

DSC-650-Week-08-Mukherjee

Author : CHitramoy MUKherjee

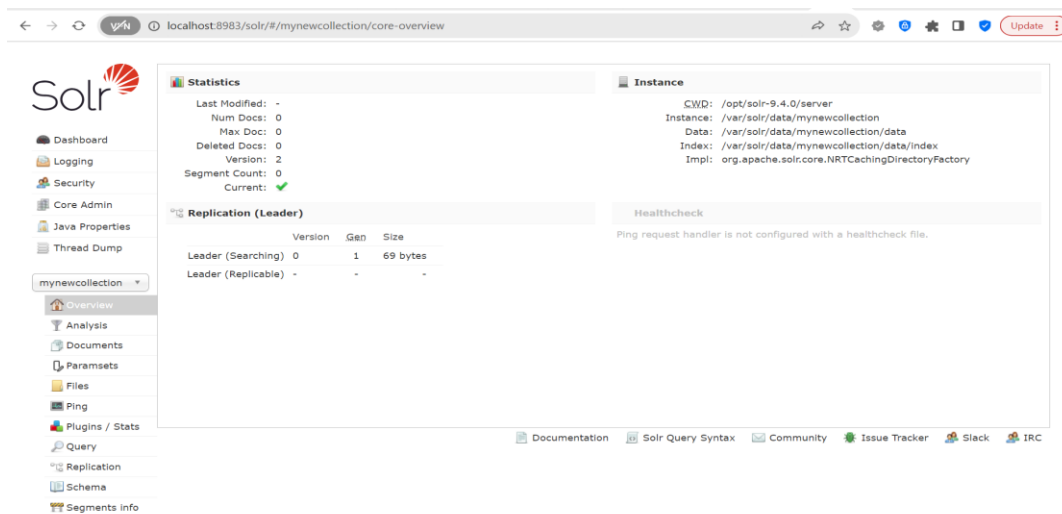
Date : 02/01/2024

Exercise 1: Create a Solr collection named 'mynewcollection'.

Screenshot showing 'mynewcollection' in the Solr Web Interface.

Exercise 2: Verify that 'mynewcollection' has been successfully created.

Access Solr's Web Interface at <http://localhost:8983/solr/> and check for 'mynewcollection' under the "Core Selector" dropdown.



Exercise 3: Let's add the generated data to our collection.

Screenshot showing successful data ingestion messages in the terminal.

```
drwxrwxr-x 2 chitramoy chitramoy 4096 Dec 5 01:26 solr
drwxr-xr-x 3 root root 4096 Dec 5 01:29 nifi
drwxrwxr-x 6 chitramoy chitramoy 4096 Dec 5 01:31 hadoop-hive-spark-hbase
drwxrwxr-x 4 chitramoy chitramoy 4096 Jan 28 19:09 kafka
chitramoy@bigdata:~/dsc650-infra/bellevue-bigdata$ cd solr
chitramoy@bigdata:~/dsc650-infra/bellevue-bigdata/solr$ sudo docker-compose up -d
Starting solr_1 ... done
chitramoy@bigdata:~/dsc650-infra/bellevue-bigdata/solr$ sudo docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                    NAMES
6a404867c512   solr       "docker-entrypoint.s..." 22 hours ago   Up 8 seconds   0.0.0.0:8983->8983/tcp, :::8983->8983/tcp   solr_solr_1
chitramoy@bigdata:~/dsc650-infra/bellevue-bigdata/solr$ sudo docker exec -it solr_solr_1 bash
solr@6a404867c512:/opt/solr-9.4.0$ /opt/solr/bin/solr create -c mynewcollection
WARNING: Using _default_ configset with data driven schema functionality. NOT RECOMMENDED for production use.
To turn off: bin/solr config -c mynewcollection -p 8983 -action set-user-property -property update.autoCreateFields -value false

Created new core 'mynewcollection'
solr@6a404867c512:/opt/solr-9.4.0$ echo '[
{"id":"1", "name":"Product A", "category":"Electronics", "price":100},
{"id":"2", "name":"Product B", "category":"Books", "price":20}
]' > /tmp/products.json
solr@6a404867c512:/opt/solr-9.4.0$ /opt/solr/bin/post -c mynewcollection /tmp/products.json
The bin/post script is deprecated in favour of the bin/solr post command. Please update your scripts.
/opt/java/openjdk/bin/java -classpath /opt/solr/server/solr-webapp/webapp/WEB-INF/lib/solr-core-9.4.0.jar -Dauto=yes -Dc=mynewcollection -Ddata=files org.apa
che.solr.cli.SimplePostTool /tmp/products.json
SimplePostTool version 9.4.0
Posting files to [base] url http://localhost:8983/solr/mynewcollection/update...
Entering auto mode. File endings considered are xml,json,jsonl, csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ots,rtf,htm,html,txt,log
POSTing file products.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/mynewcollection/update...
Time spent: 0:00:00.648
solr@6a404867c512:/opt/solr-9.4.0$
```

