HASH AGILE TECHNOLOGIES

TASK- 2

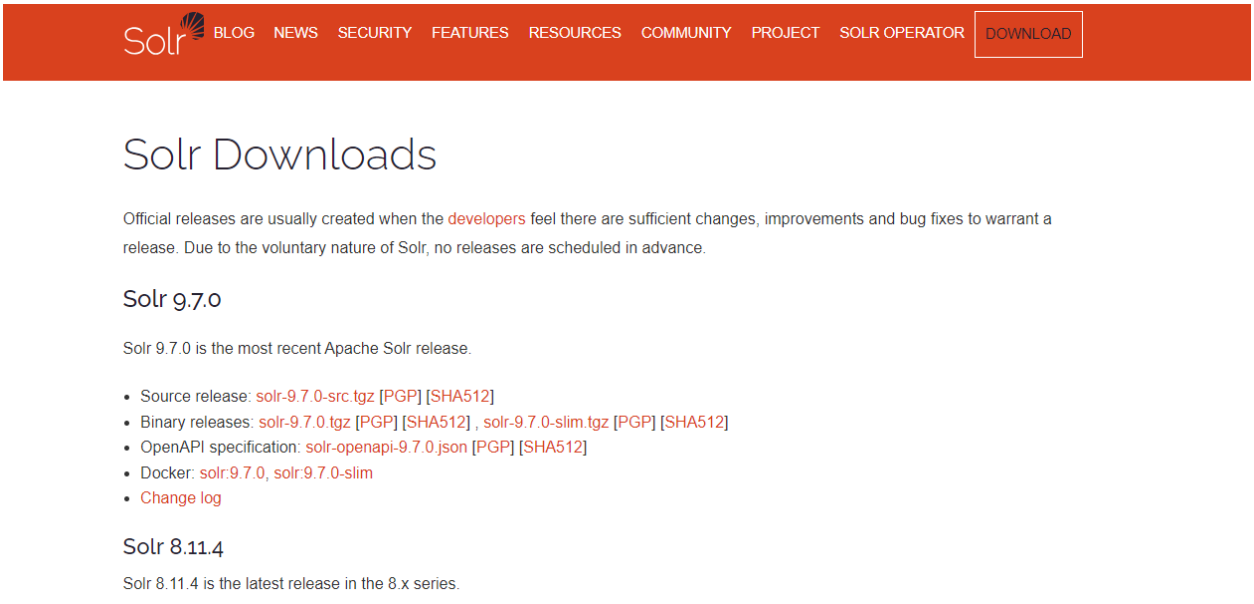
TASK GIVEN :

1. Read about Apache Solr: Solr Tutorial.
2. Install Solr on your local machine.
3. Create a collection in Solr.
4. Index the Employee data from https://www.kaggle.com/datasets/williamlucas0/employee-sample-data

STEPS USED :

