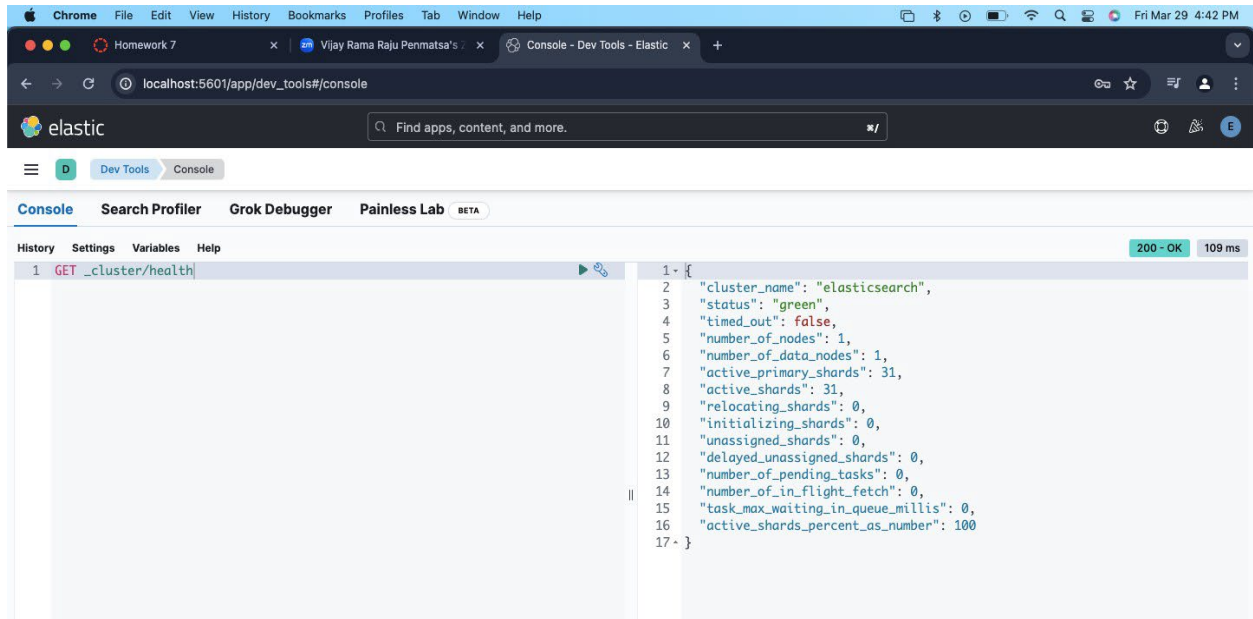


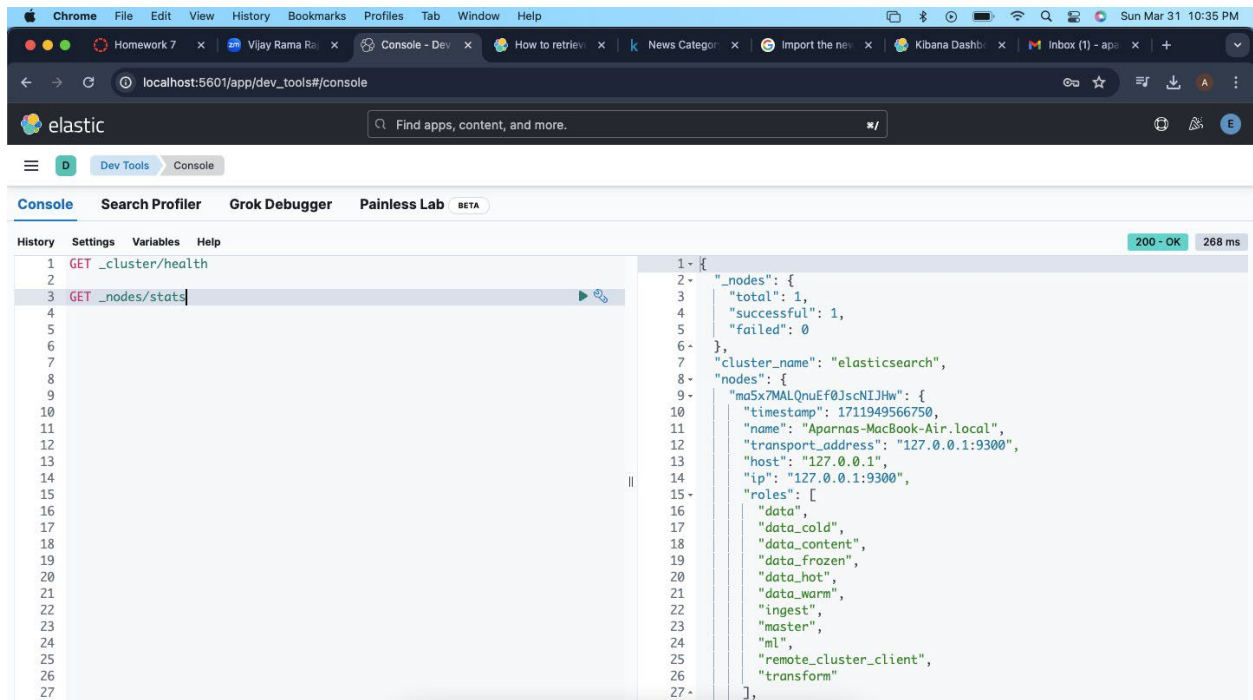
Homework_7_AparnaBharathi_Suresh

Question 1:

Show the cluster health and node stats.

