytes from 172.20.10.1: icmp_seq=2 ttl=64 time=10.7 ms  
4 bytes from 172.20.10.1: icmp_seq=3 ttl=64 time=11.9 ms  
4 bytes from 172.20.10.1: icmp_seq=4 ttl=64 time=10.7 ms  
4 bytes from 172.20.10.1: icmp_seq=5 ttl=64 time=11.2 ms  
C  
-- 172.20.10.1 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4007ms  
rtt min/avg/max/mdev = 10.653/11.188/11.870/0.468 ms  
ristina-scarlat@cristina-scarlat-VirtualBox:~$
```

-m am conectat la portul 22 prin cmd

```
C:\Users\Cristi>ssh cristti@172.20.10.2
The authenticity of host '172.20.10.2 (172.20.10.2)' can't be established.
ED25519 key fingerprint is SHA256:D5WsbNW60gjCkh3PwszhwA0KPRhXHkQzhbnWAI73LI4.
This host key is known by the following other names/addresses:
  C:\Users\Cristi/.ssh/known_hosts:1: localhost
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.20.10.2' (ED25519) to the list of known hosts.
cristti@172.20.10.2's password:
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.
```

```
cristti@DESKTOP-CPE06GA C:\Users\Cristi>ip a
```

-m am conectat si la vm



Apoi am creat un enviroment pentru a putea instala pachetul elasticsearch si faker si am creat un editor cu nano în care am scris:

---

```
from elasticsearch import Elasticsearch, helpers
from faker import Faker
import random

fake = Faker()

# Conectare la Elasticsearch
es = Elasticsearch("http://172.20.10.5:9200", verify_certs=False)
# Verificare conexiune
if not es.ping():
```

```

print("✗ Conexiunea a eșuat.")
exit()
else:
    print("✓ Conexiunea a fost realizată cu succes.")

# Nume index
index_name = "index_cursuri_test"

# Generare documente
def generate_documents(n=1000):
    for _ in range(n):
        yield {
            "_index": index_name,
            "_source": {
                "Denumire": fake.catch_phrase(), # titlu curs
                "Nr_credite": random.randint(1, 10),
                "Reprezentant_curs": random.randint(1000, 9999)
            }
        }

# Inserare în Elasticsearch
try:
    response = helpers.bulk(es, generate_documents(1000))
    print(f"✓ Inserat {response[0]} documente în indexul '{index_name}'")
except Exception as e:
    print(f"Eroare la inserare: {e}")

```

---

The screenshot shows the Elasticsearch Dev Tools Console interface. On the left, there is a code editor with a history of Elasticsearch queries. On the right, the results of a search query are displayed. A security warning dialog is overlaid on the interface.

```

565 { "Index": {} }
566 { "Denumire": "Programare Web", "Nr_credite": 4, "Reprezentant_curs": 107 }
567 { "Index": {} }
568 { "Denumire": "Securitate Informatică", "Nr_credite": 5, "Reprezentant_curs": 109 }
569 { "Index": {} }
570 { "Denumire": "Proiectare OOP", "Nr_credite": 6, "Reprezentant_curs": 109 }
571 { "Index": {} }
572 { "Denumire": "Optional: Machine Learning", "Nr_credite": 3, "Reprezentant_curs": 110 }

573
574
575 get /index_cursuri_1shard/_search
576
577 Post _reindex
578 {
579   "source" : { "index" : "index_cursuri_1shard" },
580   "dest" : { "index" : "index_cursuri_3sharduri" }
581 }
582
583
584 get /index_cursuri_3sharduri/_search
585
586 GET _cat/shards/index_cursuri_1shard?v
587 GET _cat/shards/index_cursuri_3sharduri?v
588
589 get /index_cursuri_test/_count
590
591

```

Result:

```

1 * {
2   "count": 1000,
3   "_shards": [
4     {
5       "total": 5,
6       "successful": 5,
7       "skipped": 0,
8       "failed": 0
9     }

```

Message: Your data is not secure

Details: Don't lose one bit. Enable our security features.

Buttons: Don't show again, Enable security, Dismiss

Pentru a vedea performanța acestuia, l-am comparat cu același index (însă doar cu 1 shrad), în ceea ce privește executarea unei căutări.

The screenshots show the Elasticsearch Dev Tools Console interface. Both screenshots show the same sequence of queries: a search on a single shard, a reindex operation, a search on three shards, and finally the count of documents in the test index. The results are identical in both cases, indicating no performance difference.

```

108 ]
109 { "Index": {} }
110 { "Denumire": "Proiectare OOP", "Nr_credite": 6, "Reprezentant_curs": 109 }
111 { "Index": {} }
112 { "Denumire": "Optional: Machine Learning", "Nr_credite": 3, "Reprezentant_curs": 110 }

get /index_cursuri_1shard/_search
Post _reindex
{
  "source" : { "index" : "index_cursuri_test" },
  "dest" : { "index" : "index_cursuri_test1" }
}

get /index_cursuri_3sharduri/_search
GET _cat/shards/index_cursuri_1shard?v
GET _cat/shards/index_cursuri_3sharduri?v
get /index_cursuri_test/_count
get /index_cursuri_test/_search
get /index_cursuri_test1/_search

```

Result (1 shard):

```

1 * {
2   "took": 12,
3   "timed_out": false,
4   "_shards": [
5     {
6       "total": 5,
7       "successful": 5,
8       "skipped": 0,
9       "failed": 0
10    },
11    "hits": {
12      "total": {
13        "value": 1000,
14        "relation": "eq"
15      },
16      "max_score": 1,
17      "hits": [
18        {
19          "_index": "index_cursuri_test",
20          "_id": "tkk-hpVbjEBhBJuYhq67",
21          "_score": 1,
22          "_source": {
23            "Denumire": "Front-line needs-based system engine",
24            "Nr_credite": 1,
25            "Reprezentant_curs": 7962
26          }
27        },
28        {
29          "_index": "index_cursuri_test",
30          "_id": "uek-hpVbjEBhBJuYhq67",
31          "_score": 1,
32          "_source": {
33            "Denumire": "Front-line needs-based system engine",
34            "Nr_credite": 1,
35            "Reprezentant_curs": 7962
36          }
37        }
38      ]
39    }
40  }
41
42  {
43    "took": 12,
44    "timed_out": false,
45    "_shards": [
46      {
47        "total": 1,
48        "successful": 1,
49        "skipped": 0,
50        "failed": 0
51      },
52      "hits": {
53        "total": {
54          "value": 1000,
55          "relation": "eq"
56        },
57        "max_score": 1,
58        "hits": [
59          {
60            "_index": "index_cursuri_test1",
61            "_id": "tkk-hpVbjEBhBJuYhq67",
62            "_score": 1,
63            "_source": {
64              "Denumire": "Front-line needs-based system engine",
65              "Nr_credite": 1,
66              "Reprezentant_curs": 7962
67            }
68          },
69          {
70            "_index": "index_cursuri_test1",
71            "_id": "suk-hpVbjEBhBJuYhq67",
72            "_score": 1,
73            "_source": {
74              "Denumire": "Front-line needs-based system engine",
75              "Nr_credite": 1,
76              "Reprezentant_curs": 7962
77            }
78          }
79        ]
80      }
81    }
82  }
83
84  {
85    "took": 12,
86    "timed_out": false,
87    "_shards": [
88      {
89        "total": 1,
90        "successful": 1,
91        "skipped": 0,
92        "failed": 0
93      },
94      "hits": {
95        "total": {
96          "value": 1000,
97          "relation": "eq"
98        },
99        "max_score": 1,
100       "hits": [
101         {
102           "_index": "index_cursuri_test1",
103           "_id": "suk-hpVbjEBhBJuYhq67",
104           "_score": 1,
105           "_source": {
106             "Denumire": "Front-line needs-based system engine",
107             "Nr_credite": 1,
108             "Reprezentant_curs": 7962
109           }
110         }
111       ]
112     }
113   }
114 }
```

Putem observa o diferență de 50 de ms între cele 2 comenzi, ceea ce ne arată ca nr. de fragmente îmbunătățește performanța.

## Exercițiu Practic - Fragmentarea (Sharduri) în Elasticsearch

Scopul exercițiului:

Înțelegerea shardurilor în Elasticsearch și a impactului lor asupra performanței, scalabilității și distribuirii datelor.

Context:

Vom lucra pe indexul 'index\_studenti\_test' pentru a testa conceptul de sharduri.

Pasul 1: Creează două indexuri cu structura identică dar număr diferit de sharduri

1. Index cu 2 sharduri:

```
PUT /index_studenti_2shards
{
  "settings": {
    "number_of_shards": 2,
    "number_of_replicas": 0
  },
  "mappings": {
    "properties": {
      "Nr_matricol": { "type": "integer" },
      "Nume": { "type": "text" },
      "Varsta": { "type": "integer" },
      "Media": { "type": "float" },
      "tara_natala": { "type": "text" }
    }
  }
}
```

2. Index cu 4 sharduri:

```
PUT /index_studenti_4shards
{
  "settings": {
    "number_of_shards": 4,
    "number_of_replicas": 0
  },
  "mappings": {
    "properties": {
      "Nr_matricol": { "type": "integer" },
      "Nume": { "type": "text" },
      "Varsta": { "type": "integer" },
      "Media": { "type": "float" },
      "tara_natala": { "type": "text" }
    }
  }
}
```

```

        }
    }
}
```

Pasul 2: Inserează documente identice în ambii indici

POST /index\_studenti\_2shards/\_doc

```
{
    "Nr_matricol": 1001,
    "Nume": "Ion Popescu",
    "Varsta": 21,
    "Media": 9.25,
    "tara_natala": "Romania"
}
```

POST /index\_studenti\_4shards/\_doc

```
{
    "Nr_matricol": 1001,
    "Nume": "Ion Popescu",
    "Varsta": 21,
    "Media": 9.25,
    "tara_natala": "Romania"
}
```

Pasul 3: Verifică distribuția shardurilor

GET \_cat/shards/index\_studenti\_2shards?v

The screenshot shows the Elasticsearch Dev Tools interface with the 'Console' tab selected. The left pane displays the following Elasticsearch commands:

```

1 PUT /index_studenti_4shards
2 {
3     "settings": {
4         "number_of_shards": 4,
5         "number_of_replicas": 0
6     },
7     "mappings": {
8         "properties": {
9             "Nr_matricol": { "type": "integer" },
10            "Nume": { "type": "text" },
11            "Varsta": { "type": "integer" },
12            "Media": { "type": "float" },
13            "tara_natala": { "type": "text" }
14        }
15    }
16 }
17 get index_studenti_2shards/_count
18 post _reindex
19 {
20     "source": {"index": "index_studenti_4shards"},
21     "dest": {"index": "index_studenti_2shards"}
22 }
23
24 GET _cat/shards/index_studenti_2shards?v
25 GET _cat/shards/index_studenti_4shards?v

```

The right pane shows the shard distribution results:

| index                  | shard | prirep | state   | docs | store  | dataset | ip         | node         |
|------------------------|-------|--------|---------|------|--------|---------|------------|--------------|
| index_studenti_4shards | 0     | p      | STARTED | 246  | 42.2kb | 42.2kb  | 172.19.0.2 | 03f08426be93 |
| index_studenti_4shards | 1     | p      | STARTED | 283  | 46.3kb | 46.3kb  | 172.19.0.2 | 03f08426be93 |
| index_studenti_4shards | 2     | p      | STARTED | 223  | 39.2kb | 39.2kb  | 172.19.0.2 | 03f08426be93 |
| index_studenti_4shards | 3     | p      | STARTED | 248  | 43kb   | 43kb    | 172.19.0.2 | 03f08426be93 |

GET \_cat/shards/index\_studenti\_4shards?v

```

PUT /index_studenti_4shards
{
  "settings": {
    "number_of_shards": 4,
    "number_of_replicas": 0
  },
  "mappings": {
    "properties": {
      "Nr_matricol": { "type": "integer" },
      "Nume": { "type": "text" },
      "Varsta": { "type": "integer" },
      "Media": { "type": "float" },
      "tara_natala": { "type": "text" }
    }
  }
}

get index_studenti_2shards/_count
post _reindex
{
  "source": {"index": "index_studenti_4shards"},
  "dest": {"index": "index_studenti_2shards"}
}

```

200 - OK 58 ms

[Click to send request](#)

[GET \\_cat/shards/index\\_studenti\\_2shards?v](#) [GET \\_cat/shards/index\\_studenti\\_4shards?v](#)

### Exercițiu final:

#### 1. Creează un index `index_studenti_test` cu 5 sharduri.

```
PUT /index_studenti_5shards
```

```
{
  "settings": {
    "number_of_shards": 5,
    "number_of_replicas": 0
  },
  "mappings": {
    "properties": {
      "Nr_matricol": { "type": "integer" },
      "Nume": { "type": "text" },
      "Varsta": { "type": "integer" },
      "Media": { "type": "float" },
      "tara_natala": { "type": "text" }
    }
  }
}
```

#### 2. Generează și inserează 1000+ documente automat.

```
from elasticsearch import Elasticsearch, helpers
from faker import Faker
import random
```

```
fake = Faker()
```

```

# Conectare la Elasticsearch
es = Elasticsearch("http://172.20.10.5:9200", verify_certs=False)

# Verificare conexiune
if not es.ping():
    print("🔴 Conexiunea la Elasticsearch a eșuat.")
    exit()
else:
    print("✅ Conexiunea la Elasticsearch a fost realizată cu succes.")

# Numele indexului
index_name = "index_studenti_5shards"

# Configurare index (dacă nu există deja)
if not es.indices.exists(index=index_name):
    print(f"🔧 Creez indexul '{index_name}'...")
    es.indices.create(
        index=index_name,
        body={
            "settings": {
                "number_of_shards": 4,
                "number_of_replicas": 0
            },
            "mappings": {
                "properties": {
                    "Nr_matricol": { "type": "integer" },
                    "Nume": { "type": "text" },
                    "Varsta": { "type": "integer" },
                    "Media": { "type": "float" },
                    "tara_natala": { "type": "text" }
                }
            }
        }
    )
    print("✅ Indexul a fost creat.")

else:
    print(f"⚠️ Indexul '{index_name}' există deja.")