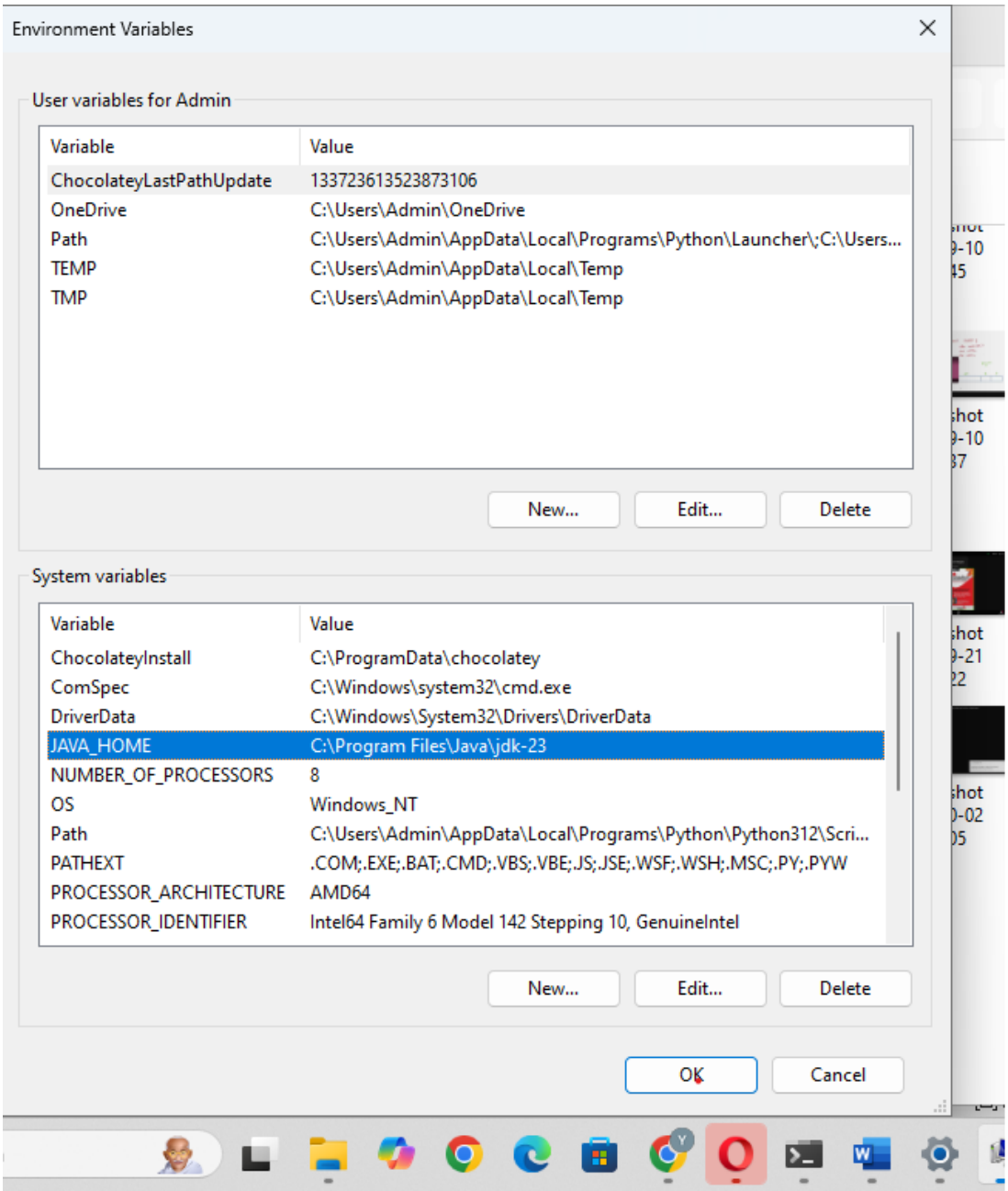
1) INSTALLATION

Downloaded zip file (Solr.9.7.0.tgz) from Apache Solr Downloader



2) Next , Installed Java JDK 23 for establishing connection in the system

3) JDK path setup



Java Environment Variables is created as JAVA_HOME and path has been set up

4) SOLR START

To start the solr the start command should be initialized in command prompt inside the solr/bin directory .

COMMAND :