Screenshot 1:





Query 1:

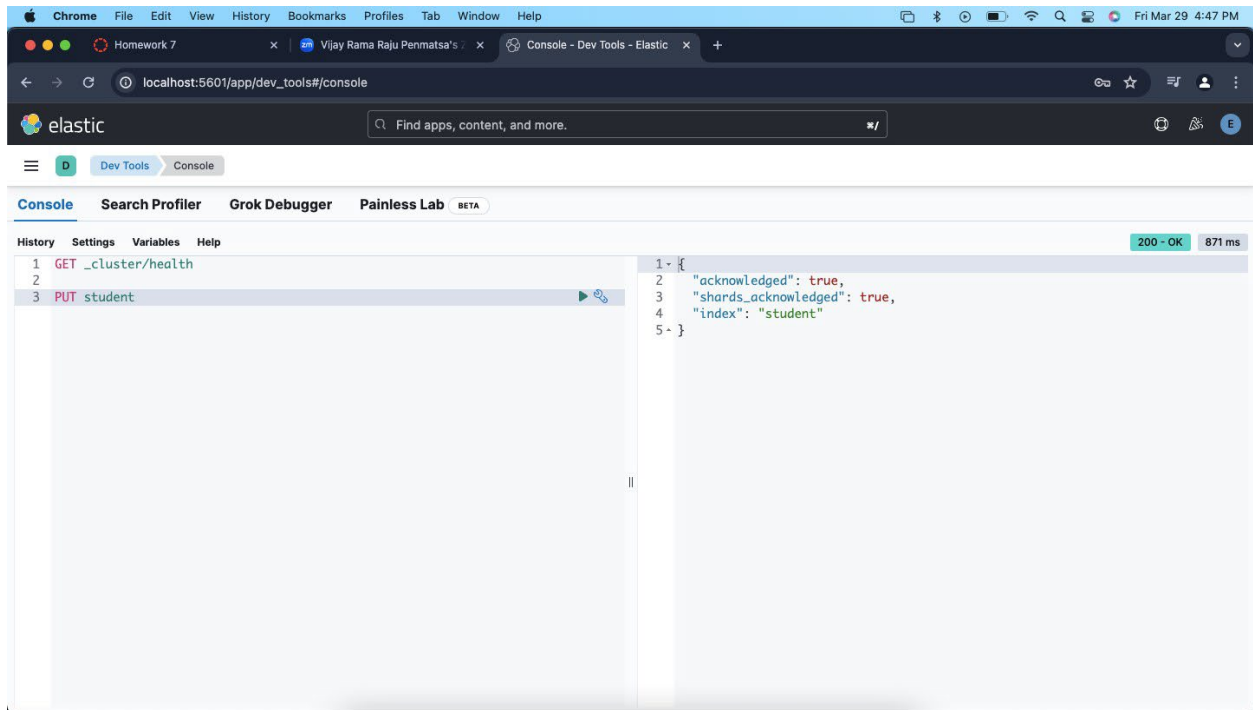
GET _cluster/health

GET _nodes/stats

Question 2:

Create an index with the name 'student'.

Screenshot 2:



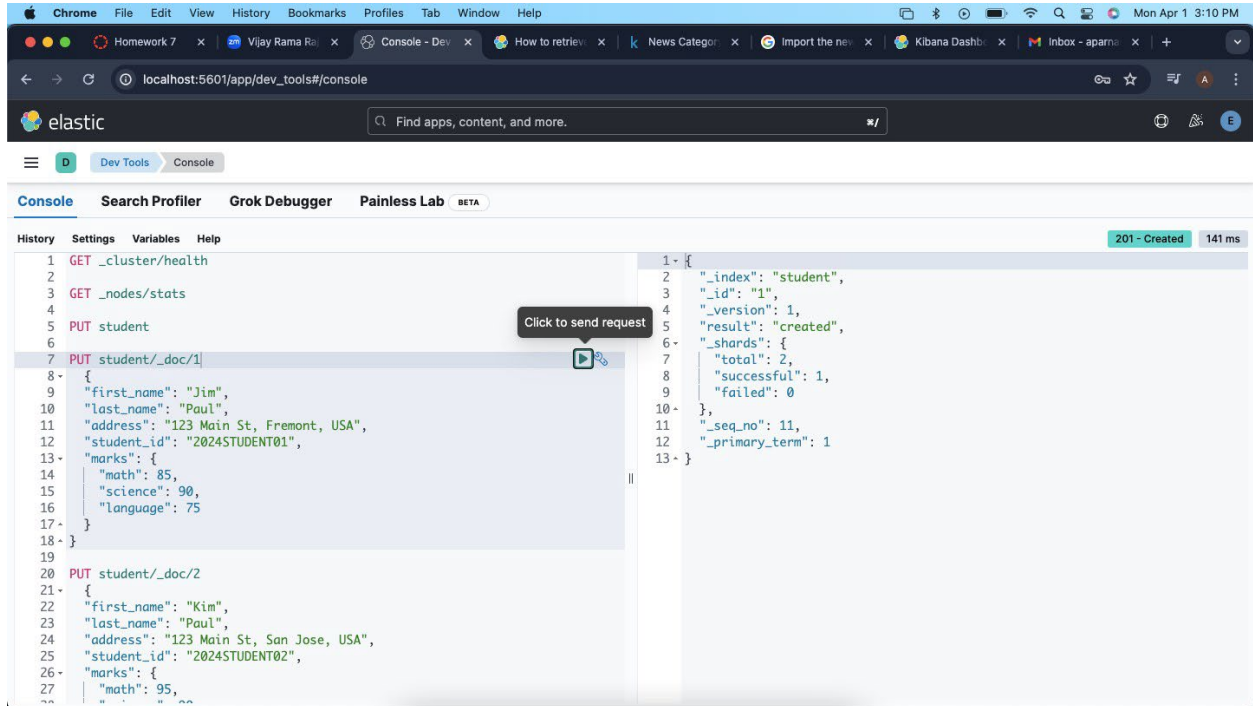
Query 2:

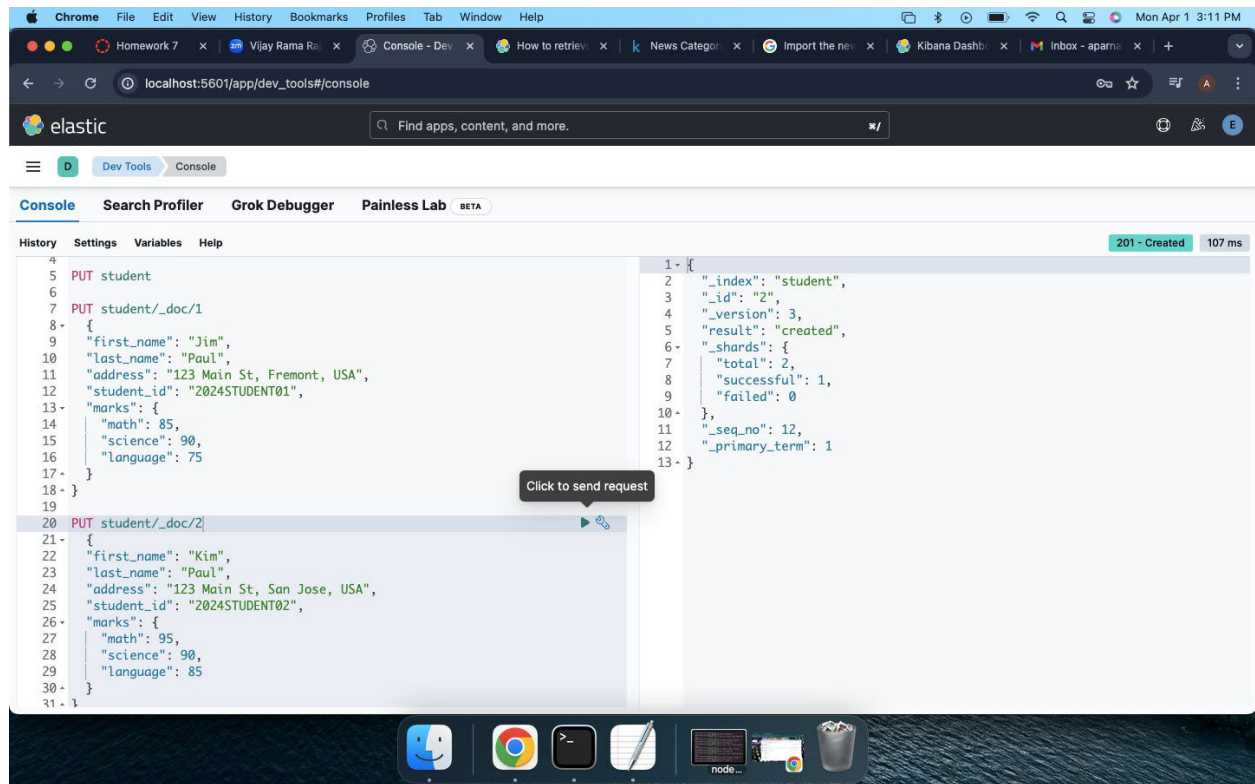
PUT student

Question 3:

Generate an id of your choice for the student to create 2 documents [Student should have fields: First name, last name, Address, Student_id, Marks]

Screenshot 3:





Query 3:

PUT student/_doc/1

```
{
  "first_name": "Jim",
  "last_name": "Paul",
  "address": "123 Main St, Fremont, USA",
  "student_id": "2024STUDENT01",
  "marks": {
    "math": 85,
    "science": 90,
    "language": 75
  }
}
```

PUT student/_doc/2

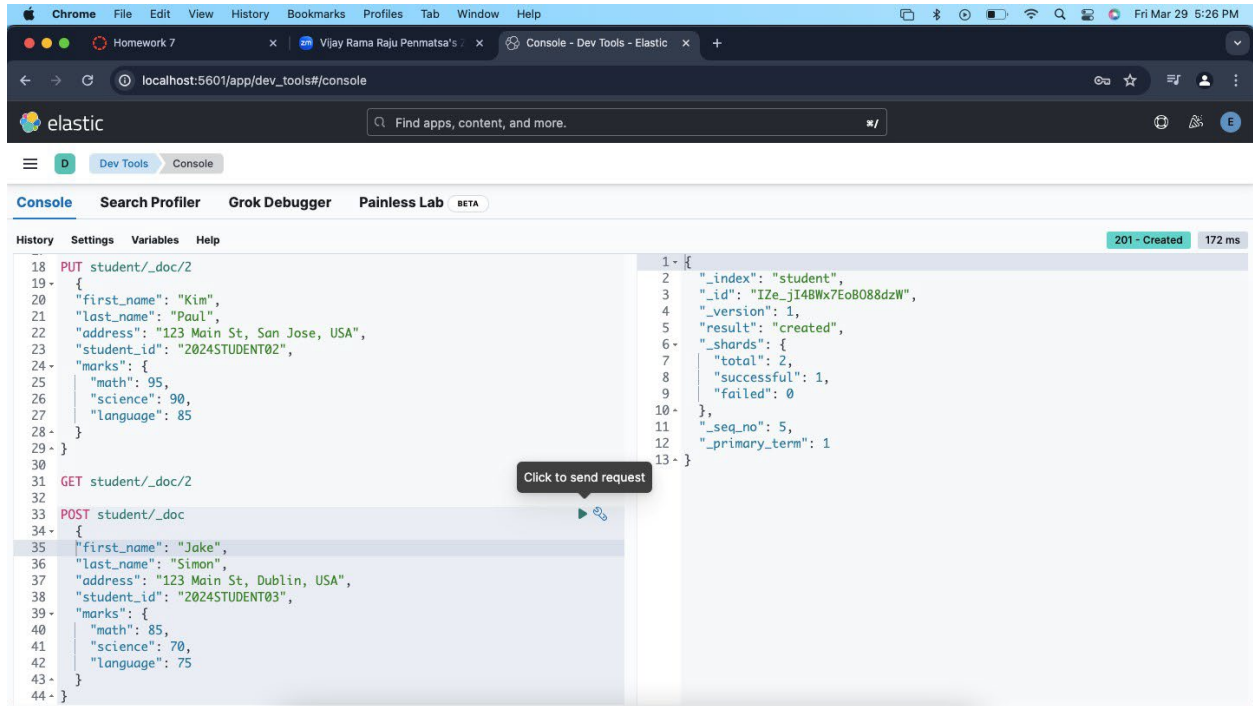
```
{
  "first_name": "Kim",
  "last_name": "Paul",
  "address": "123 Main St, San Jose, USA",
  "student_id": "2024STUDENT02",
  "marks": {
    "math": 95,
    "science": 90,
    "language": 85
  }
}
```

```
“student_id”:”2024STUDENT02”,  
“marks”:{  
  “math”:95,  
  “science”:90,  
  “language”:85  
}  
}
```

Question 4:

Autogenerate id for the student to create 1 document.