Exercise 4: Use Solr's query interface to retrieve all documents from 'mynewcollection'.

curl "http://localhost:8983/solr/mynewcollection/select?q=*:*"

Exercise 5: Query for products in the 'Electronics' category. curl

"http://localhost:8983/solr/mynewcollection/select?q=category:Electronics"

```
solr@6a404867c512:/opt/solr-9.4.0$ curl "http://localhost:8983/solr/mynewcollection/select?q=*"
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{"q":"*:*"
  },
  "response":{
    "numFound":2,
    "start":0,
    "numFoundExact":true,
    "docs":[{
      "id":"1",
      "name":["Product A"],
      "category":["Electronics"],
      "price":100,
      "_version_":1789846779737407488
    },{
      "id":"2",
      "name":["Product B"],
      "category":["Books"],
      "price":20,
      "_version_":1789846779825487872
    }]
  }
}
solr@6a404867c512:/opt/solr-9.4.0$ curl "http://localhost:8983/solr/mynewcollection/select?q=category:Electronics"
{
  "responseHeader":{
    "status":0,
    "QTime":1,
    "params":{"q":"category:Electronics"
  },
  "response":{
    "numFound":1,
    "start":0,
    "numFoundExact":true,
    "docs":[{
      "id":"1",
      "name":["Product A"],
      "category":["Electronics"],
      "price":100,
      "_version_":1789846779737407488
    }]
  }
}
solr@6a404867c512:/opt/solr-9.4.0$
```

Exercise 6: Try a faceted search to count the number of products in each category.

The screenshot shows the Solr Admin UI interface. On the left is a sidebar with navigation links: Dashboard, Logging, Security, Core Admin, Java Properties, Thread Dump, mynewcollection (selected), Overview, Analysis, Documents, Params, Files, Ping, Plugins / Stats, Query (selected), Replication, Schema, and Segments Info. The main content area is divided into two panels. The left panel contains search configuration options: sort (empty), start/rows (0/10), fl (empty), df (empty), paramset(s) (empty), wt (dropdown), indent on (checked), debugQuery (unchecked), defType (dropdown), hl (unchecked), facet (checked), facet.query (empty), facet.field (category), facet.prefix (empty), and facet.contains (empty). The right panel displays the JSON response for the query `q=*&q.op=OR&indent=true&facet=true&facet.field=category&usePara...`. The response shows 2 documents found, with facets for 'category' showing counts for 'books' (1) and 'electronics' (1). The document details for 'Product A' (category: Electronics, price: 100) and 'Product B' (category: Books, price: 20) are listed.

Select name, category where price less than 25.

localhost:8983/solr/#/mynewcollection/query?q=*&q.op=OR&indent=true&spatial=true&fl=name,category&fq=price:[* TO 25]

Solr

- Dashboard
- Logging
- Security
- Core Admin
- Java Properties
- Thread Dump
- mynewcollection
 - Overview
 - Analysis
 - Documents
 - Paramsets
 - Files
 - Ping
 - Plugins / Stats
 - Query
 - Replication
 - Schema
 - Segments info

Request-Handler (qt)

/select

common

q

q.op

OR

fq

price:[* TO 25]

sort

start, rows

0 10

fl

name,category

df

paramset(s)

Select paramset(s)...

wt

☒ indent on

☐ debugQuery

```
{
  "responseHeader": {
    "status": 0,
    "QTime": 1,
    "params": {
      "q": "*",
      "indent": "true",
      "fl": "name,category",
      "q.op": "OR",
      "fq": "price:[* TO 25]",
      "spatial": "true",
      "useParams": "",
      "_": "1706931696452"
    }
  },
  "response": {
    "numFound": 1,
    "start": 0,
    "numFoundExact": true,
    "docs": [
      {
        "name": "Product B",
        "category": "Books"
      }
    ]
  }
}
```