# Generare documente

```

```

def generate_documents(n=1000):
    for i in range(1, n + 1):
        yield {
            "_index": index_name,
            "_source": {
                "Nr_matricol": i,
                "Nume": fake.name(),
                "Varsta": random.randint(18, 30),
                "Media": round(random.uniform(5.0, 10.0), 2),
                "tara_natala": fake.country()
            }
        }

```

# Inserare în Elasticsearch

try:

```

response = helpers.bulk(es, generate_documents(1000))
print(f" ✅ Inserat {response[0]} documente în indexul '{index_name}'!")
except Exception as e:
    print(f" ❌ Eroare la inserare: {e}")

```

### 3. Măsoară timpul pentru o interogare match\_all.

The screenshot shows the Elasticsearch Dev Tools Console interface. The top navigation bar includes 'elastic', a search bar, and various icons. Below the bar, tabs for 'Console', 'Search Profiler', 'Grok Debugger', and 'Painless Lab' are visible, with 'Console' being the active tab. The main area has tabs for 'History', 'Settings', 'Variables', and 'Help'. A code editor on the left contains a series of Elasticsearch API requests (e.g., `PUT` for creating an index, `POST` for reindexing, and `GET` for searching). On the right, the results of the search request are displayed in a JSON format. A tooltip 'Click to send request' points to the search button. The status bar at the bottom indicates '200 - OK' and '119 ms'.

```

8+     "properties": {
9|       "Nr_matricol": { "type": "integer" },
10|       "Nume": { "type": "text" },
11|       "Varsta": { "type": "integer" },
12|       "Media": { "type": "float" },
13|       "tara_natala": { "type": "text" }
14+
15+
16+
17
18 get index_studenti_2shards/_count
19
20 post _reindex
21{
22   "source": {"index" : "index_studenti_4shards"},
23   "dest": {"index": "index_studenti_2shards"}
24+
25
26
27 GET _cat/shards/index_studenti_5shards?v
28 GET _cat/shards/index_studenti_4shards?v
29
30 get /index_studenti_5shards/_search
31{
32  "query": {
33    "match_all": {}
34+
35}
36

```

```

1 ↴ {
2   "took": 53,
3   "timed_out": false,
4   "shards": {
5     "total": 5,
6     "successful": 5,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": {
12      "value": 1000,
13      "relation": "eq"
14    },
15    "max_score": 1,
16    "hits": [
17      {
18        "_index": "index_studenti_5shards",
19        "_id": "fbItiJVBbaYCYsIkSU4l",
20        "_score": 1,
21        "_source": {
22          "Nr_matricol": 1,
23          "Nume": "Mark Salas",
24          "Varsta": 28,
25          "Media": 5.72,
26          "tara_natala": "Peru"
27        }
28      }
29    ]
30  }
31}
32}
33}
34}
35}
36}

```

### 4. Compară performanța cu un index cu 1 shard.

Fata de un index cu 1 shard, acesta, care are 5 fragmentari, diminueaza timpul de executie.

## -Reaplicarea

Ce este replicarea in Elasticsearch?

In Elasticsearch, replicarea inseamna ca fiecare index poate avea copii de rezerva ale shard-urilor (primary shards).

Acste copii se numesc replica shards si sunt distribuite pe alte noduri ale clusterului pentru:

- Disponibilitate: daca un nod moare, datele sunt accesibile in continuare.
- Performanta: cautarile pot fi servite si de replici, nu doar de shard-ul primar.

Implicit, Elasticsearch creeaza 1 replica pentru fiecare shard.

Comanda:

```
GET /index_cursuri/_settings
```

Explicatie:

- Cauta campul "number\_of\_replicas": "1" pentru a vedea setarea curenta.

### Exercitiu 2: Modificarea numarului de replici

Comanda:

```
PUT /index_cursuri/_settings
```

```
{
```

```
    "number_of_replicas": 2
```

```
}
```

Explicatie:

- Acum fiecare shard are 2 replici pentru si mai multa siguranta.

The screenshot shows the Elasticsearch Dev Tools interface. In the left panel, the code for modifying the index settings is visible:

```
25
26
27 GET _cat/shards/index_studenti_5shards?v
28 GET _cat/shards/index_studenti_4shards?v
29
30 get /index_studenti_5shards/_search
31 {
32     "query": {
33         "match_all": {}
34     }
35 }
36
37 get index_departamente_4sharduri/_mapping
38
39 get _cat/indices?v
40
41 get index_profesori/_settings
42 PUT /index_cursuri/_settings
43 {
44     "number_of_replicas": 2
45 }
46
47
48 PUT /index_cursuri/_settings
49 {
50     "number_of_replicas": 2
51 }
52
53 get /index_cursuri/_settings
```

In the right panel, the response to the last PUT request is displayed:

```
1  {
2     "index_cursuri": {
3         "settings": {
4             "index": {
5                 "routing": {
6                     "allocation": {
7                         "include": {
8                             "_tier_preference": "data_content"
9                         }
10                    }
11                }
12            }
13        }
14    }
15
16    "number_of_shards": "1",
17    "provided_name": "index_cursuri",
18    "creation_date": "1744826058492",
19    "number_of_replicas": "2",
20    "uuid": "cfXBEfLTrIKlw8og-HDQ",
21    "version": {
22        "created": "8500010"
23    }
24
25 }
```

The status bar at the bottom indicates "200 - OK" and "59 ms".

### Exercitiul 3: Observarea replicilor

Comanda:

GET /\_cat/shards/index\_cursuri?v=true

|   |               |       |        |            |      |        |         |            |      | 200 - OK | 134 ms |
|---|---------------|-------|--------|------------|------|--------|---------|------------|------|----------|--------|
| 1 | index         | shard | prirep | state      | docs | store  | dataset | ip         | node |          |        |
| 2 | index_cursuri | 0     | p      | STARTED    | 11   | 27.4kb | 27.4kb  | 172.19.0.2 |      |          |        |
| 3 |               | 0     | r      | UNASSIGNED |      |        |         |            |      |          |        |
| 4 | index_cursuri | 0     | r      | UNASSIGNED |      |        |         |            |      |          |        |
| 5 |               |       |        |            |      |        |         |            |      |          |        |

---

### Exercitiul 4: Test de disponibilitate

Scenariu:

- Daca un nod Elasticsearch cade, o replica este promovata automat ca shard primar.

Ce demonstram:

- Datele raman accesibile datorita replicarii.

---

### Exercitiul 5: Replicare manuala prin reindexare

Comanda:

POST /\_reindex

```
{  
  "source": {  
    "index": "index_cursuri"  
  },  
  "dest": {  
    "index": "index_cursuri_backup"  
  }  
}
```

Explicatie:

- Cream o copie manuala a intregului index index\_cursuri.

---

### Concluzie

- Shard primar** = sursa principala de date.
- Replica** = copie de rezerva activa.
- Replicarea protejeaza datele si ajuta la scalabilitate.
- Reindexarea** = replica la nivel de documente sau index.

## **Secțiunea 8- Noduri**

### **Definiție:**

Un nod este o instanță Elasticsearch care rulează pe o mașină fizică sau virtuală și care face parte dintr-un cluster. Nodurile comunică între ele pentru a împărți datele și pentru a răspunde eficient cererilor utilizatorului.

### **Tipuri de Noduri**

În funcție de rolurile atribuite, nodurile Elasticsearch pot îndeplini una sau mai multe funcții. Cele mai comune tipuri de noduri sunt:

#### **1. Master Node**

Rol principal: Administrarea clusterului.

Responsabilități:

- Gestionarea metadatelor indexurilor și shard-urilor.
- Monitorizarea și organizarea stării clusterului.
- Alegerea și validarea altor noduri master în caz de eșec.

Configurație:

```
yaml
node.master: true
node.data: false
```

#### **2. Data Node**

Rol principal: Stocarea și procesarea datelor.

Responsabilități:

- Găzduirea shard-urilor primare și de replică.
- Executarea operațiunilor de indexare, căutare și agregare.

Configurație:

```
yaml
node.master: false
node.data: true
```

### 3. Ingest Node

Rol principal: Preprocesarea documentelor înainte de indexare.

Responsabilități:

- Aplicarea de pipeline-uri de procesare (e.g. parsare, transformări).

Configurație:

```
yaml
node.ingest: true
```

### 4. Coordinating Node

Rol principal: Intermedierea cererilor.

Responsabilități:

- Preia cererile de la client.
- Le rutează către nodurile relevante.
- Agregă rezultatele și le returnează.

Configurație:

```
yaml
node.master: false
node.data: false
node.ingest: false
```

### 3.5. Machine Learning Node (optional, în X-Pack)

Rol principal: Executarea joburilor de învățare automată.

Responsabilități:

- Detectarea anomalieiilor.
- Analiza modelelor de date.

## Comportamentul în Cluster

- Fiecare cluster are un master node activ (ales automat din nodurile master-eligible).
- Datele sunt împărțite în shard-uri (partiții logice) și replicate pe mai multe noduri.
- În caz de eșec al unui nod, clusterul redistribuie automat shard-urile pentru a menține disponibilitatea.

## Exemple de Configurații

- Nod cu roluri multiple (default)

```
yaml
node.name: node-1
node.master: true
node.data: true
node.ingest: true
```

- Nod dedicat doar ca Master

```
yaml
node.name: master-node
node.master: true
node.data: false
node.ingest: false
```

## Bună Practică în Producție

- Se recomandă separarea rolurilor pe noduri dedicate în sistemele de producție.
- Este ideal să existe un număr impar de noduri master-eligible (de exemplu, 3) pentru a evita blocajele în procesul de alegere a masterului.
- Se recomandă monitorizarea constantă a nodurilor și configurarea alertelor pentru eșecuri.

## Concluzie

Nodurile sunt componente esențiale ale infrastructurii Elasticsearch. Înțelegerea tipurilor și rolurilor acestora este crucială pentru proiectarea, configurarea și întreținerea unui cluster robust și scalabil. Distribuirea eficientă a responsabilităților între noduri contribuie direct la performanța și reziliența sistemului.

## Sectiunea 8-Dashboarduri

### Ce este un Dashboard în Kibana?

Un dashboard în Kibana este o colecție de grafice, tabele și alte vizualizări organizate într-o singură pagină pentru a analiza rapid datele din Elasticsearch. Dashboard-urile permit o interacțiune dinamică prin filtre, căutări și selecții.

### De ce folosim Dashboard-uri?

- Pentru a vizualiza tendințe și modele în date.
- Pentru a monitoriza în timp real anumite metriki sau activități.
- Pentru a crea rapoarte vizuale rapide și intuitive.

### Grafice în Kibana (în general)

Kibana oferă mai multe tipuri de grafice:

1. **Bar Chart (grafic de bare)**
  - Similar cu histograma, dar mai flexibil (poți pune pe axa X categorii, nu doar intervale).
  - Exemplu: vânzări pe județ.
2. **Line Chart (grafic liniar)**
  - Ideal pentru analiza seriilor temporale.
  - Exemplu: evoluția traficului pe site în ultimele 30 de zile.
3. **Pie Chart (diagramă circulară)**
  - Bun pentru împărțiri procentuale.
  - Exemplu: proporția vânzărilor pe categorii de produse.
4. **Heatmap**
  - Reprezintă valorile în două dimensiuni prin culoare.
  - Exemplu: activitatea pe ore și zile.
5. **Data Table**
  - Tabel interactiv derivat din datele vizualizate.

### Ghid Pas cu Pas: Crearea unui Dashboard Simplu

1. Accesează interfața Kibana (de obicei <http://localhost:5601>).
2. Verifică datele disponibile în secțiunea 'Discover'.
3. Creează un Index Pattern:

- Meniu: 'Stack Management' -> 'Kibana' -> 'Index Patterns'.
- Creează un pattern nou pentru indexul dorit (ex: index\_cursuri\*).

The screenshot shows the 'data\_view\_cursuri' index pattern configuration. It lists 15 fields: AN\_studiu, An\_studiu, Denumire, Denumire.keyword, Nr\_credite, and others. The interface includes tabs for Fields, Scripted fields, Field filters, and Relationships. There are buttons for Delete, Set as default, and Edit.

#### 4. Creează vizualizări:

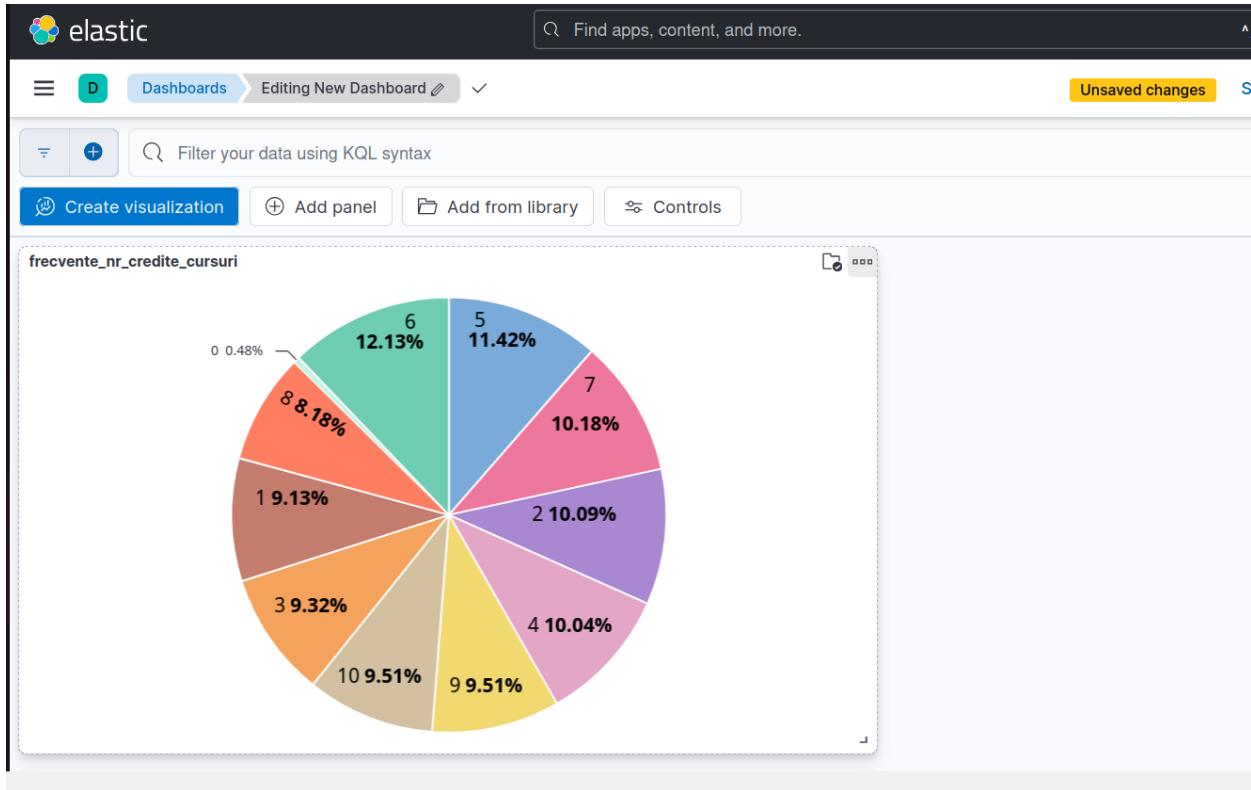
- Meniu: 'Visualize Library' -> 'Create visualization'.
- Selectează tipul dorit (ex: bar chart, pie chart).
- Alege indexul potrivit și setează câmpurile.

The screenshot shows a pie chart visualization titled 'Pie' for the index pattern 'data\_view\_cursuri'. The chart represents the distribution of 'Nr\_credite' values. The data is as follows:

| Categorie (Nr_credite) | Percenție (%) |
|------------------------|---------------|
| 6                      | 12.13%        |
| 5                      | 11.42%        |
| 7                      | 10.18%        |
| 2                      | 10.09%        |
| 4                      | 10.04%        |
| 9                      | 9.51%         |
| 10                     | 9.51%         |
| 3                      | 3.93%         |
| 1                      | 1.91%         |
| 8                      | 8.18%         |
| 0                      | 0.48%         |

#### 5. Creează Dashboard-ul:

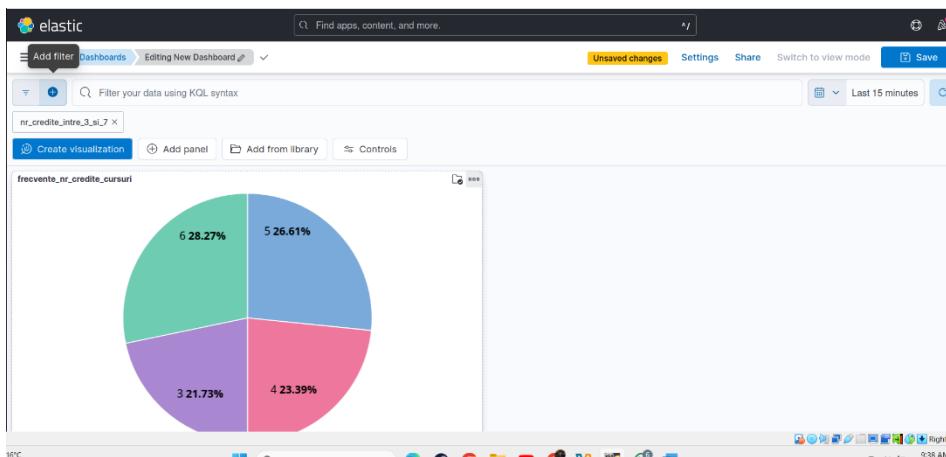
- Meniu: 'Dashboard' -> 'Create new dashboard'.
- Adaugă vizualizările create anterior.
- Salvează dashboard-ul.



## 6. Interactivitate:

- Adaugă filtre și intervale de timp.

- Folosește filtrele



Deoarece nu avem niciun camp de tip de data, am creat un nou index, folosind script-ul:

---