Screenshot 4:



Query 4:

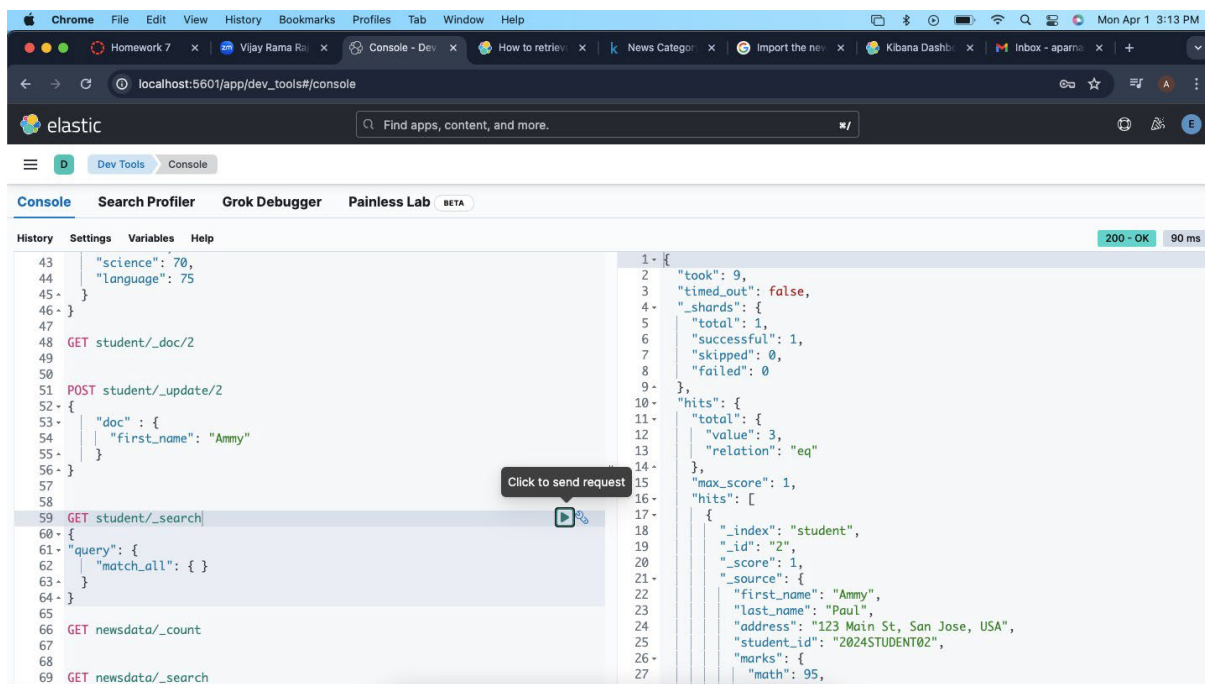
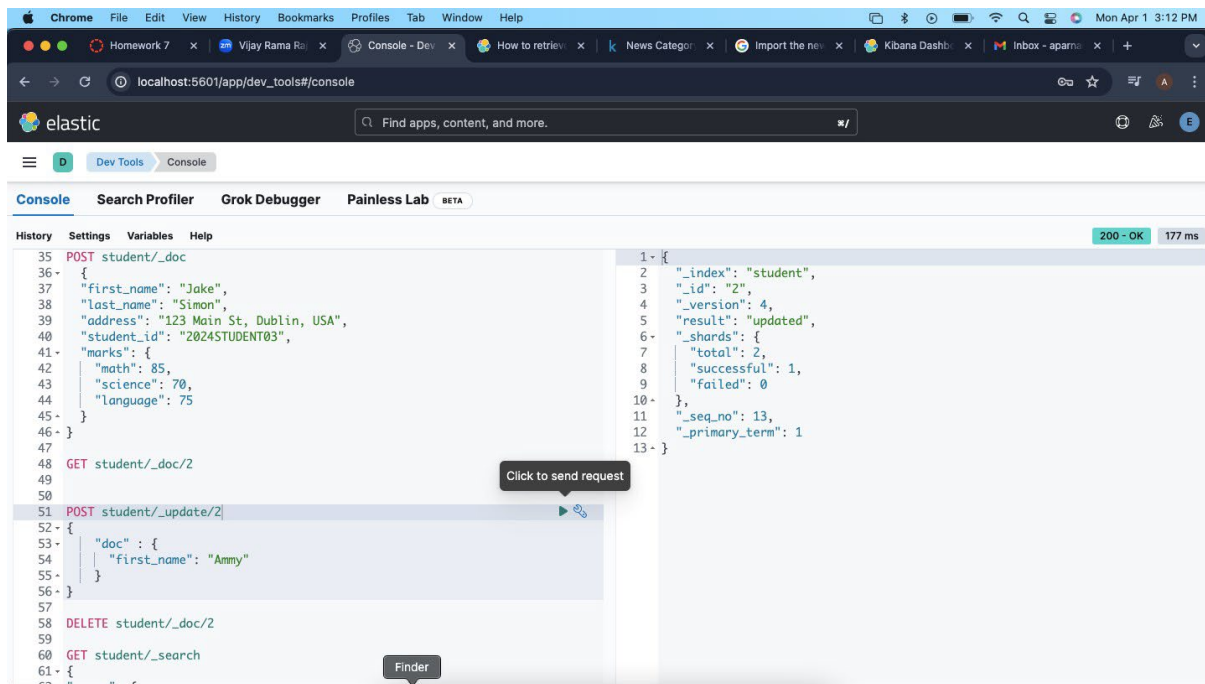
POST student/_doc

```
{
  "first_name": "Jake",
  "last_name": "Simon",
  "address": "123 Main St, Dublin, USA",
  "student_id": "2024STUDENT03",
  "marks": {
    "math": 85,
    "science": 70,
    "language": 75
  }
}
```

Question 5:

Perform update operation by any field of the document and display all docs after updation.