Display data in xml format.

localhost:8983/solr/#/mynewcollection/query?q=*&q.op=OR&indent=true&wt=xml&useParams=

Solr

- Dashboard
- Logging
- Security
- Core Admin
- Java Properties
- Thread Dump
- mynewcollection
 - Overview
 - Analysis
 - Documents
 - Paramsets
 - Files
 - Ping
 - Plugins / Stats
 - Query
 - Replication
 - Schema
 - Segments info

q.op

OR

fq

sort

start, rows

0 10

fl

df

paramset(s)

Select paramset(s)...

wt

xml

☒ indent on

☐ debugQuery

deftype

☐ hl

☐ facet

☐ spatial

☐ spellcheck

Raw Query Parameters

```
<lst name="params">
  <str name="q">*</str>
  <str name="indent">true</str>
  <str name="q.op">OR</str>
  <str name="wt">xml</str>
  <str name="useParams"/>
  <str name="_">1706969750459</str>
</lst>
</lst>
<result name="response" numFound="2" start="0" numFoundExact="true">
  <doc>
    <str name="id">1</str>
    <arr name="name">
      <str>Product A</str>
    </arr>
    <arr name="category">
      <str>Electronics</str>
    </arr>
    <arr name="price">
      <long>100</long>
    </arr>
    <long name="_version_">1789846779737407488</long></doc>
  <doc>
    <str name="id">2</str>
    <arr name="name">
      <str>Product B</str>
    </arr>
    <arr name="category">
      <str>Books</str>
    </arr>
    <arr name="price">
      <long>20</long>
    </arr>
    <long name="_version_">1789846779825487872</long></doc>
</result>
```

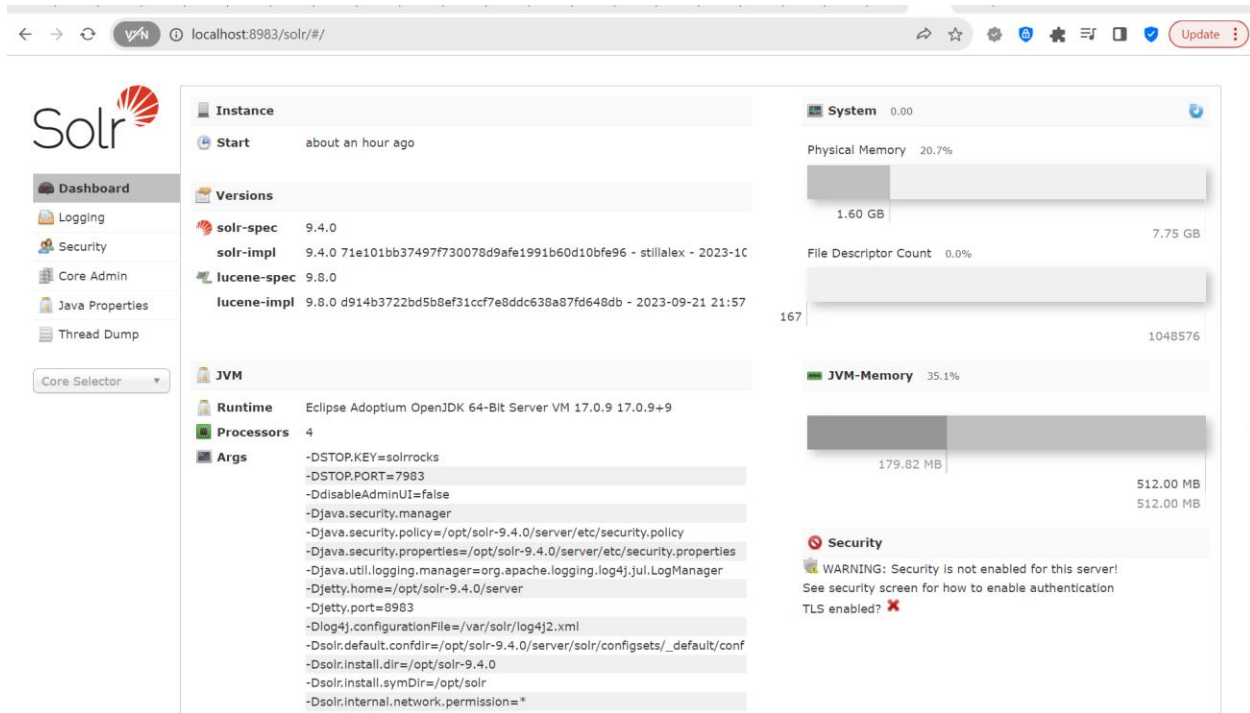
Select category, price for name:"Product B" and xml output.