```
from elasticsearch import Elasticsearch, helpers
from faker import Faker
import random

fake = Faker()

# Conectare la Elasticsearch
es = Elasticsearch("http://172.20.10.5:9200", verify_certs=False)

# Verificare conexiune
if not es.ping():
    print("🔴 Conexiunea la Elasticsearch a eşuat.")
    exit()
else:
    print("🟢 Conectat la Elasticsearch cu succes.")

# Numele indexului actualizat
index_name = "index_profesori_data"

# Configurare index (cu 4 sharduri și 0 replici + camp data_angajare)
if not es.indices.exists(index=index_name):
    print(f'🔧 Creez indexul "{index_name}"')
    es.indices.create(
        index=index_name,
        body={
            "settings": {
                "number_of_shards": 4,
                "number_of_replicas": 0
            },
            "mappings": {
                "properties": {
                    "Id_profesor": { "type": "integer" },
                    "Id_prefesor": { "type": "long" },
                    "Nume": { "type": "text" },
                    "Birou": { "type": "integer" },
                    "data_angajare": { "type": "date" }
                }
            }
        }
    )

```

```

        }
    }
}

)
print("✓ Indexul a fost creat.")

else:
    print(f"✗ Indexul '{index_name}' există deja.")

# Generare documente
def generate_documents(n=1000):
    for i in range(1, n + 1):
        yield {
            "_index": index_name,
            "_source": {
                "Id_profesor": i,
                "Id_prefesor": random.randint(1_000_000, 9_999_999),
                "Nume": fake.name(),
                "Birou": random.randint(1, 100),
                "data_angajare": fake.date_between(start_date="-10y",
end_date="today").isoformat()
            }
        }

# Inserare documente în Elasticsearch
try:
    response = helpers.bulk(es, generate_documents(1000))
    print(f"✓ Inserat {response[0]} documente în indexul '{index_name}'")
except Exception as e:
    print(f"✗ Eroare la inserare: {e}")