Screenshot 5:



Query 5:

POST student/_update/2

```
{  
  "doc":{  
    "first_name":"Ammy"  
  }  
}
```

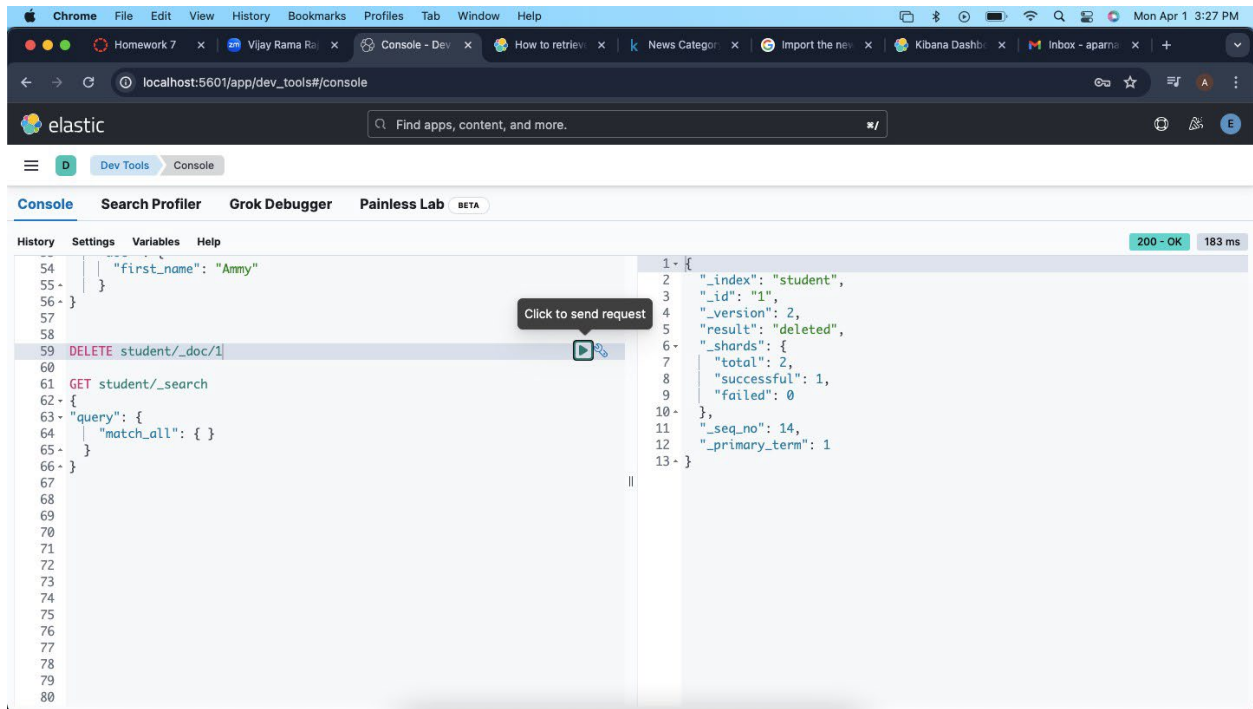
GET student/_search

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

Question 6:

Delete any one document and display all docs after deletion

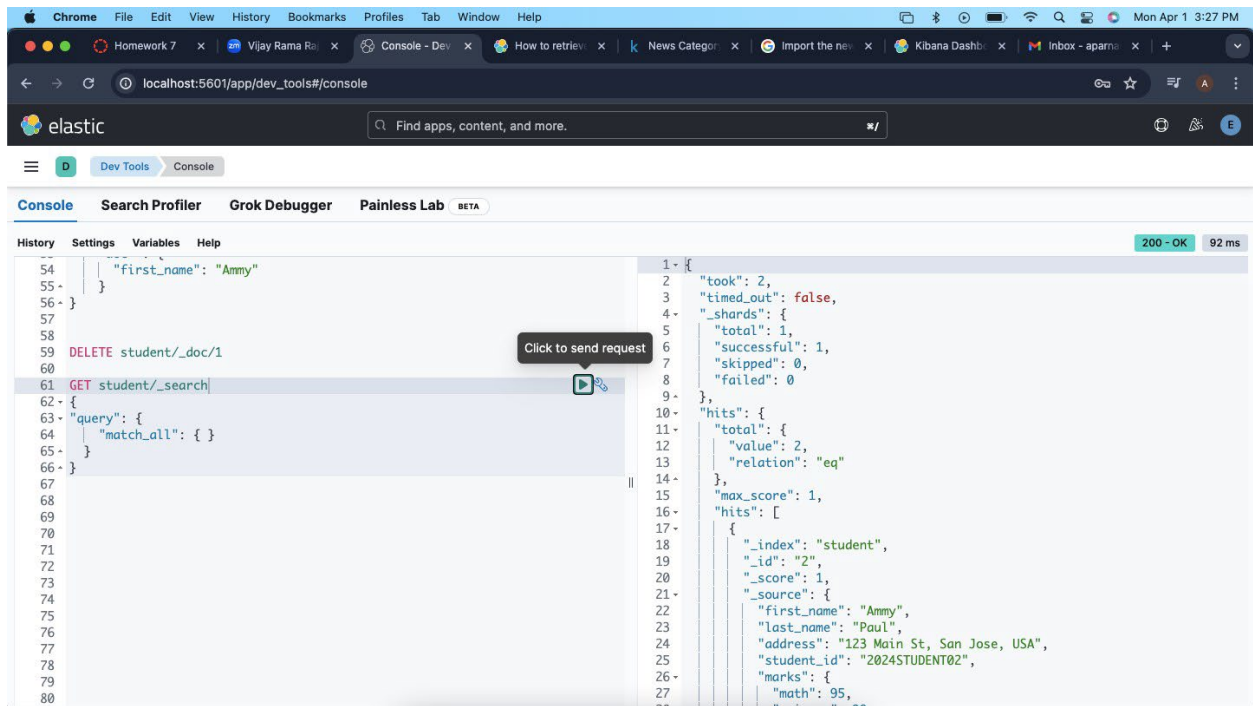
Screenshot 6:



The screenshot shows the Elastic DevTools console in a Chrome browser. The console is displaying a series of requests and responses. The first request is a DELETE request to delete a document with ID 1 from the 'student' index. The response is a JSON object indicating the deletion was successful.

```
54 { "first_name": "Ammy"
55 }
56 }
57
58
59 DELETE student/_doc/1
60
61 GET student/_search
62 {
63   "query": {
64     "match_all": { }
65   }
66 }
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

```
1 {
2   "_index": "student",
3   "_id": "1",
4   "_version": 2,
5   "result": "deleted",
6   "_shards": {
7     "total": 2,
8     "successful": 1,
9     "failed": 0
10  },
11   "_seq_no": 14,
12   "_primary_term": 1
13 }
```



The screenshot shows the Elastic DevTools console in a Chrome browser. The console is displaying a series of requests and responses. The first request is a GET request to search for documents in the 'student' index. The response is a JSON object indicating the search was successful and returned one hit.