The screenshot shows the Solr Admin interface at localhost:8983. The left sidebar contains navigation links: Dashboard, Logging, Security, Core Admin, Java Properties, Thread Dump, mynewcollection (selected), Overview, Analysis, Documents, Paramsets, Files, Ping, Plugins / Stats, Query, Replication, Schema, and Segments info. The main panel is divided into two sections. The top section, titled 'common', contains query configuration fields: 'q' (set to '*:*'), 'q.op' (set to 'AND'), 'fq' (set to 'name:"Product B"'), 'sort' (empty), 'start, rows' (set to 0, 10), 'fl' (empty), 'df' (set to 'category,price'), 'paramset(s)' (set to 'Select paramset(s)...'), 'wt' (set to 'xml'), 'indent on' (checked), 'debugQuery' (unchecked), 'defType' (set to 'lucene'), 'hl' (unchecked), 'facet' (unchecked), and 'facet.sort' (empty). The bottom section displays the XML response for the query. The response is a SOAP-style XML with a 'responseHeader' containing status, QTime, and parameters, and a 'result' containing a document with fields 'id', 'name', 'category', and 'price'.

Create a new collection in Solr mynewcollection_1.

```
solr@6a404867c512:/opt/solr-9.4.0$ /opt/solr/bin/solr create -c mynewcollection_1
WARNING: Using _default configset with data driven schema functionality. NOT RECOMMENDED for production use.
To turn off: bin/solr config -c mynewcollection_1 -p 8983 -action set-user-property -property update.autoCreateFields -value false
Created new core 'mynewcollection_1'
solr@6a404867c512:/opt/solr-9.4.0$
```

Verify that 'mynewcollection_1' has been successfully created



Generate JSON data and add it to mycollection_1.

Structure of result.json file

```
echo '['
```

```
{ "student_id": "1", "name": "John", "Grade": "A", "Result": "Pass"},  
{ "student_id": "2", "name": "Ron", "Grade": "A+", "Result": "Pass"},  
{ "student_id": "3", "name": "Andrew", "Grade": "B", "Result": "Pass"},  
{ "student_id": "4", "name": "Sophia", "Grade": "B", "Result": "Pass"},  
{ "student_id": "5", "name": "Jason", "Grade": "F", "Result": "Fail"},  
{ "student_id": "6", "name": "Mady", "Grade": "A", "Result": "Pass"},  
{ "student_id": "7", "name": "Grayson", "Grade": "F", "Result": "Fail" }
```

```
]' > /tmp/result.json
```

```

Created new core 'mynewcollection_1'
solr@6a404867c512:/opt/solr-9.4.0$ echo '[
{"student_id":1,"name":"John","Grade":"A","Result":"Pass"},
{"student_id":2,"name":"Ron","Grade":"A+","Result":"Pass"},
{"student_id":3,"name":"Andrew","Grade":"B","Result":"Pass"},
{"student_id":4,"name":"Sophia","Grade":"B","Result":"Pass"},
{"student_id":5,"name":"Jason","Grade":"F","Result":"Fail"},
{"student_id":6,"name":"Mady","Grade":"A","Result":"Pass"},
{"student_id":7,"name":"Grayson","Grade":"F","Result":"Fail"}
]' > /tmp/result.json
solr@6a404867c512:/opt/solr-9.4.0$ /opt/solr/bin/post -c mynewcollection_1 /tmp/result.json
The bin/post script is deprecated in favour of the bin/solr post command. Please update your scripts.
/opt/java/openjdk/bin/java -classpath /opt/solr/server/solr-webapp/webapp/WEB-INF/lib/solr-core-9.4.0.jar -Dauto=yes -Dc=mynewcollection_1 -Ddata=files org.a
pache.solr.cli.SimplePostTool /tmp/result.json
SimplePostTool version 9.4.0
Posting files to [base] url http://localhost:8983/solr/mynewcollection_1/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ots,rtf,htm,html,txt,log
POSTing file result.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/mynewcollection_1/update...
Time spent: 0:00:00.304
solr@6a404867c512:/opt/solr-9.4.0$