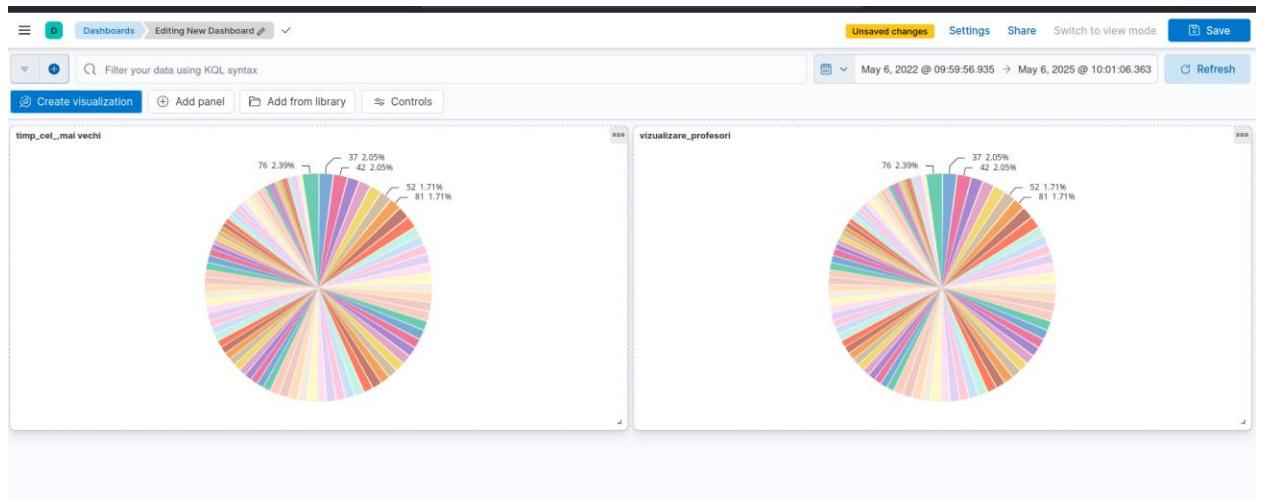
```

---

Am creat un nou index\_pattern pentru index\_profesori.

The screenshot shows the 'Create data view' page in Kibana. On the left sidebar, under 'Data Views', 'Data View' is selected. The main area is titled 'Create data view' with fields for 'Name' (data\_view\_profesori), 'Index pattern' (index\_profesori\*), and 'Timestamp field' (data\_angajare). A note says 'Select a timestamp field for use with the global time filter.' Below these are 'Show advanced settings' and 'Save data view to Kibana' buttons. To the right, a panel titled 'Your index pattern matches 6 sources.' lists 'All sources' and 'Matching sources'. The matching sources are: index\_profesori, index\_profesori2, index\_profesori3, index\_profesori\_4sharduri, index\_profesori\_data, and index\_profesori\_grad, each with an 'Index' button. At the bottom right is a toolbar with various icons.

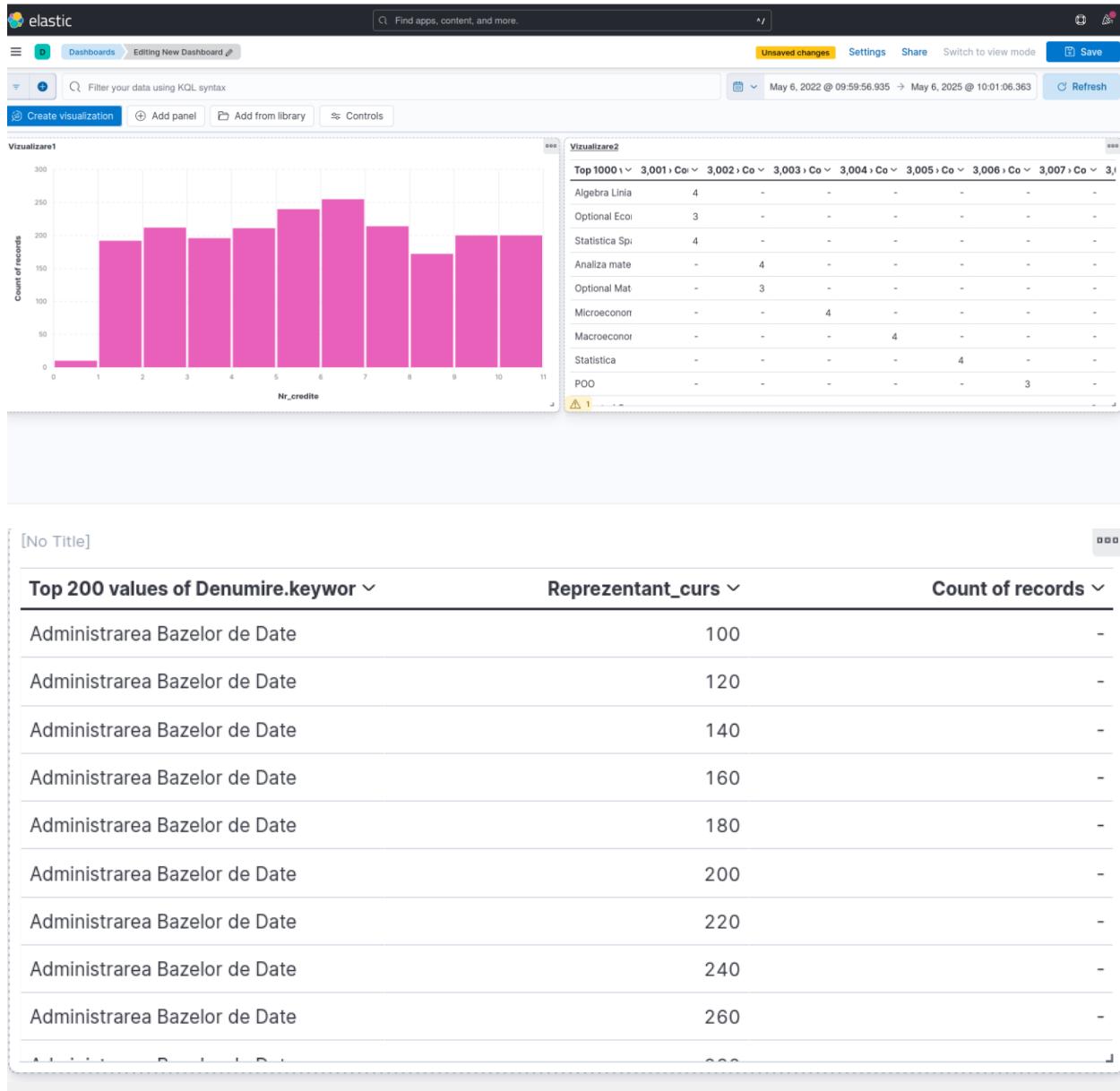
Dashboard cu filtre de timp



Exerciții de bază:

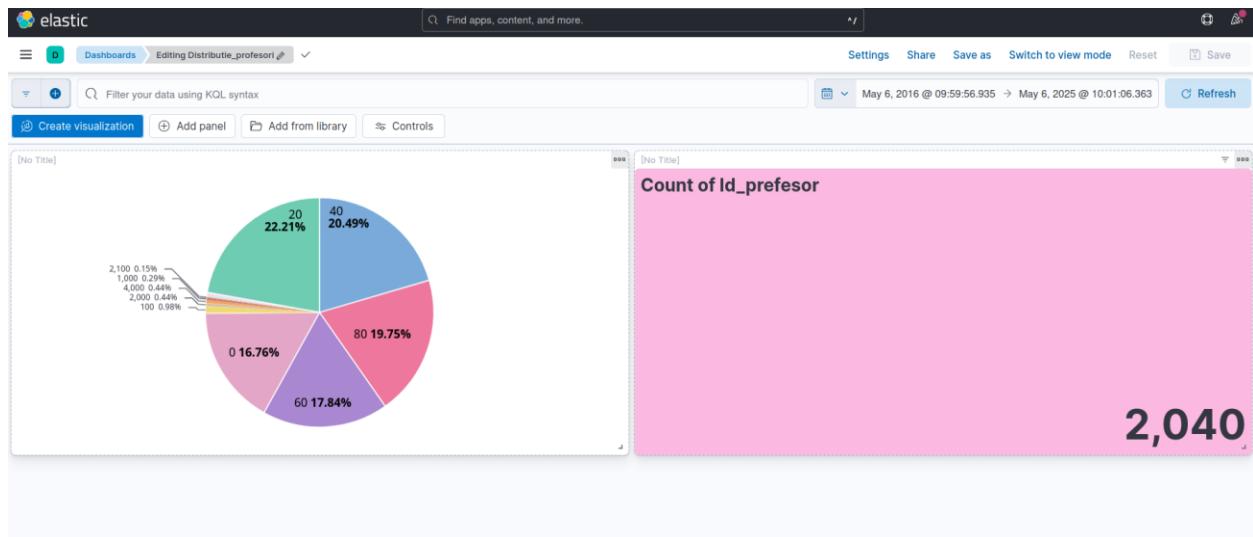
1. Creează un dashboard numit 'Analiză Cursuri'.

- Vizualizare 1: Histogramă cu distribuția numărului de credite.
- Vizualizare 2: Tabel cu toate cursurile și reprezentanții lor.



## 2. Creează un dashboard numit 'Distribuție Profesori'.

- Vizualizare 1: Pie chart cu distribuția profesorilor pe birouri.
- Vizualizare 2: Metrică ce afișează numărul total de profesori.



3. Creează un dashboard numit 'Departamente și Birouri'.

- Vizualizare 1: Bar chart cu numărul de departamente după criteriu birou.

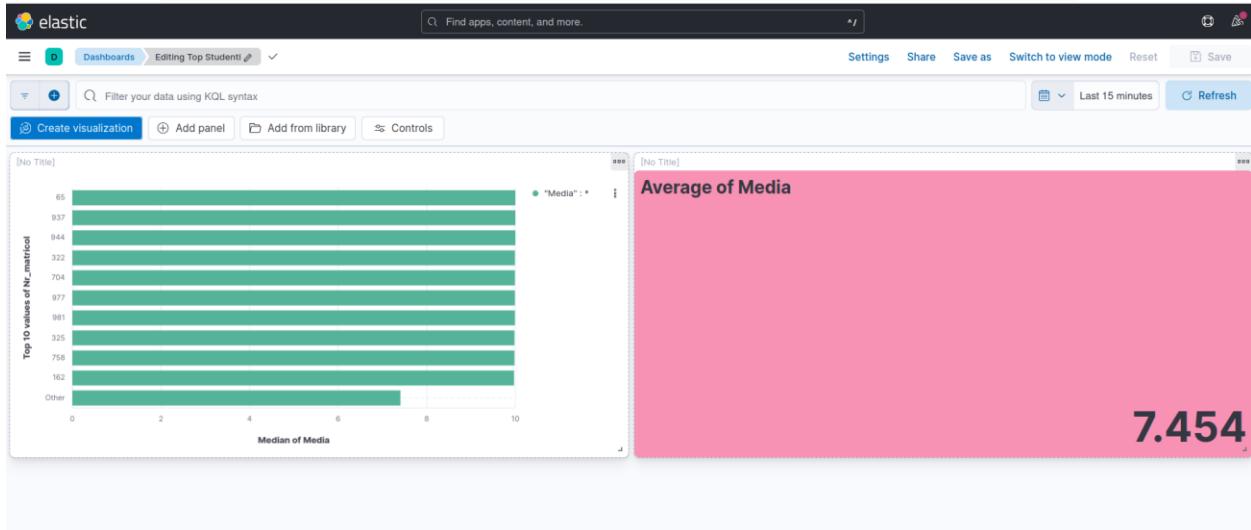
- Vizualizare 2: Data table cu toate departamentele și ID-urile lor.

| interval_birou | Id_Dep | 1 > Count of records | Other > Count of records |
|----------------|--------|----------------------|--------------------------|
| 100            | 0      | -                    | -                        |
| 100            | 50     | -                    | -                        |
| 100            | 100    | -                    | -                        |
| 100            | 150    | -                    | -                        |
| 100            | 200    | -                    | -                        |
| 100            | 250    | -                    | -                        |
| 100            | 300    | -                    | -                        |
| 100            | 350    | -                    | -                        |
| 100            | 400    | -                    | -                        |
| 100            | 450    | -                    | -                        |

Exerciții avansate:

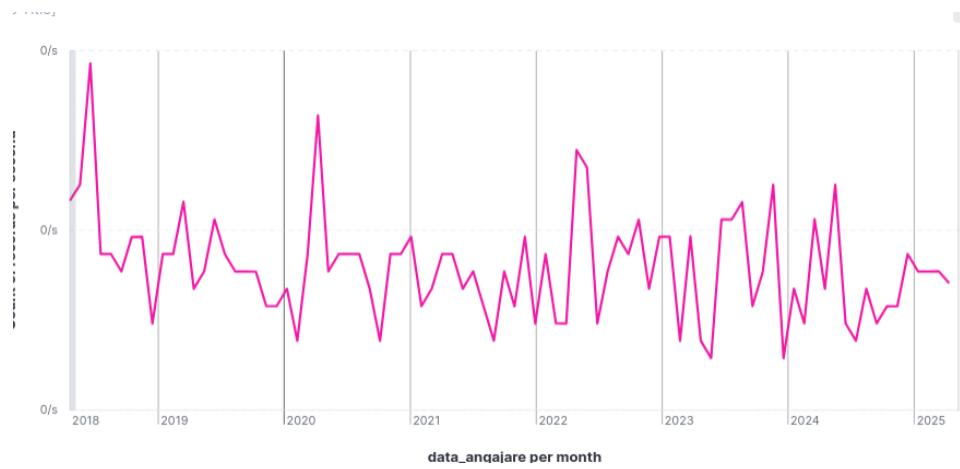
4. Dashboard 'Top Studenți':

- Creează un nou index pattern dacă ai date pentru studenți.
- Vizualizare: Top 10 studenți după media generală (bar chart).
- Vizualizare: Metrică cu media generală a întregului grup.



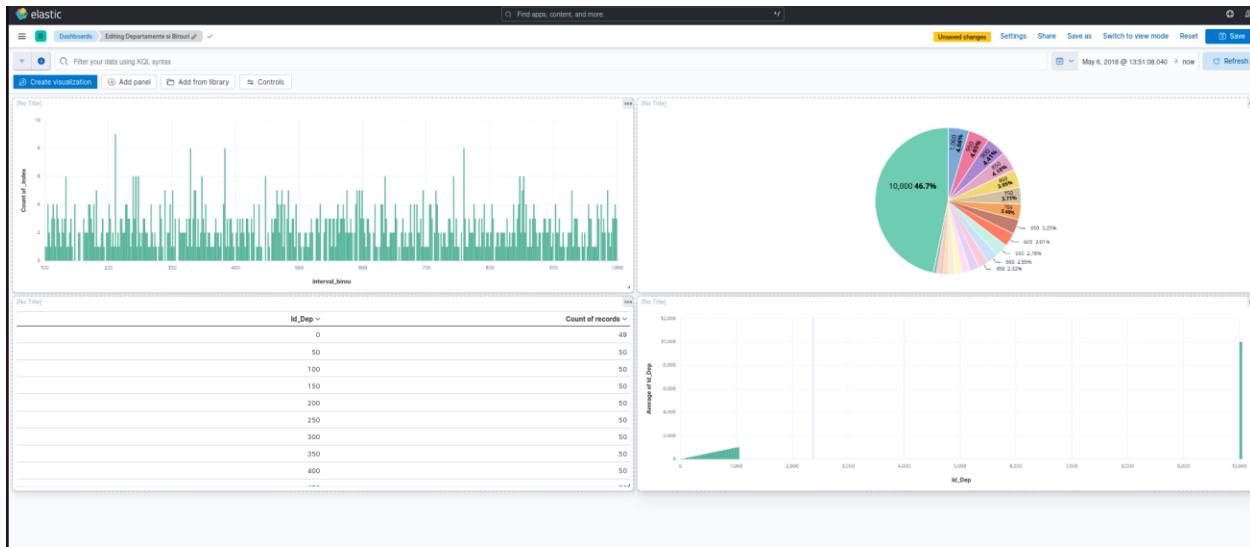
## 5. Dashboard 'Evoluție în timp':

- Creează o vizualizare de tip line chart care să urmărească numărul de înscrieri/cursuri în timp (dacă există date cu timestamp).
- Folosește 'date histogram' pentru axa X.



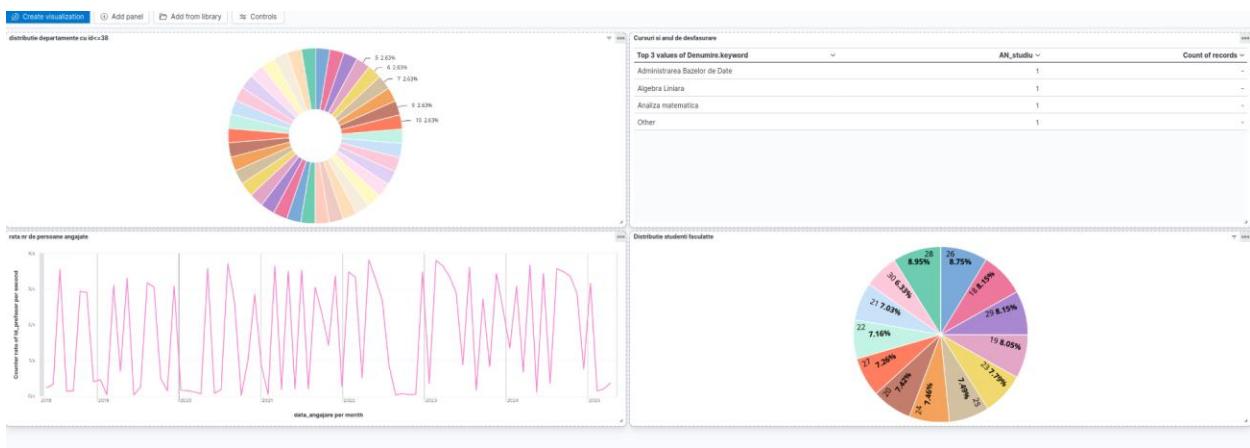
## 6. Dashboard 'Comparare Departamente':

- Pie chart: procentajul fiecărui departament în totalul structurii.
- Table: numărul de profesori pe departament.
- Bar chart: media numărului de cursuri asociate fiecărui departament.



## Temă suplimentară:

- Creează un dashboard complet personalizat combinând date din toate indexurile disponibile.
- Include minim 5 vizualizări diferite.
- Aplică filtre dinamice și experimentează cu Time Picker-ul.



## Secțiunea 9- Audit

```
PUT /audit_loguri
{
  "mappings": {
```

```
"properties": {  
    "Data conectare": {  
        "type": "date"  
    },  
    "Message": {  
        "type": "keyword"  
    },  
    "event dataset": {  
        "type": "keyword"  
    },  
    "nume nod": {  
        "type": "keyword"  
    }  
}  
}
```

Am scris log-urile intr-un fisier de tip text, iar mai apoi folosind un script am inserat datele selectate în index.

- Pentru a vedea logurile în elasticsearch am folosit sudo docker compose elasticsearch

Pentru această comandă avem și opțiunea cu **-f** (follow), care arată logurile în timp real.

Un aspect important este că la pornirea docker-ului acesta nu avea niciun output trimis, astfel rezultatul instrucțiunii era gol, dar și pentru că la început conteiner-ul u scrie nimic relevant în stdout/stderr. Așadar, tocmai după crearea indexului și inserarea documentelor în acesta, comanda a oferit un rezultat vizibil.

- Inserare în fișier: sudo docker logs elasticsearch > elasticsearch\_logs.txt
  - Script: from elasticsearch import Elasticsearch, helpers

```
from datetime import datetime
```

```
import json

# 🔒 Conectare la Elasticsearch
es = Elasticsearch("http://localhost:9200", verify_certs=False)

# ✅ Verificare conexiune
if not es.ping():
    print("❌ Conexiunea la Elasticsearch a eșuat.")
    exit()
else:
    print("✅ Conectat la Elasticsearch cu succes.")