```
54 { "first_name": "Ammy"
55 }
56 }
57
58
59 DELETE student/_doc/1
60
61 GET student/_search
62 {
63   "query": {
64     "match_all": { }
65   }
66 }
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

```
1 {
2   "took": 2,
3   "timed_out": false,
4   "_shards": {
5     "total": 1,
6     "successful": 1,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": {
12      "value": 2,
13      "relation": "eq"
14    },
15    "max_score": 1,
16    "hits": [
17      {
18        "_index": "student",
19        "_id": "2",
20        "_score": 1,
21        "_source": {
22          "first_name": "Ammy",
23          "last_name": "Paul",
24          "address": "123 Main St, San Jose, USA",
25          "student_id": "2024STUDENT02",
26          "marks": {
27            "math": 95,
28            "science": 88
29          }
30        }
31      }
32    ]
33  }
34 }
```

Query 6:

DELETE student/_doc/1

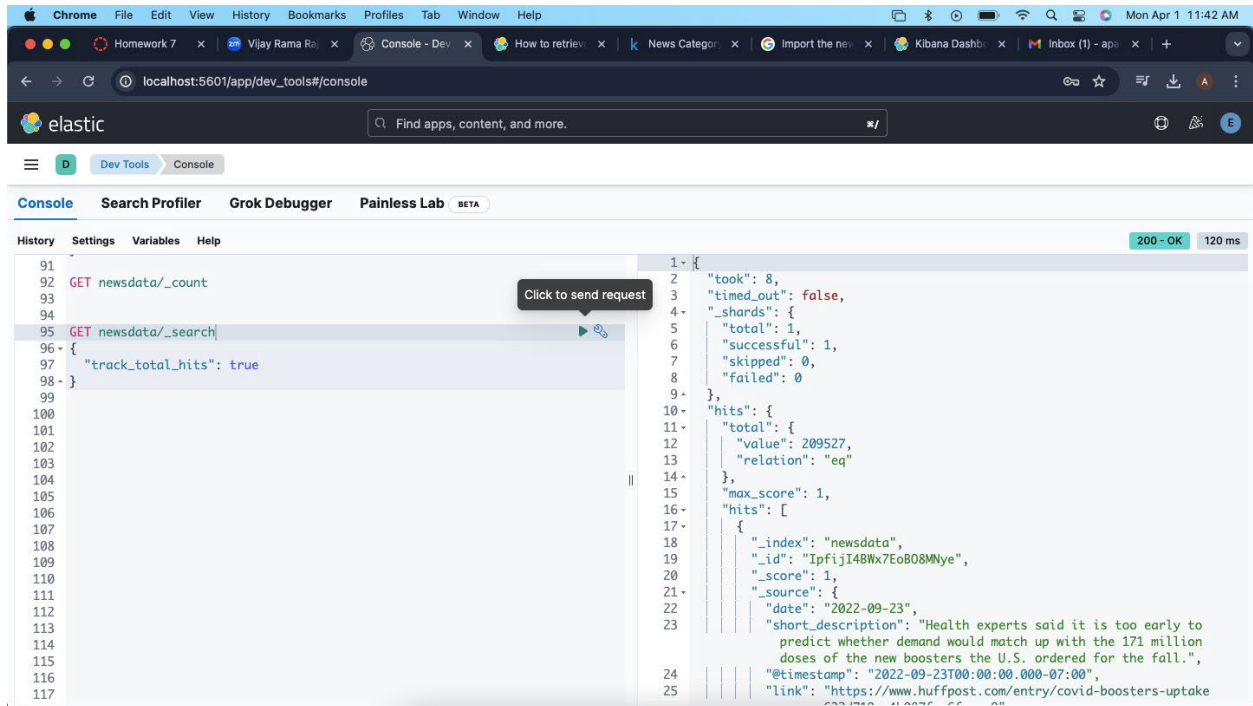
GET student/_search

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

Question 7:

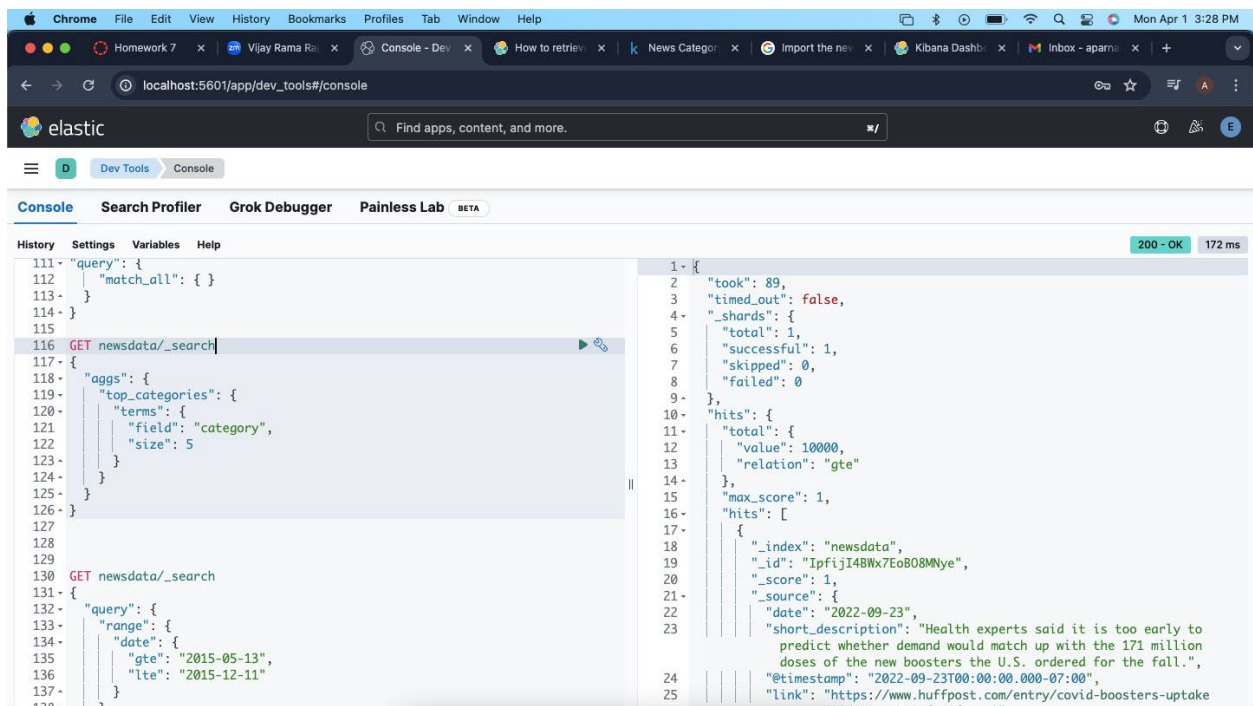
Show the total no of hits in the document and analyze the data to show the top 5 categories of news headlines in our dataset