```

Query your collection from command line and web interface :

```

chitramoy@bigdata:~/dsac650-infra/bellevue-bigdata/solr$ docker exec -it solr_solr_1 bash
solr@6a404867c512:/opt/solr-9.4.0$ curl "http://localhost:8983/solr/mynewcollection_1/select?q=*:*"
{
  "responseHeader":{
    "status":0,
    "QTime":1,
    "params":{
      "q":"*:*"
    }
  },
  "response":{
    "numFound":7,
    "start":0,
    "numFoundExact":true,
    "docs":[
      {
        "student_id":1,
        "name":["John"],
        "Grade":["A"],
        "Result":["Pass"],
        "id":"ca971317-fde8-4a3e-a29e-36c43622453c",
        "_version_":1789849572365828096
      },
      {
        "student_id":2,
        "name":["Ron"],
        "Grade":["A+"],
        "Result":["Pass"],
        "id":"bd9a6560-6b18-412b-9651-1208b93aca5b",
        "_version_":1789849572371070976
      },
      {
        "student_id":3,
        "name":["Andrew"],
        "Grade":["B"],
        "Result":["Pass"],
        "id":"53185e92-53f6-49d1-a749-eff9f63569b9",
        "_version_":1789849572372119552
      },
      {
        "student_id":4,
        "name":["Sophia"],
        "Grade":["B"],
        "Result":["Pass"],
        "id":"2439635f-00ac-4adf-a00d-15aa27d2996b",
        "_version_":1789849572373168128
      }
    ]
  }
}

```

```

      "Grade":["A"],
      "Result":["Pass"],
      "id":"ca971317-fde8-4a3e-a29e-36c43622453c",
      "_version_":1789849572365828096
    },{
      "student_id":2,
      "name":["Ron"],
      "Grade":["A+"],
      "Result":["Pass"],
      "id":"bd9a6560-6b18-412b-9651-1208b93aca5b",
      "_version_":1789849572371070976
    },{
      "student_id":3,
      "name":["Andrew"],
      "Grade":["B"],
      "Result":["Pass"],
      "id":"53185e92-53f6-49d1-a749-eff9f63569b9",
      "_version_":1789849572372119552
    },{
      "student_id":4,
      "name":["Sophia"],
      "Grade":["B"],
      "Result":["Pass"],
      "id":"2439635f-00ac-4adf-a00d-15aa27d2996b",
      "_version_":1789849572373168128
    },{
      "student_id":5,
      "name":["Jason"],
      "Grade":["F"],
      "Result":["Fail"],
      "id":"226a7293-51ee-44d5-87e7-9a27bf33b4ff",
      "_version_":1789849572374216704
    },{
      "student_id":6,
      "name":["Mady"],
      "Grade":["A"],
      "Result":["Pass"],
      "id":"05941724-5b8e-490b-86d5-b427525a180b",
      "_version_":1789849572374216705
    },{
      "student_id":7,
      "name":["Grayson"],
      "Grade":["F"],
      "Result":["Fail"],
      "id":"dbc961c0-0a56-47ac-a470-0ae35dce843",
      "_version_":1789849572375265280
    }
  ]
}
}
solr@6a404867c512:/opt/solr-9.4.0$

```

Query your collection from command line and web interface :

Select where name is Ron and select * where Grade = 'B'.

```

solr@6a404867c512:/opt/solr-9.4.0$ curl "http://localhost:8983/solr/mynewcollection_1/select?q=name:Ron"
{
  "responseHeader":{
    "status":0,
    "QTime":1,
    "params":{
      "q":"name:Ron"
    }
  },
  "response":{
    "numFound":1,
    "start":0,
    "numFoundExact":true,
    "docs":[{
      "student_id":2,
      "name":["Ron"],
      "Grade":["A+"],
      "Result":["Pass"],
      "id":"bd9a6560-6b18-412b-9651-1208b93aca5b",
      "_version_":1789849572371070976
    }
  ]
}
}
solr@6a404867c512:/opt/solr-9.4.0$ curl "http://localhost:8983/solr/mynewcollection_1/select?q=Grade:B"
{
  "responseHeader":{
    "status":0,
    "QTime":2,
    "params":{
      "q":"Grade:B"
    }
  },
  "response":{
    "numFound":2,
    "start":0,
    "numFoundExact":true,
    "docs":[{
      "student_id":3,
      "name":["Andrew"],
      "Grade":["B"],
      "Result":["Pass"],
      "id":"53185e92-53f6-49d1-a749-eff9f63569b9",
      "_version_":1789849572372119552
    },{
      "student_id":4,
      "name":["Sophia"],
      "Grade":["B"],
      "Result":["Pass"],
      "id":"2439635f-00ac-4adf-a00d-15aa27d2996b",
      "_version_":1789849572373168128
    }
  ]
}
}