# 💡 Numele indexului
index_name = "audit_loguri"

# 🛠️ Creare index dacă nu există
if not es.indices.exists(index=index_name):
    print(f"🔧 Creez indexul '{index_name}'...")
    es.indices.create(
        index=index_name,
        body={
            "mappings": {
                "properties": {
                    "Data conectare": { "type": "date" },
                    "Message": { "type": "keyword" },
                    "event dataset": { "type": "keyword" },
                }
            }
        }
    )
```

```
        "nume nod": { "type": "keyword" }

    }

}

)

print(" ✅ Indexul a fost creat.")

else:
```

```
    print(f"❗ Indexul '{index_name}' există deja.")
```

```
# 📁 Pregătire documente din fișierul log
```

```
actions = []
```

```
with open("elasticsearch_logs.txt", "r") as f:
```

```
    for line in f:
```

```
        try:
```

```
            log = json.loads(line)
```

```
            timestamp = log.get("@timestamp")
```

```
            if not timestamp:
```

```
                continue # dacă lipsește data, ignorăm linia
```

```
# Convertim @timestamp în format calendaristic (YYYY-MM-DD)
```

```
            try:
```

```
                data_conecțare = datetime.strptime(timestamp, "%Y-%m-%dT%H:%M:%S.%fZ").date().isoformat()
```

```
            except ValueError:
```

```
                # fallback dacă lipsesc microsecunde
```

```
    data_conecțare = datetime.strptime(timestamp, "%Y-%m-%dT%H:%M:%SZ").date().isoformat()
```

```
doc = {  
    "_index": index_name,  
    "_source": {  
        "Data conețtere": data_conecțare,  
        "Message": log.get("message", ""),  
        "event dataset": log.get("event.dataset", ""),  
        "nume nod": log.get("elasticsearch.node.name", "")  
    }  
}
```

```
actions.append(doc)
```

```
except json.JSONDecodeError:  
    print("⚠️ Linie invalidă JSON, ignor...")
```

```
# 🚀 Inserare în Elasticsearch
```

```
if actions:  
    helpers.bulk(es, actions)  
    print(f"✅ {len(actions)} documente au fost încărcate în indexul '{index_name}'.")  
else:  
    print("⚠️ Niciun document valid nu a fost găsit în fișier.")
```

- 
- Apoi am creat un nou Data View care să aibă ca index pattern : audit\_loguri.

**Create data view**

Name: audit\_loguri

Index pattern: audit\*

Timestamp field: Data conectare

Show advanced settings

Your index pattern matches 1 source.

All sources Matching sources

audit\_loguri Index

Rows per page: 10

X Close Save data view to Kibana

Right Ctrl

Folosind discover, putem selecta datele după diferite criterii.

elastic

Discover

audit\_loguri Message : "closed"

Find apps, content, and more.

New Open Share Alerts Inspect Save

May 8, 2022 @ 09:49:39.295 → now Refresh

2 hits

Break down by: Select field

May 8, 2022 @ 09:49:39.295 - May 8, 2025 @ 10:01:25.394 (Interval: Auto - month)

Available fields: Data conectare, event dataset, Message, nume nod

Empty fields: 0

Meta fields: 3

Documents Field statistics

Get the best look at your search results

Take the tour Dismiss

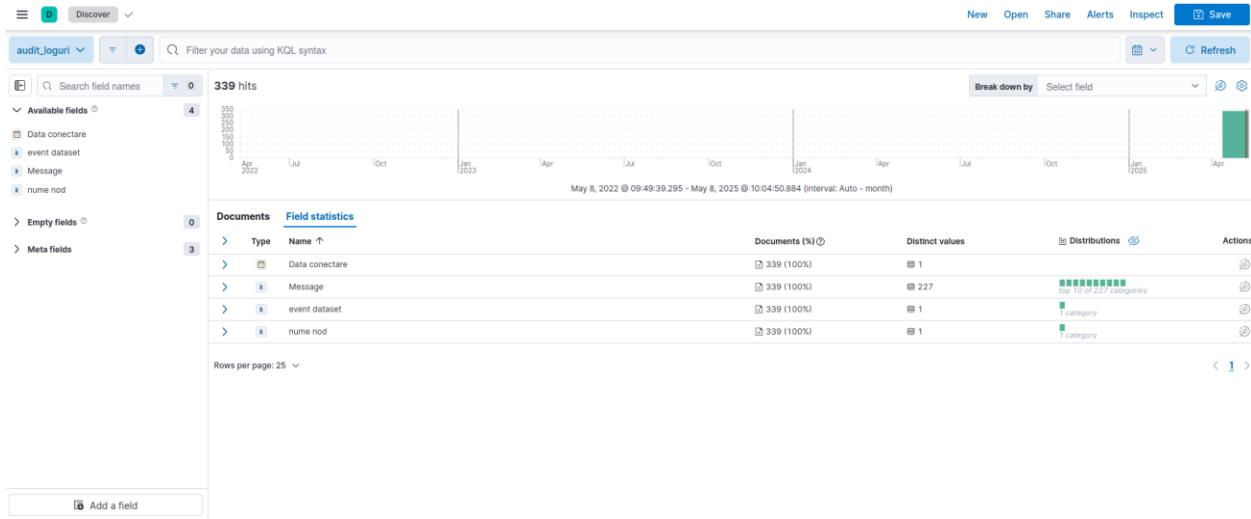
Document

Apr 22, 2025 @ 03:00:00.000 Message closed Data conectare Apr 22, 2025 @ 03:00:00.000 event dataset elasticsearch.server nume nod 8ff743bcdf9d \_id ZbCdrpYB4\_eT9vogkIDY \_index audit\_loguri \_score -

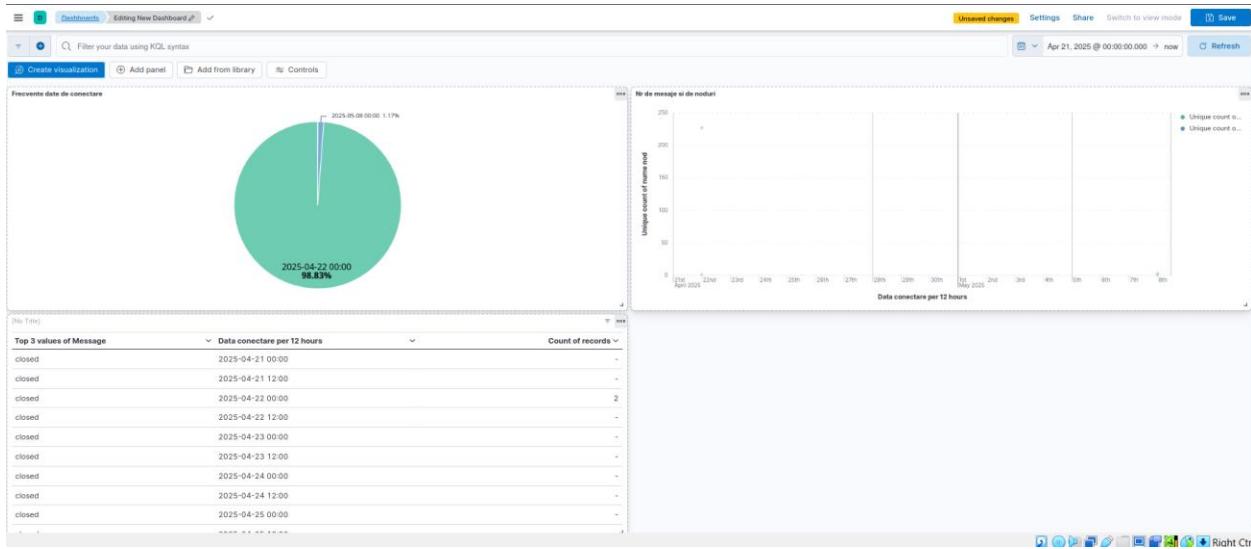
Apr 22, 2025 @ 03:00:00.000 Message closed Data conectare Apr 22, 2025 @ 03:00:00.000 event dataset elasticsearch.server nume nod 8ff743bcdf9d \_id 6bCdrpYB4\_eT9vogkX \_index audit\_loguri \_score -

Add a field

În imaginea de mai sus, am ales să cau log-urile cu mesajul "closed". Ca și output, avem 2 loguri, ambele fiind în luna aprilie, lucrul ce se poate observa și în graficul de mai sus. Iar în imaginea de mai jos, putem vedea mini statisticile.



Mai departe, am ales să fac un dashboard pentru a putea vizualiza mai bine conținutul indexului.



La final, le-am mutat în volume, pentru a nu le pierde

version: '3.8'

services:

elasticsearch:

image: docker.elastic.co/elasticsearch/elasticsearch:8.12.2

container\_name: elasticsearch2

environment:

- network.host=0.0.0.0
- xpack.security.http.ssl.enabled=false
- xpack.security.enrollment.enabled=false
- discovery.type=single-node
- path.repo=/usr/share/elasticsearch/backup
- xpack.security.enabled=false
- bootstrap.memory\_lock=true
- ES\_JAVA\_OPTS=-Xms512m -Xmx512m
- xpack.security.transport.ssl.enabled=false

ports:

- "9200:9200"
- "9300:9300"

volumes:

- esdata:/usr/share/elasticsearch/data
- /home/es\_backup:/usr/share/elasticsearch/backup
- eslogs:/usr/share/elasticsearch/logs #  nou

networks:

- elasticnet

ulimits:

memlock:

- soft: -1

- hard: -1

kibana:

- image: docker.elastic.co/kibana/kibana:8.12.2
- container\_name: kibana

environment:

```
- ELASTICSEARCH_HOSTS=http://elasticsearch2:9200
```

```
- xpack.security.enabled=false
```

ports:

```
- "5601:5601"
```

networks:

```
- elasticnet
```

depends\_on:

```
- elasticsearch
```

volumes:

esdata:

```
external: true
```

```
name: cristina-scarlat_esdata
```

```
eslogs: #  definit volumul pentru loguri
```

```
name: cristina-scarlat_eslogs
```

networks:

elasticnet:

```
driver: bridge
```

## Secțiunea 10- Testare Securitate

Pentru început am simulat erori de securitate:

-am generat diferite situații și apoi le am pus într-un fișier txt- loguri\_suspecte.txt

```
{"@timestamp":"2025-05-08T10:01:12.000Z","log.level":"WARN","message":"failed login for user admin from 192.168.1.10","event.dataset":"auth.failure","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:01:15.000Z","log.level":"WARN","message":"failed login for user admin from 192.168.1.10","event.dataset":"auth.failure","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:01:18.000Z","log.level":"WARN","message":"failed login for user admin from 192.168.1.10","event.dataset":"auth.failure","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:02:10.000Z","log.level":"ERROR","message":"unauthorized access attempt to .security index by user guest","event.dataset":"access.control","elasticsearch.node.name":"sim02"}
```

```
{"@timestamp":"2025-05-08T10:03:00.000Z","log.level":"ERROR","message":"unauthorized DELETE request to index '.kibana' by user 'external_user'","event.dataset":"api.security","elasticsearch.node.name":"sim02"}
```

```
{"@timestamp":"2025-05-08T10:03:50.000Z","log.level":"INFO","message":"successful login for user admin","event.dataset":"auth.success","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:04:30.000Z","log.level":"WARN","message":"possible SQL injection detected: SELECT * FROM users WHERE username = 'admin' --","event.dataset":"web.requests","elasticsearch.node.name":"web01"}
```

```
{"@timestamp":"2025-05-08T10:05:10.000Z","log.level":"WARN","message":"XSS attempt detected in input: <script>alert('xss')</script>","event.dataset":"web.requests","elasticsearch.node.name":"web01"}
```

```
{"@timestamp":"2025-05-08T10:06:00.000Z","log.level":"INFO","message":"user 'ana' accessed index 'audit_loguri'","event.dataset":"access.ok","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:06:45.000Z","log.level":"ERROR","message":"user 'test' attempted to delete document from index 'audit_loguri' without permission","event.dataset":"api.security","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:07:30.000Z","log.level":"WARN","message":"failed login for user test from 192.168.1.100","event.dataset":"auth.failure","elasticsearch.node.name":"sim01"}
```

```
{"@timestamp":"2025-05-08T10:08:12.000Z","log.level":"INFO","message":"user 'root' connected to cluster","event.dataset":"connection","elasticsearch.node.name":"sim02"}
```

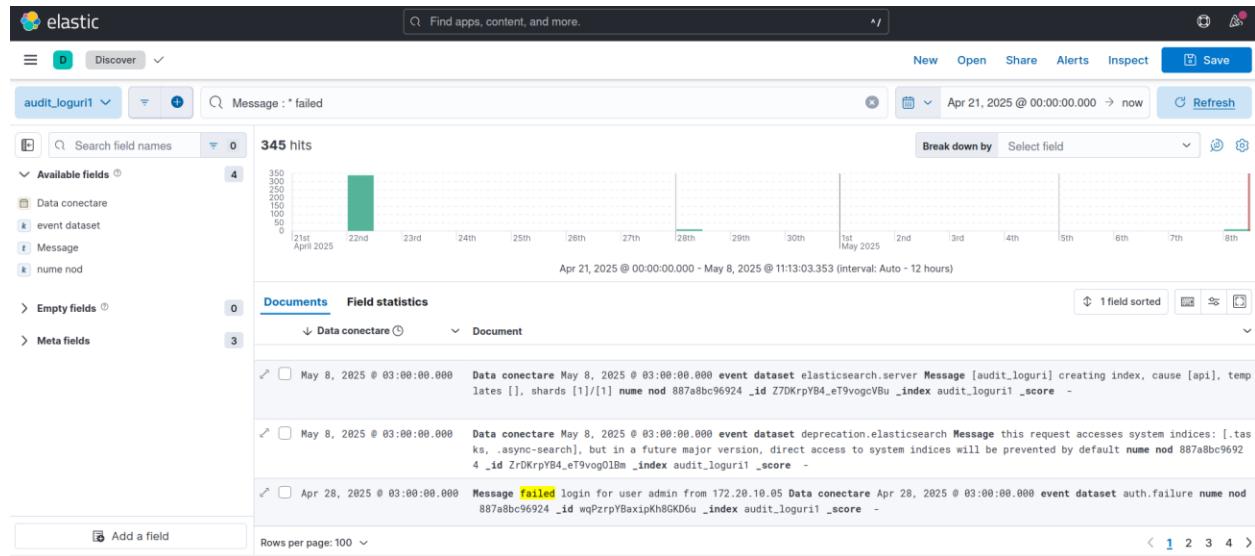
```

{@timestamp:"2025-05-08T10:08:45.000Z","log.level":"ERROR","message":"unauthorized
access attempt to .tasks index by user
hacker","event.dataset":"access.control","elasticsearch.node.name":"sim02"}

{@timestamp:"2025-05-08T10:09:10.000Z","log.level":"WARN","message":"failed login for
user guest from
192.168.1.150","event.dataset":"auth.failure","elasticsearch.node.name":"sim01"}

{@timestamp:"2025-05-08T10:10:00.000Z","log.level":"INFO","message":"index
'audit_loguri' created by
system","event.dataset":"elasticsearch.server","elasticsearch.node.name":"sim-core"}

```



200 -OK 86 ms

```
History Settings Variables Help
eleged to delete document from index _source/_loguri without permission , event
dataset:"api.timestamp","elasticsearch.node.name":"sim01"
230 {"index":[]}
231 {"@timestamp": "2025-05-08T10:07:30.000Z", "log.level": "WARN", "message": "failed login for
user test from 192.168.1.100", "event.dataset": "auth.failure", "elasticsearch.node.name":
"sim01"}
232 {"index":[]}
233 {"@timestamp": "2025-05-08T10:08:12.000Z", "log.level": "INFO", "message": "user 'root'
connected to cluster", "event.dataset": "connection", "elasticsearch.node.name": "sim02" }
234 {"index":[]}
235 {"@timestamp": "2025-05-08T10:08:45.000Z", "log.level": "ERROR", "message": "unauthorized
access attempt to .tasks index by user 'hacker'", "event.dataset": "access.control"
"elasticsearch.node.name": "sim02" }
236 {"index":[]}
237 {"@timestamp": "2025-05-08T10:09:10.000Z", "log.level": "WARN", "message": "failed login for
user guest from 192.168.1.150", "event.dataset": "auth.failure", "elasticsearch.node
.name": "sim01" }
238 {"index":[]}
239 {"@timestamp": "2025-05-08T10:10:00.000Z", "log.level": "INFO", "message": "index
'audit_loguri' created by system", "event.dataset": "elasticsearch.server"
"elasticsearch.node.name": "sim-core" }

240
241
242 GET /audit_loguri/_search
243 {
244-   "query": {
245-     "match": {
246-       | "log.level": "WARN"
247-     }
248-   }
249- }
250

  -> 🔍
```

## **Secțiunea 11- Analiza calității vietii în Europa**

Pentru început am creat indecsii aferenti datelor.