Screenshot 7:



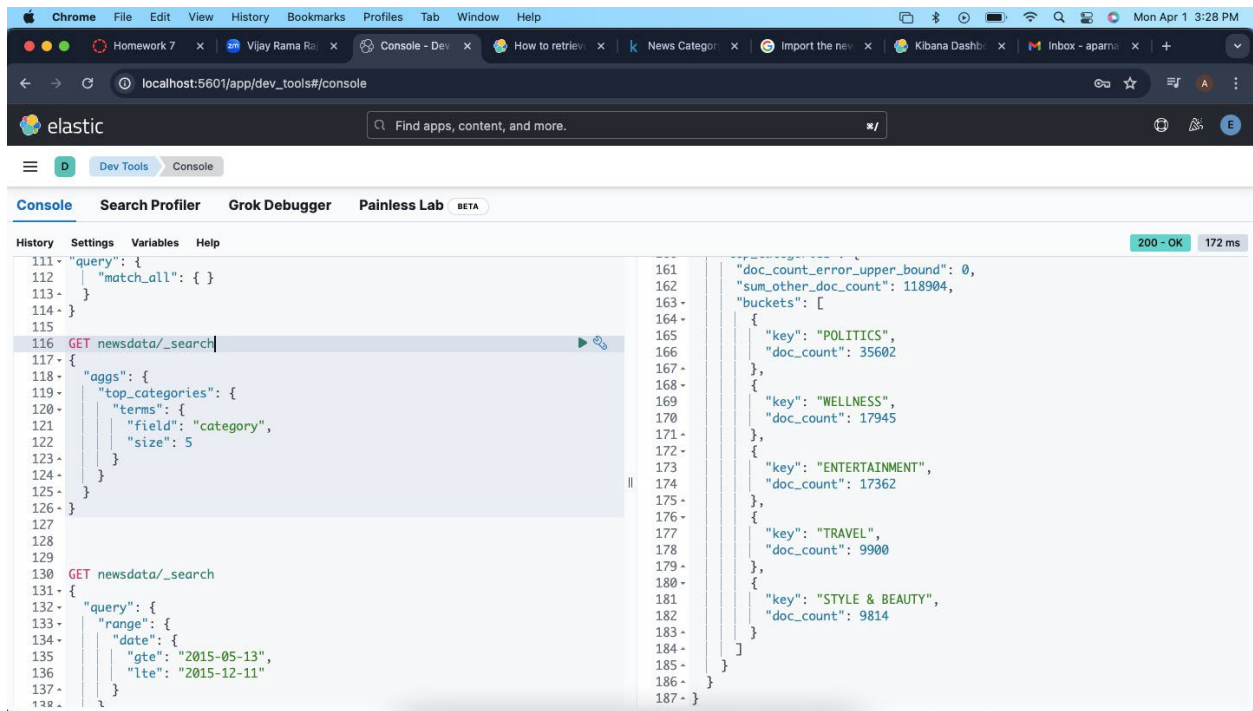
```
1 GET newsdata/_count
2
3
4
5 GET newsdata/_search
6 {
7   "track_total_hits": true
8 }
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

```
1 {
2   "took": 8,
3   "timed_out": false,
4   "_shards": {
5     "total": 1,
6     "successful": 1,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": {
12      "value": 209527,
13      "relation": "eq"
14    },
15    "max_score": 1,
16    "hits": [
17      {
18        "_index": "newsdata",
19        "_id": "IpfiI48Wx7EoB08Mnye",
20        "_score": 1,
21        "_source": {
22          "date": "2022-09-23",
23          "short_description": "Health experts said it is too early to predict whether demand would match up with the 171 million doses of the new boosters the U.S. ordered for the fall.",
24          "@timestamp": "2022-09-23T00:00:00.000-07:00",
25          "link": "https://www.huffpost.com/entry/covid-boosters-uptake"
```



```
111 "query": {
112   "match_all": { }
113 }
114 }
115
116 GET newsdata/_search
117 {
118   "aggs": {
119     "top_categories": {
120       "terms": {
121         "field": "category",
122         "size": 5
123       }
124     }
125   }
126 }
127
128
129 GET newsdata/_search
130 {
131   "query": {
132     "range": {
133       "date": {
134         "gte": "2015-05-13",
135         "lte": "2015-12-11"
136       }
137     }
138   }
139 }
```