C:\Users\Admin\Downloads\solr-9.7.0\solr-9.7.0\bin>solr start

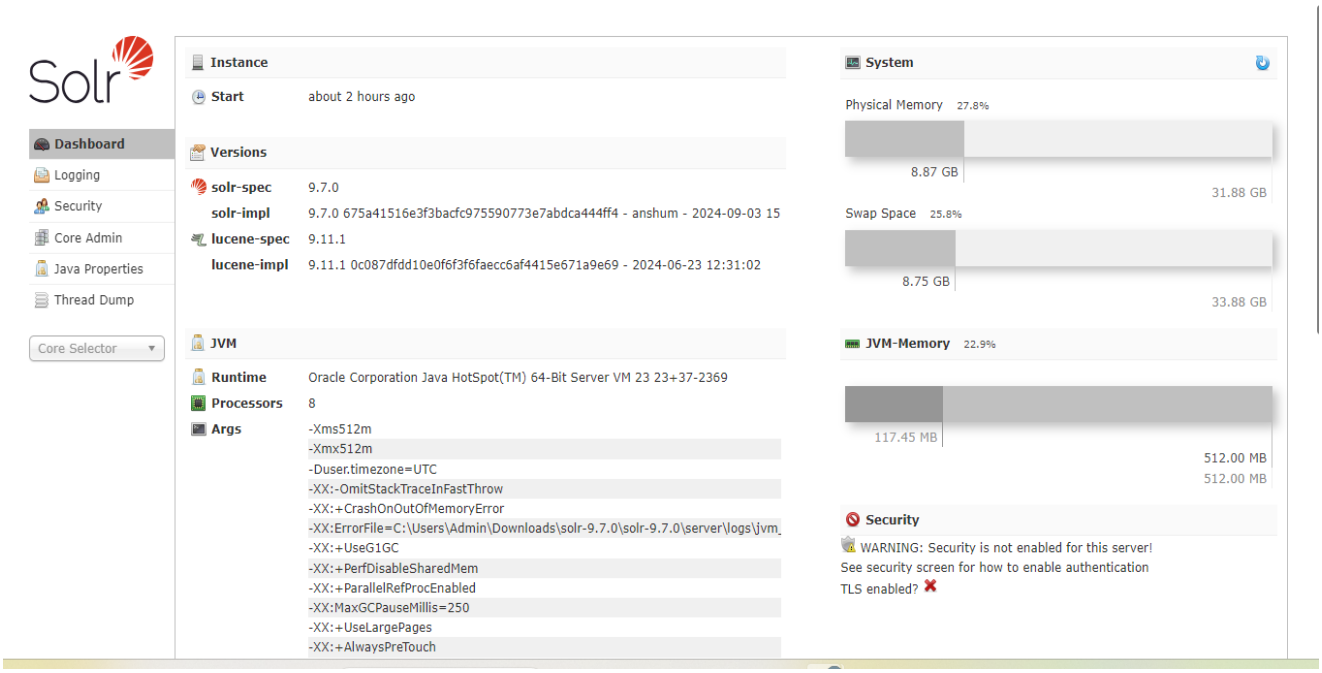
```
C:\Users\Admin\Downloads\solr-9.7.0\solr-9.7.0\bin>solr start
Java 23 detected. Enabled workaround for SOLR-16463
Java HotSpot(TM) 64-Bit Server VM warning: JVM cannot use large page memory because it does not have enough privilege to
 lock pages in memory.
WARNING: A command line option has enabled the Security Manager
WARNING: The Security Manager is deprecated and will be removed in a future release
Failed to parse command-line arguments due to: Unrecognized option: --max-wait-secs

usage: bin/solr status [-maxWaitSecs <SECS>] [--solr-url <URL>]

List of options:
 -maxWaitSecs <SECS>   Wait up to the specified number of seconds to see Solr running.
    --solr-url <URL>   Address of the Solr Web application, defaults to: http://localhost:8983.
```