```
1  PUT /life_satisfaction
2  {
3    "mappings": {
4      "properties": {
5        "country": { "type": "keyword" },
6        "satisfaction_score": { "type": "float" }
7      }
8    }
9  }
10 }
11
12 PUT /education_level
13 {
14   "mappings": {
15     "properties": {
16       "country": { "type": "keyword" },
17       "percent_with_education": { "type": "float" }
18     }
19   }
20 }
21
22 PUT /self_perceived_health
23 {
24   "mappings": {
25     "properties": {
26       "country_code": { "type": "keyword" },
27       "country": { "type": "keyword" },
28       "value": { "type": "float" },
29       "year": { "type": "integer" },
30       "Indicator": { "type": "keyword" }
31     }
32   }
33 }
34
35 PUT /income
36 {
37   "mappings": {
38     "properties": {
39       "country_code": { "type": "keyword" },
40       "country": { "type": "keyword" }
41     }
42   }
43 }
```

## **Secțiunea 13- Analiza exploratorie a datelor**

Pentru a înțelege și mai bine rolul acestor platforme în analiza exploratorie a datelor, am ales să studiez disparitățile calității vieții la nivel european în anul 2024.

- Am început prin a crea indecșii în care am inserat date din fișiere csv, colectate de pe Eurostat.

PUT /life\_satisfaction

```
{  
  "mappings": {  
    "properties": {  
      "country": { "type": "keyword" },  
      "satisfaction_score": { "type": "float" }  
    }  
  }  
}
```

Descriere:

Acest index stochează scorul mediu de satisfacție a vieții per țară. Este util pentru a evalua starea subiectivă a populației și poate fi corelat cu venitul, educația sau sănătatea percepță.

Index: education\_level

Fișier sursă: %perscucelputinunniveldeeduc.csv

Comanda PUT pentru crearea indexului:

```
PUT /education_level
{
  "mappings": {
    "properties": {
      "country": { "type": "keyword" },
      "percent_with_education": { "type": "float" }
    }
  }
}
```

Descriere:

Acest index reflectă procentul populației cu cel puțin un nivel de educație. Se pot realiza analize comparative între țări și se poate corela cu satisfacția vieții sau venitul.

Index: self\_perceived\_health

Fișier sursă: self\_perceived\_health\_good.csv

Comanda PUT pentru crearea indexului:

```
PUT /self_perceived_health
{
  "mappings": {
    "properties": {
      "country_code": { "type": "keyword" },
      "country": { "type": "keyword" },
      "value": { "type": "float" },
      "year": { "type": "integer" },
      "indicator": { "type": "keyword" }
    }
  }
}
```