```
1 {
2   "took": 89,
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": 10000,
13      "relation": "gte"
14    },
15    "max_score": 1,
16    "hits": [
17      {
18        "_index": "newsdata",
19        "_id": "IpfiI48Wx7EoB08Mnye",
20        "_score": 1,
21        "_source": {
22          "date": "2022-09-23",
23          "short_description": "Health experts said it is too early to predict whether demand would match up with the 171 million doses of the new boosters the U.S. ordered for the fall.",
24          "@timestamp": "2022-09-23T00:00:00.000-07:00",
25          "link": "https://www.huffpost.com/entry/covid-boosters-uptake"
```



Query 7:

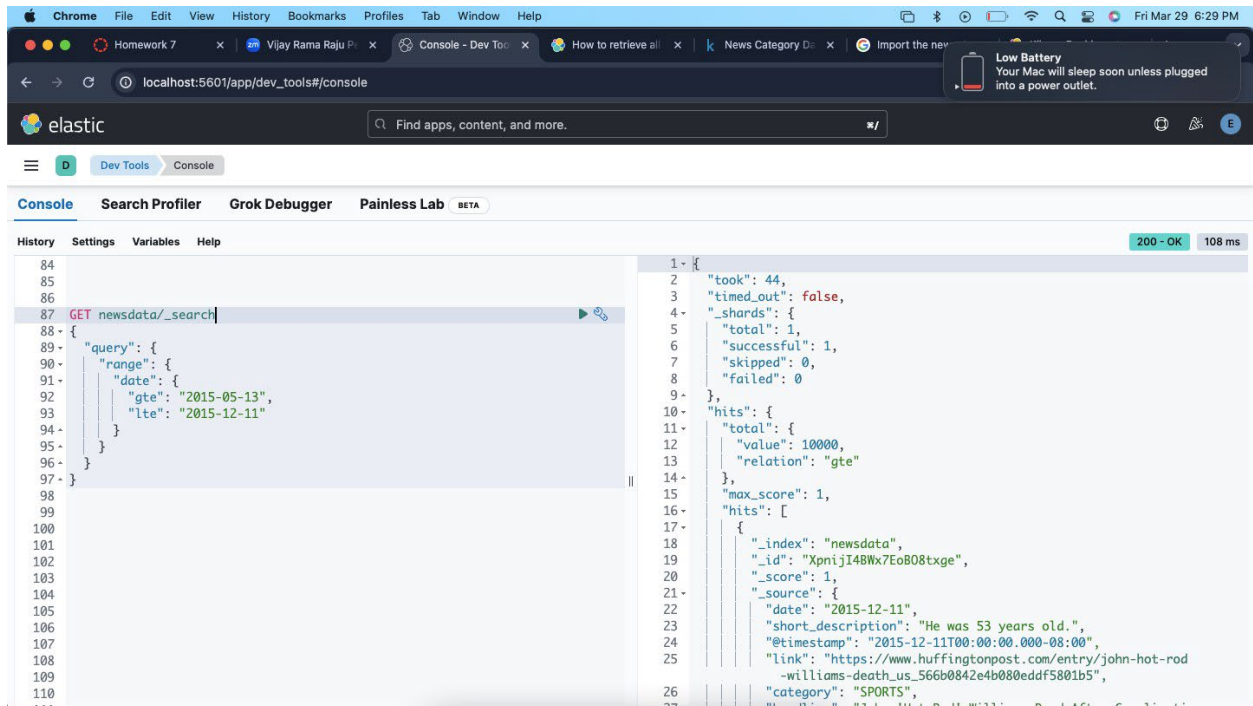
```
GET newsdata/_search
{
  "track_total_hits": true
}
```

```
GET newsdata/_search
{
  "aggs": {
    "top_categories": {
      "terms": {
        "field": "category",
        "size": 5
      }
    }
  }
}
```

Question 8:

Search for data between May 13, 2015, to December 11, 2015

Screenshot 8:



Question 9:

Write a SQL Query to display date, channelGrouping as Channel, total visits, total transactions and total transaction revenue Records, and display the top 1500 rows.

Screenshot 9:

The screenshot shows the Google Cloud BigQuery console interface. The Explorer on the left lists various datasets, including 'google_analytics_sample' and 'ga_sessions_20170801'. The main editor displays a SQL query: `SELECT `date`, channelGrouping AS Channel, totals.visits, totals.transactions, totals.totalTransactionRevenue FROM `bigquery-public-data.google-analytics-sample-ga_sessions_20170801` LIMIT 1500;`. The 'Query results' section at the bottom shows a table with columns: Row, date, Channel, Visits, transactions, and totalTransactionRevenue. The table contains 8 rows of data, all with 'visits' of 1 and 'transactions' of null. The 'totalTransactionRevenue' column is also null for all rows.

Row	date	Channel	Visits	transactions	totalTransactionRevenue
1	20170801	Organic Search	1	null	null
2	20170801	Referral	1	null	null
3	20170801	Referral	1	null	null
4	20170801	Referral	1	null	null
5	20170801	Referral	1	null	null
6	20170801	Paid Search	1	null	null
7	20170801	Referral	1	null	null
8	20170801	Referral	1	null	null

Query 9:

```
SELECT `date`, channelGrouping AS  
Channel, totals.visits, totals.transactions, totals.totalTransactionRevenue FROM  
`bigquery-public-data.google-analytics-sample-ga_sessions_20170801` LIMIT 1500;
```

Question 10:

Write a SQL query to retrieve the number of sessions that started from the 'United States' country for each browser version in the dataset.

Screenshot 10:

The screenshot shows the Google Cloud BigQuery console interface. On the left, the Explorer pane displays the project hierarchy, including the 'ga_sessions' table. The main editor area shows a SQL query titled 'Untitled 2'. The query is as follows:

```
1 SELECT 'date', channelGrouping AS Channel, totals.newVisits, totals.transactions, totals.totalTransactionRevenue FROM `bigquery-public-data.google_analytics_sample.ga_sessions_20170801`, LIMIT 1500;  
2  
3 SELECT device.browserVersion, COUNT(*) AS Number_of_Sessions FROM `bigquery-public-data.google_analytics_sample.ga_sessions_20170801` WHERE geoNetwork.country="United States" GROUP BY device.browserVersion;
```

Below the query editor, the 'Query results' section is visible, showing a table with the following data:

Row	browserVersion	Number_of_Sessions
1	not available in demo dataset	1287

Query 10:

```
SELECT device.browserVersion, COUNT(*) AS Number_of_Sessions FROM `bigquery-public-data.google_analytics_sample.ga_sessions_20170801` WHERE geoNetwork.country="United States" GROUP BY device.browserVersion;
```