```

Query your collection from web interface:


Select name, Grade, Result from mynewcollection_1 where Result:"Pass" and extract in csv format.

The screenshot shows the Solr Admin interface. On the left is a sidebar with navigation links: Dashboard, Logging, Security, Core Admin, Java Properties, Thread Dump, mynewcollection_1 (selected), Overview, Analysis, Documents, Paramsets, Files, Ping, Plugins / Stats, Query (selected), Replication, Schema, and Segments Info. The main panel is titled 'Request-Handler (qt)' and shows the following configuration:
 - common: /select
 - q: *:*
 - q.op: OR
 - fq: Result:"Pass"
 - sort:
 - start, rows: 0, 10
 - fl: name,Grade,Result
 - df:
 - paramset(s): Select paramset(s)...
 - wt: csv
 - Indent on: ☒
 - debugQuery: ☐
 The URL bar shows: localhost:8983/solr/#/mynewcollection_1/query?q=*:*&q.op=OR&indent=true&fl=name,Grade,Result&facet=true&...
 The response area on the right shows the following data:
 name,Grade,Result
 John,A,Pass
 Ron,A+,Pass
 Andrew,B,Pass
 Sophia,B,Pass
 Mady,A,Pass

Select name, Grade, Result from mynewcollection_1 where Grade = 'A' and extract in python format (df).

The screenshot shows the Solr Admin interface with the following configuration:
 - common: /select
 - q: *:*
 - q.op: OR
 - fq: Grade:A
 - sort:
 - start, rows: 0, 10
 - fl: name,Grade,Result
 - df:
 - paramset(s): Select paramset(s)...
 - wt: python
 - Indent on: ☒
 - debugQuery: ☐
 The URL bar shows: localhost:8983/solr/#/mynewcollection_1/query?q=*:*&q.op=OR&indent=true&fl=name,Grade,Result&wt=python&spatial...
 The response area on the right shows a JSON response:
 {
 'responseHeader': {
 'status': 0,
 'QTime': 1,
 'params': {
 'q': '*:*',
 'indent': 'true',
 'fl': 'name,Grade,Result',
 'q.op': 'OR',
 'fq': 'Grade:A',
 'spatial': 'true',
 'wt': 'python',
 'useParams': ''
 },
 '_': '1706934225228'
 },
 'response': { 'numFound': 3, 'start': 0, 'numFoundExact': true, 'docs': [
 {
 'name': ['John'],
 'Grade': ['A'],
 'Result': ['Pass']
 },
 {
 'name': ['Ron'],
 'Grade': ['A+'],
 'Result': ['Pass']
 },
 {
 'name': ['Mady'],
 'Grade': ['A'],
 'Result': ['Pass']
 }
]
 } }

Grade wise count from from mynewcollection_1.



The screenshot shows the Solr Admin interface. On the left is a navigation menu with options: Dashboard, Logging, Security, Core Admin, Java Properties, Thread Dump, mynewcollection (selected), Overview, Analysis, Documents, Paramsets, Files, Ping, Plugins / Stats, Query (selected), Replication, Schema, and Segments info. The main content area is titled 'localhost:8983/solr/#/mynewcollection_1/query?q=*:*&q.op=AND&indent=true&facet=true&facet.field=Grade&wt=json...'. It contains a 'Query' input field with the text 'spatial'. Below this are several facet configuration options: 'facet.contains' (empty), 'facet.contains.ignoreCase' (checkbox), 'facet.limit' (empty), 'facet.matches' (empty), 'facet.sort' (dropdown set to '-----'), 'facet.mincount' (empty), 'facet.missing' (checkbox), and 'json.facet' (help icon). The right side of the interface displays the JSON response of the query, which includes facet counts and facet fields for the 'Grade' field. The JSON is formatted with indentation.