```
    }  
}  
}
```

Descriere:

Acum indexul conține date despre percepția populației asupra propriei stări de sănătate. Va necesita preprocesare pentru a extrage și normaliza valorile. Poate fi folosit pentru a evidenția corelații între sănătate, educație și venituri.

Index: income

Fișier sursă: income.csv

Comanda PUT pentru crearea indexului:

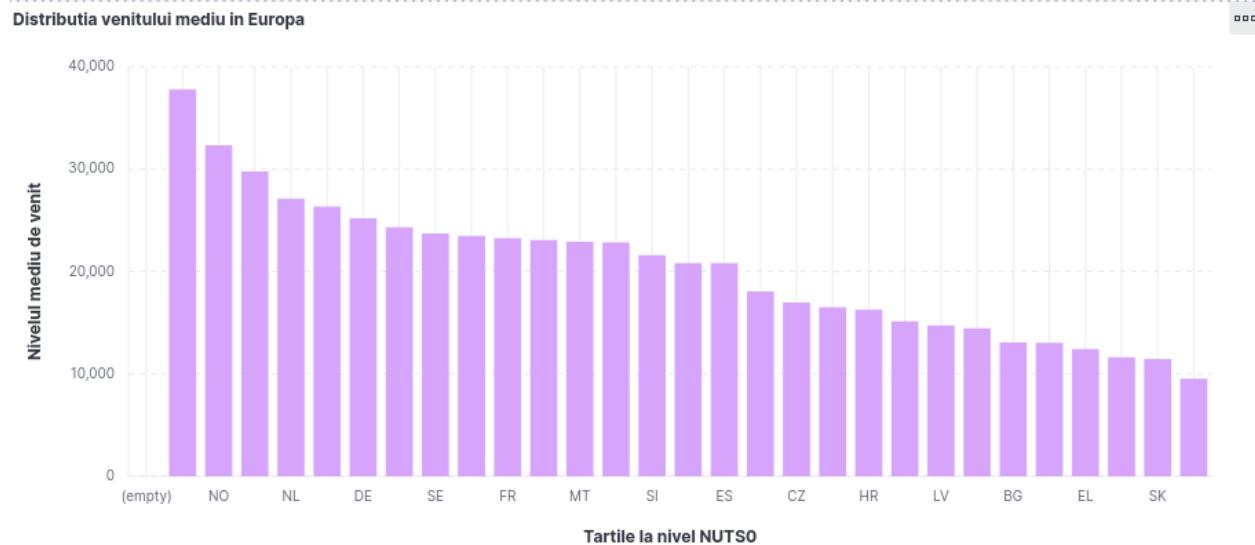
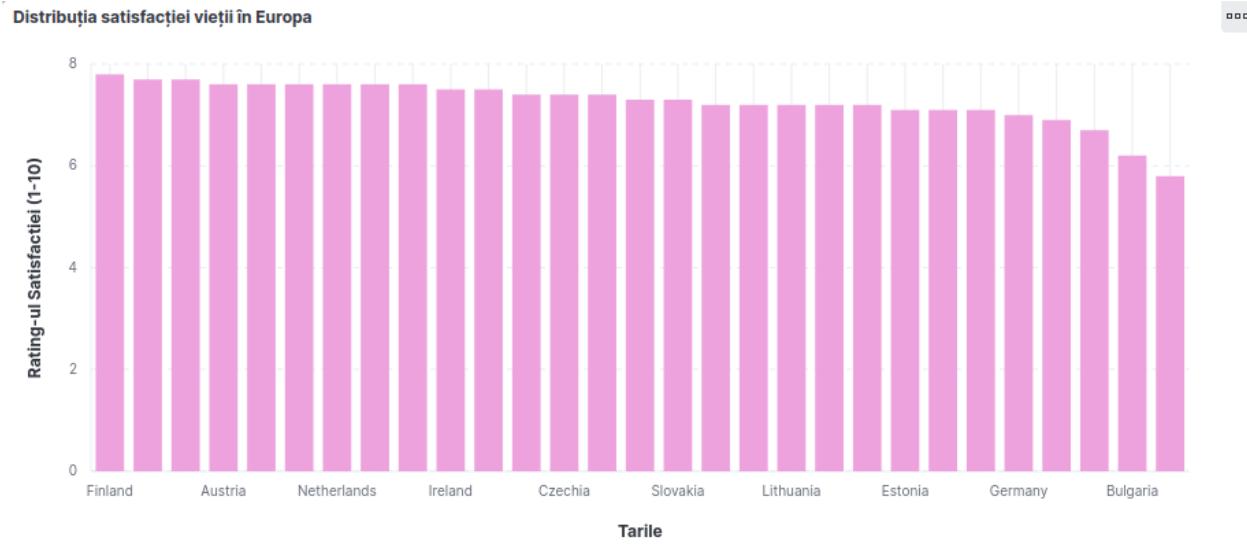
PUT /income

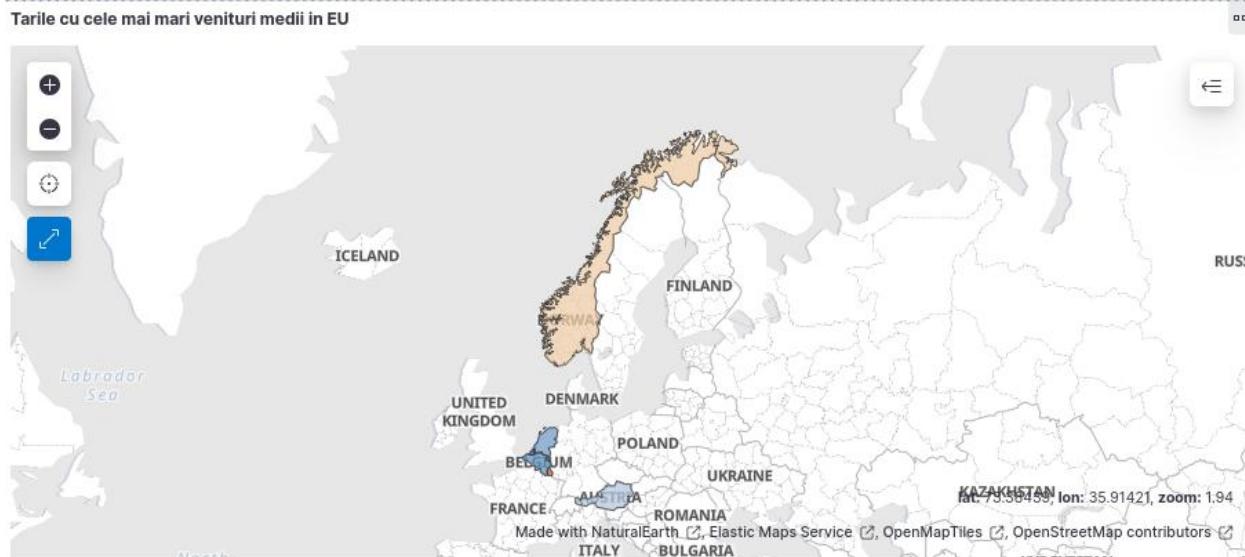
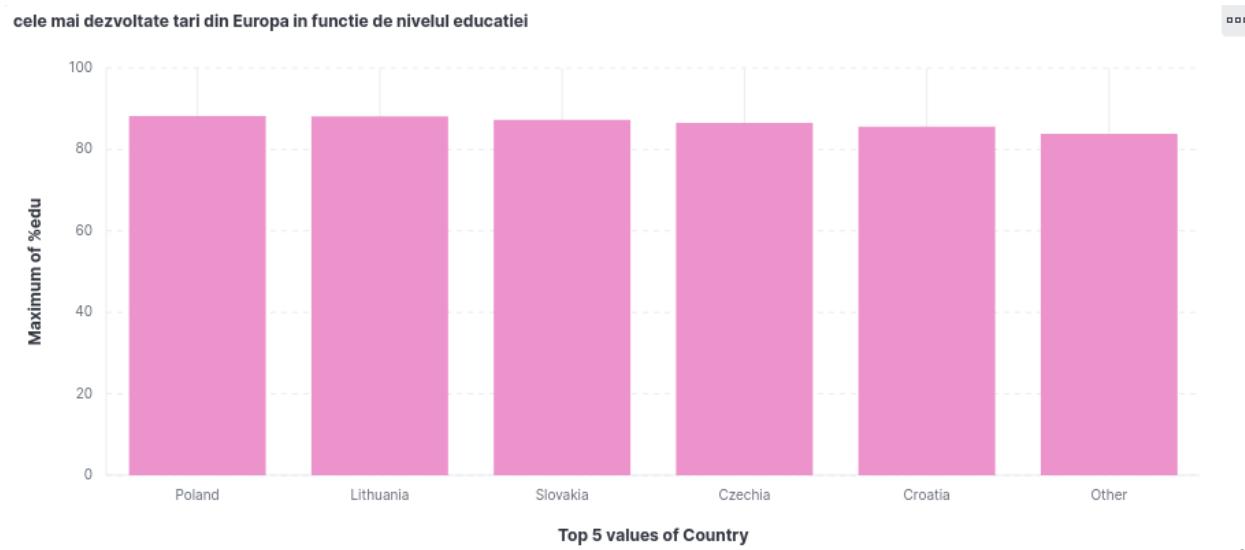
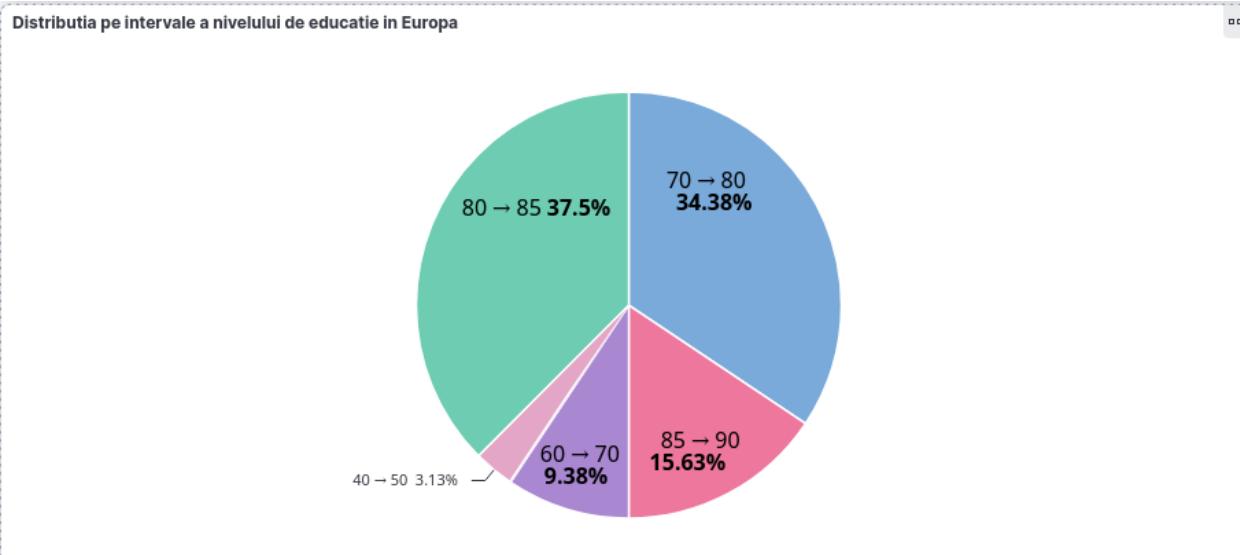
```
{  
  "mappings": {  
    "properties": {  
      "country_code": { "type": "keyword" },  
      "country": { "type": "keyword" },  
      "income_eur": { "type": "float" }  
    }  
  }  
}
```

Descriere:

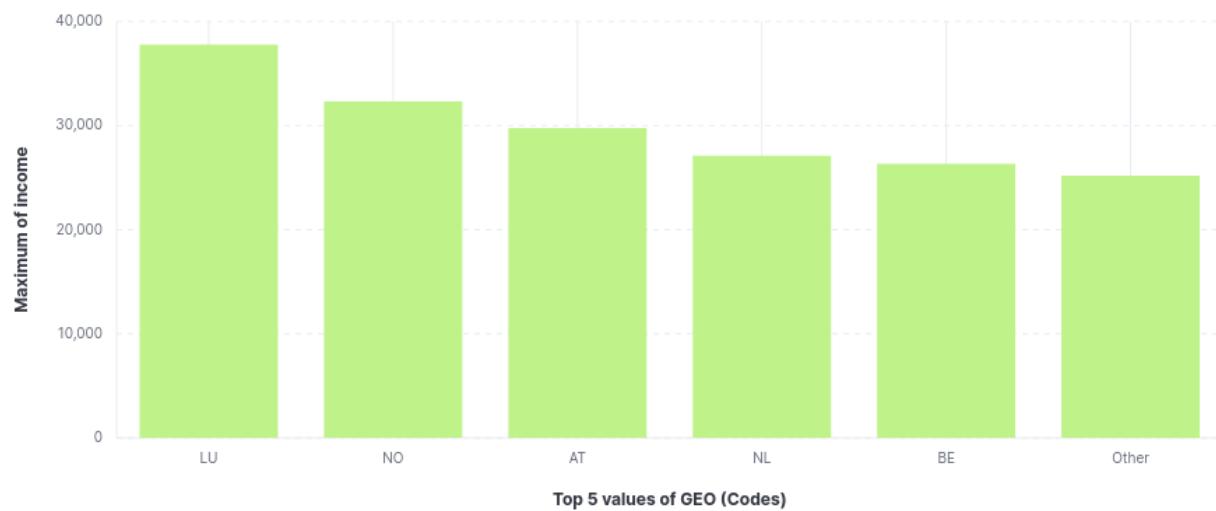
Indexul conține date despre venitul mediu per țară. Valorile trebuie convertite din string cu virgulă în float înainte de inserare. Este o bază importantă pentru analiza nivelului de trai.

- Ca un al doilea pas, am decis să creez o serie de grafice pentru a putea vizualiza mai bine anumite legături și distribuții.

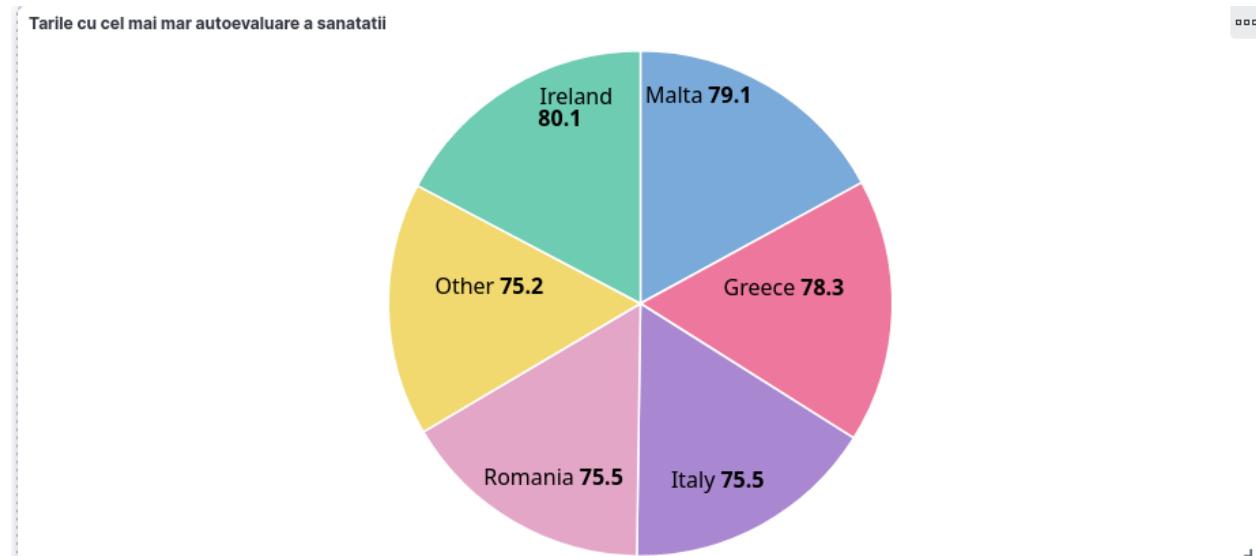




Tarile cu cele mai mari venituri medii in EU



Tarile cu cel mai mare autoevaluare a sanatatii



## **Secțiunea 14- Analiza exploratorie spațială**

### **Introducere în analiza geografică**

- **Analiza spațială (geografică)**
  - Ramură a statisticii care studiază distribuția și relațiile dintre fenomene în funcție de localizarea lor geografică.
  - Permite descoperirea de modele, aglomerări sau disparități teritoriale, greu de identificat prin metode tradiționale.
- **Importanța reprezentării geografice a datelor socio-economice**
  - Oferă o perspectivă vizuală clară asupra distribuției inegalităților teritoriale.
  - Sprijină formularea de politici publice adaptate la nevoile locale și regionale.
  - Permite alocarea eficientă a resurselor în funcție de zonele cu cel mai mare impact social sau economic.
- **Exemple de indicatori cu variație spațială**
  - **Venitul mediu pe locitor** – variază între regiuni bogate și cele mai puțin dezvoltate.
  - **Nivelul de educație** – diferențe între zonele urbane și rurale sau între regiuni cu acces diferit la educație.
  - **Gradul de satisfacție al populației** – reflectă percepții diferite asupra calității vieții în diverse teritorii.
  - **Accesul la servicii de sănătate** – variază în funcție de infrastructura și resursele disponibile la nivel local sau regional.

Pentru început am creat câte un date-view pentru fiecare index.

| <input type="checkbox"/> Name ↑                                                                                                                                           | Spaces | Actions |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|---------|
| <input type="checkbox"/> data_view_cursuri ⓘ Default                                                                                                                      | D      | ⋮       |
| <input type="checkbox"/> .alerts-security.alerts-default,apm-*~transaction*,auditbeat~,endgame~,filebeat~,logs~,packetbeat~,traces-apm*,winlogbeat~,~elastic-cloud-logs~* | D      | ⋮       |
| <input checked="" type="checkbox"/> Security Data View                                                                                                                    | D      | ⋮       |
| <input type="checkbox"/> audit_loguri ⓘ                                                                                                                                   | D      | ⋮       |
| <input type="checkbox"/> audit_loguri1 ⓘ                                                                                                                                  | D      | ⋮       |
| <input type="checkbox"/> audit_loguri_1 ⓘ                                                                                                                                 | D      | ⋮       |
| <input type="checkbox"/> calitatea_vietii ⓘ                                                                                                                               | D      | ⋮       |
| <input type="checkbox"/> data_view_profesori ⓘ                                                                                                                            | D      | ⋮       |
| <input type="checkbox"/> data_view_profesori_fara_data ⓘ                                                                                                                  | D      | ⋮       |
| <input type="checkbox"/> departamente ⓘ                                                                                                                                   | D      | ⋮       |
| <input type="checkbox"/> education ⓘ                                                                                                                                      | D      | ⋮       |

Rows per page: 10 ▾

< 1 2 >

Pentru a vizualiza ce țări sunt disponibile în analiza noastră am realizat un join între codurile ISO și country\_codes din self\_perceived\_health

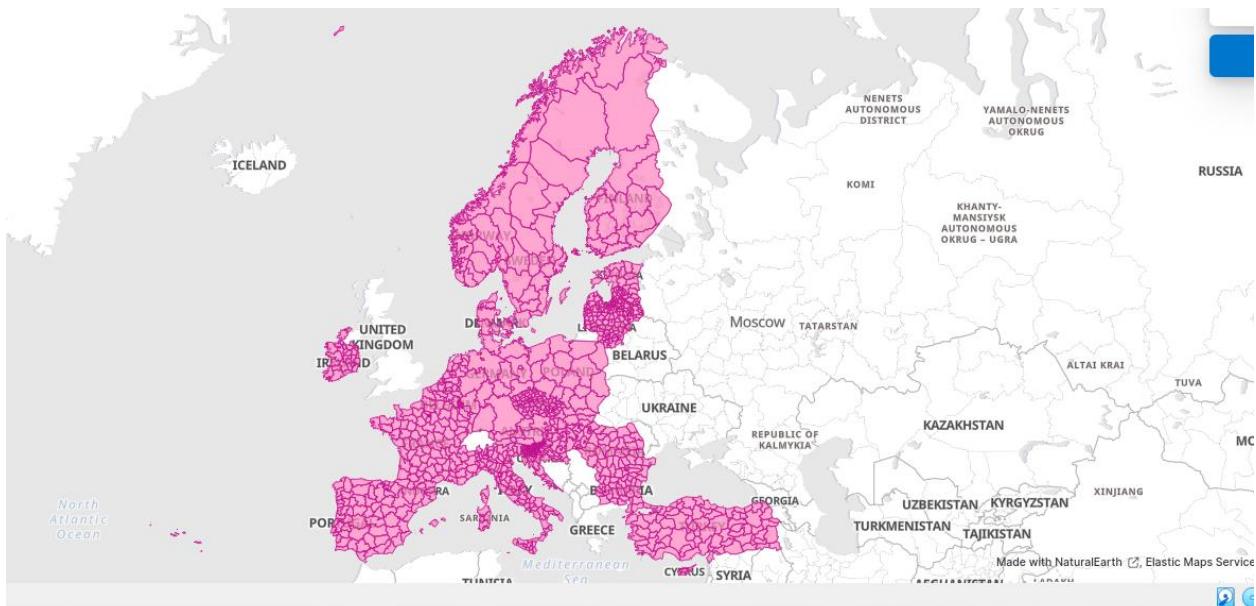
#### Joins ⓘ

Join with terms from country\_code

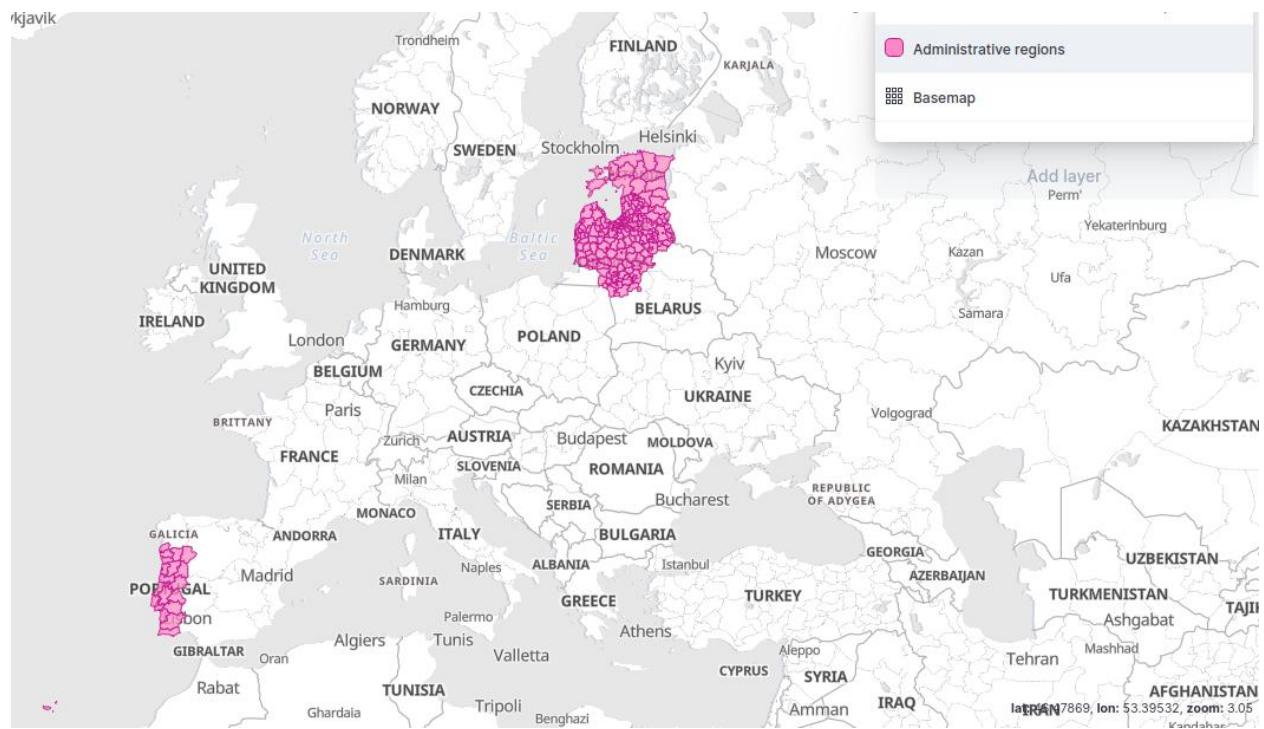
and use metric count where -- add filter --

Apply global search to join

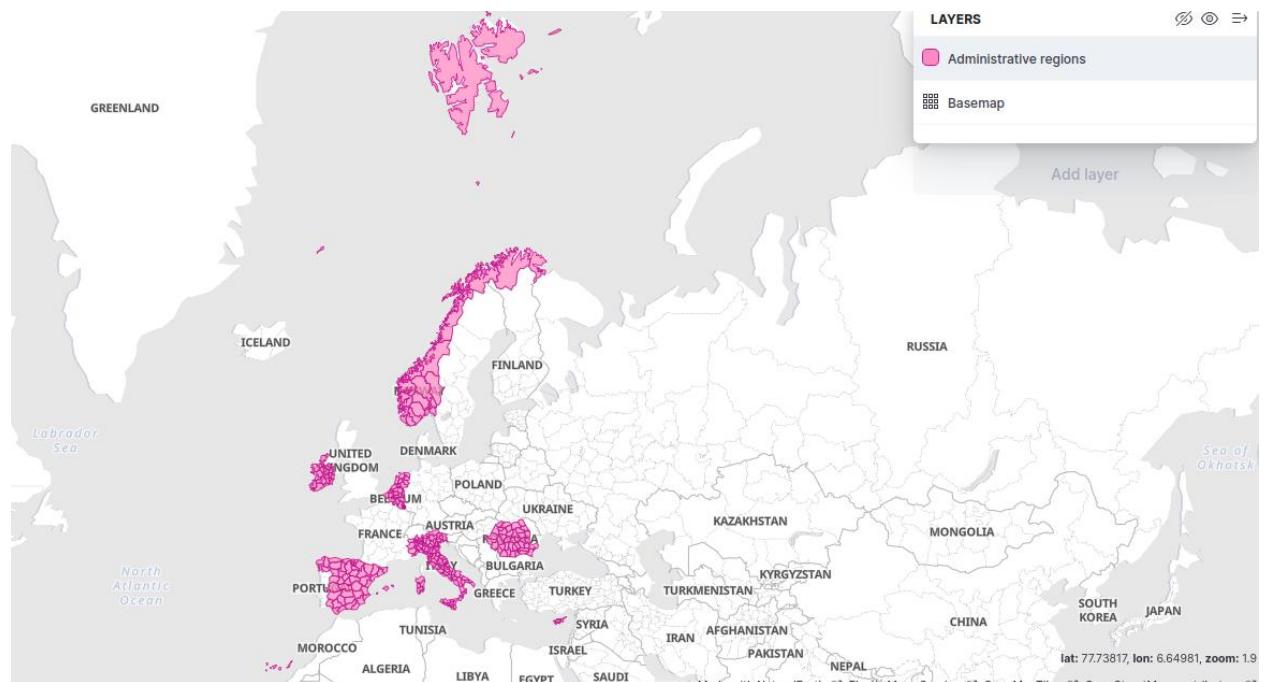
Add spatial join  Add term join



Țările cu proporția persoanelor care cred că au o sănătate bună sau foarte bună sub 60.



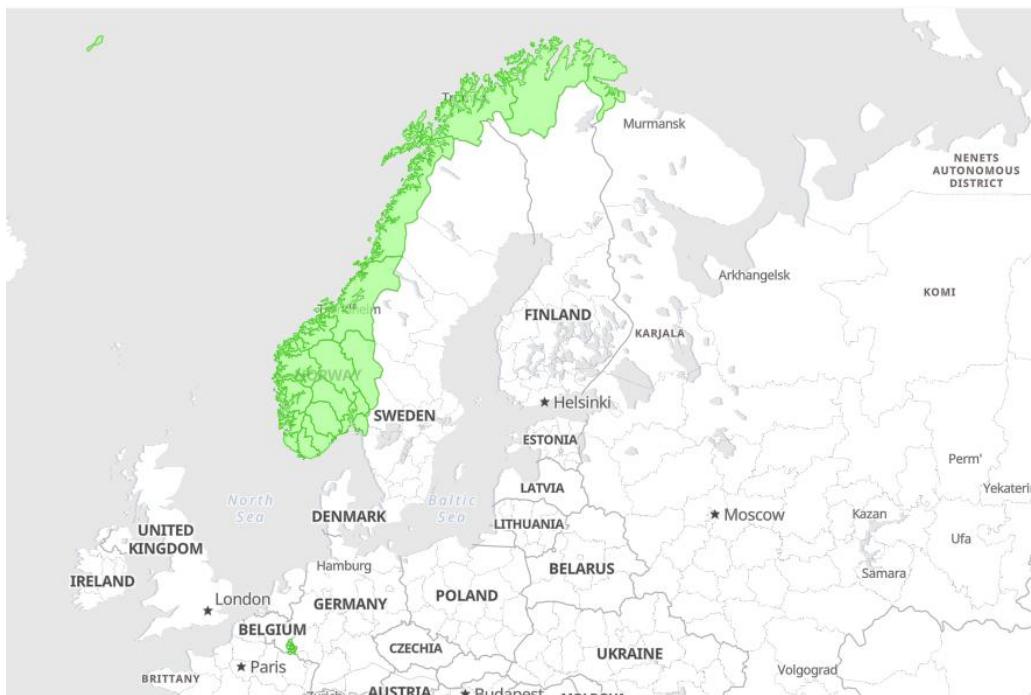
Țările cu proporția persoanelor care cred că au o sănătate bună sau foarte bună peste 70.



Inclusiv România are parte de o perspectivă pozitivă asupra sănătății.



Regiunile cu cel mai mare venit per capita.



Tara cu cel mai mic venit per capita este Turcia.



## Secțiunea 15-Agregarea datelor

### Ce este agregarea datelor?

Agregarea datelor reprezintă un proces fundamental în analiza statistică și în știința datelor, constând în reunirea și sintetizarea informațiilor dintr-un set de date pentru a obține indicatori relevanți, cum ar fi totalul, media, mediana, maximul sau minimul. Prin acest proces, datele individuale sunt transformate în informații sintetice, care permit identificarea unor tendințe generale sau a unor caracteristici globale ale fenomenului studiat.

În contextul analizelor socio-economice, agregarea datelor este esențială pentru a obține valori reprezentative la nivel de grupuri, regiuni sau perioade de timp. De exemplu, calculul venitului

mediu pe țară sau pe regiune oferă un reper util pentru compararea nivelului de trai între diferite teritorii.

Pe lângă simplificarea volumului de date, agregarea permite formularea unor concluzii cu privire la comportamente generale și susține luarea deciziilor pe baza unor informații sintetizate, ușor de interpretat și de comunicat.

### -Venitul mediu în Europa

Sintaxă:

```
POST /income/_search
```

```
{
```

```
  "size": 0,
```

```
  "aggs": {
```

```
    "average_income": {
```

```
      "avg": {
```

```
        "field": "income"
```

```
      }
```

```
    }
```

```
}
```

```
}
```

Rezultat:

20286.48275862069

```

"script": {
  "source": "ctx._source.country_code = ctx._source['GEO (Codes)']; ctx._source.remove('GEO (Codes)')",
  "lang": "painless"
}
}

get income2/_search

//agregare
POST /income/_search
{
  "size": 0,
  "ags": {
    "average_income": {
      "avg": [
        {"ffield": "income"}
      ]
    }
  }
}

1+ {
  "took": 30,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 29,
      "relation": "eq"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "average_income": {
      "value": 20286.48275862069
    }
  }
}

```

Valoarea de 20.286,48 PPS per capita evidențiază nivelul mediu al veniturilor la nivelul statelor europene, ajustat în funcție de puterea de cumpărare. Această medie reflectă o situație economică agregată care, deși oferă o imagine de ansamblu, ascunde variațiile semnificative existente între statele membre.



Țările din Europa de Vest și Nord tind, în general, să înregistreze venituri medii semnificativ peste această valoare, reflectând economii dezvoltate, cu un nivel ridicat al productivității și al calității vieții. În schimb, multe dintre statele din Europa Centrală, de Est sau din zona Balcanilor se situează sub media europeană, ceea ce indică decalaje persistente în dezvoltarea economică și în standardele de trai.

Top 5 țări cu venitul peste cel mediu:

Sintaxă:

```
POST /income/_search
{
  "size": 0,
  "aggs": {
    "top_5_countries": {
      "terms": {
        "field": "GEO (Codes)",
        "size": 5,
        "order": {
          "average_income": "desc"
        }
      },
      "aggs": {
        "average_income": {
          "avg": {
            "field": "income"
          }
        }
      }
    }
  }
}
```

```

get /uncorexx/_search
//aggregate
POST /income/_search
{
  "size": 0,
  "aggs": {
    "average_income": {
      "avg": {
        "field": "income"
      }
    }
  }
}

POST /income/_search
{
  "size": 0,
  "aggs": {
    "top_5_countries": {
      "terms": {
        "field": "GEO ((codes))",
        "order": "desc",
        "size": 5
      },
      "aggs": {
        "average_income": {
          "avg": {
            "field": "income"
          }
        }
      }
    }
  }
}

GET /income/_search

```

▶ ↻

```

18+   "stats": {
11+     "total": {
12+       "value": 29,
13+       "relation": "eq"
14+     },
15+     "max_score": null,
16+     "hits": []
17+   },
18+   "aggregations": {
19+     "top_5_countries": {
20+       "doc_count_error_upper_bound": -1,
21+       "sum_other_doc_count": 24,
22+       "buckets": [
23+         {
24+           "key": "LU",
25+           "doc_count": 1,
26+           "average_income": {
27+             "value": 37781
28+           }
29+         },
29+         {
30+           "key": "NO",
31+           "doc_count": 1,
32+           "average_income": {
33+             "value": 32358
34+           }
35+         },
36+         {
37+           "key": "AT",
38+           "doc_count": 1,
39+           "average_income": {
40+             "value": 29738
41+           }
42+         },
43+         {
44+           "key": "NL",
45+           "doc_count": 1,
46+           "average_income": {
47+             "value": 27891
48+           }
49+         },
50+         {
51+           "key": "BE",
52+           "doc_count": 1,
53+           "average_income": {
54+             "value": 26315
55+           }
56+         }
57+       ]
58+     }
59+   }
60+ }
61+ }
```

În urma acestui output putem spune că țările cu cel mai mare venit per capita sunt:

- Luxemburg
- Norvegia
- Austria
- Tările de Jos
- Belgia

## Aflarea scorului de satisfacție ca și componit.

Țara: Belgia

Pentru început am calculat valorile aggregate pentru educație, sănătate și venit, pentru țara aferentă.

```
Click to send request
```

```
[{"size": 0, "query": {"term": {"GEO (Codes)": "BE"}}, "aggs": {"average_income": {"avg": {"field": "self_perceived_health_g_v_g"}}}}, {"size": 0, "query": {"term": {"GEO (Codes)": "BE"}}, "aggs": {"average_income": {"avg": {"field": "%edu"}}}}]
```

```
3  "timed_out": false,
4  "_shards": {
5    "total": 1,
6    "successful": 1,
7    "skipped": 0,
8    "failed": 0
9  },
10 "hits": [
11   {
12     "total": {
13       "value": 1,
14       "relation": "eq"
15     },
16     "max_score": null,
17     "hits": []
18   }
19 ],
20 "aggregations": [
21   {
22     "average_income": {
23       "value": 74.5
24     }
25   }
26 ]
```

Le-am adăugat într-un excel.

| Income | %Edu | self_perceived_health |
|--------|------|-----------------------|
| 26315  | 0.77 | 0.745                 |

După le-am normalizat după formula:

$$\text{Normalizare} = \frac{\text{Valoare} - \text{Valoare min}}{\text{Valoare Maxima} - \text{Valoare Minima}}$$

După am aplicat AVG asupra acestor date, rezultatul final fiind 8.7, ceea că însemnă că pe o scără de la 1 la 10, scorul de satisfacere a vieții în România ar fi 8.7. Însă acest rezultat este diferit de cel din baza de date, mai exact, de 7.6.

## **Secțiunea 16-Ce am învățat?**

Pe parcursul utilizării platformei ELK, am avut ocazia să înțeleg și să experimentez modul în care datele pot fi colectate, procesate, căutate și vizualizate într-un sistem integrat și performant. În primul rând, am învățat rolul fiecărei componente din acest ecosistem. Elasticsearch mi-a demonstrat cât de important este să ai o soluție care să permită interogarea rapidă și flexibilă a unor volume mari de date, folosind un limbaj de interogare JSON ușor de învățat și aplicat. Am înțeles cum sunt structurate datele sub formă de documente și cum se realizează indexarea lor pentru a optimiza căutările.

În Kibana, am descoperit partea vizuală a analizei. Am învățat să construiesc dashboarduri interactive și grafice care să ajute la interpretarea și comunicarea rezultatelor. Am realizat cât de important este ca datele să nu fie doar colectate și procesate, ci și prezentate într-o formă ușor de înțeles pentru utilizatori sau factori de decizie.

Pe lângă aspectele tehnice, experiența cu ELK m-a ajutat să conștientizez importanța securității și a restaurării datelor, învățând despre rolul autentificării, autorizării și realizării de backup-uri pentru a proteja informațiile gestionate.

În concluzie, lucrul cu ELK m-a ajutat să-mi dezvolt abilități practice valoroase în gestionarea și analiza datelor, dar și să înțeleg mai bine procesele care stau la baza unui sistem modern de procesare a datelor în timp real.