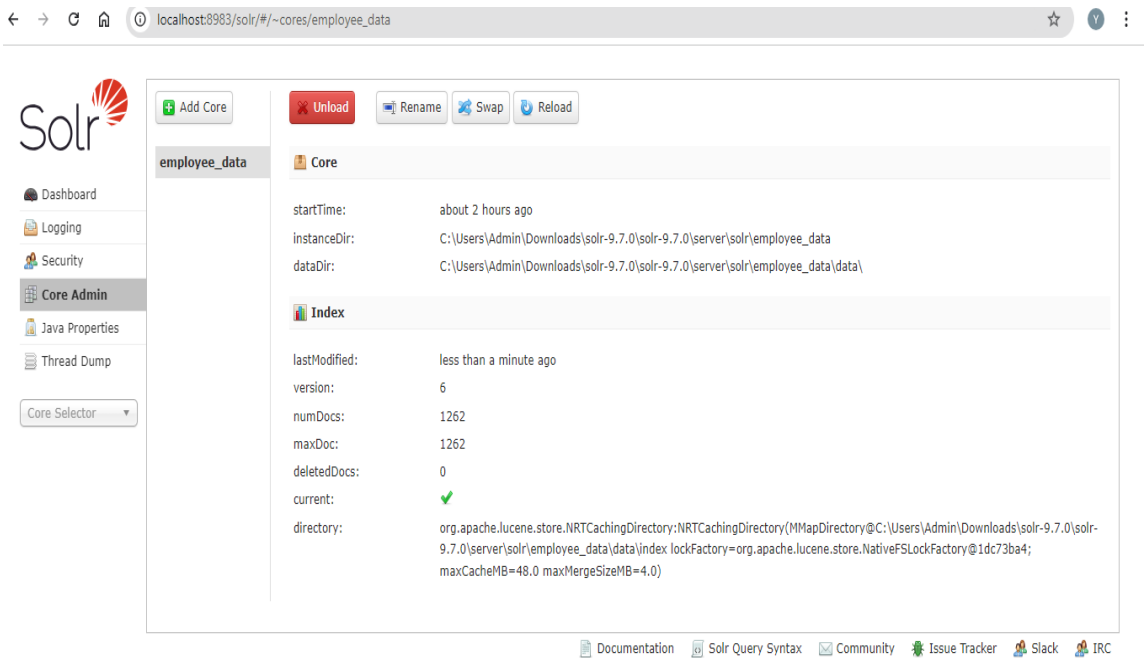
5) LOCAL HOSTING

After the above step , the local machine setup will be initialized will be in the **local host :8393 port .**



6) CREATING COLLECTION

To create a collection , **create-c employee_data**(name of the collection) command is used .



COMMAND

C:\Users\Admin\Downloads\solr9.7.0\solr-9.7.0\bin>solr create -c employee_data

Here , the employee data collection is created inside the core admin

7) CONVERTING CSV FILE TO JSON FILE

Next step is to convert the csv file to json format , for that I have used python script for automating

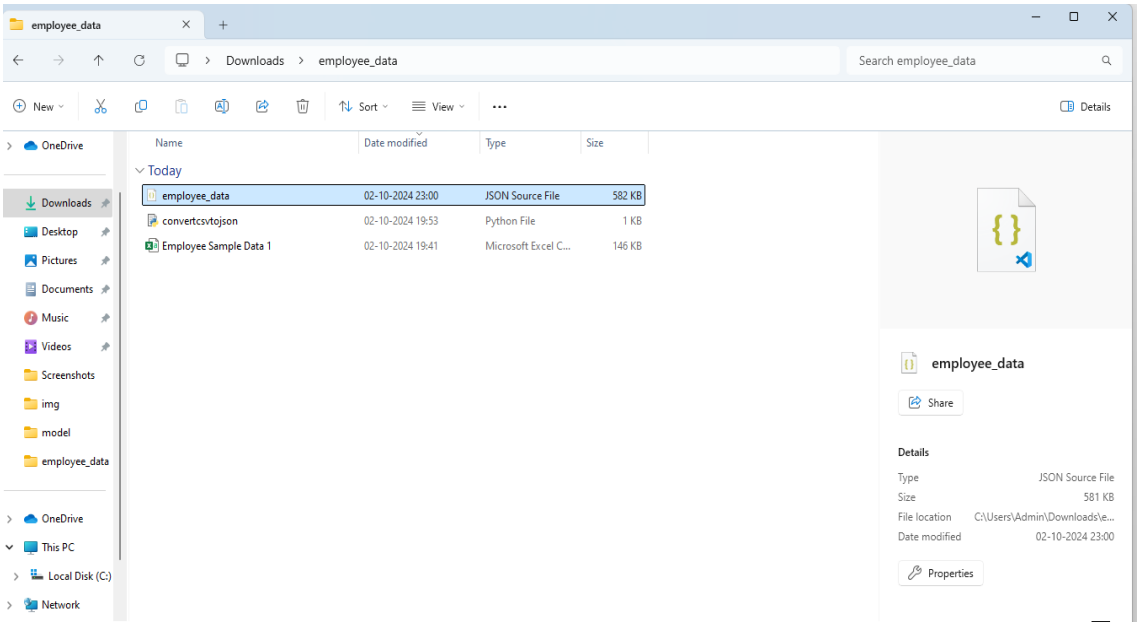
CODE :

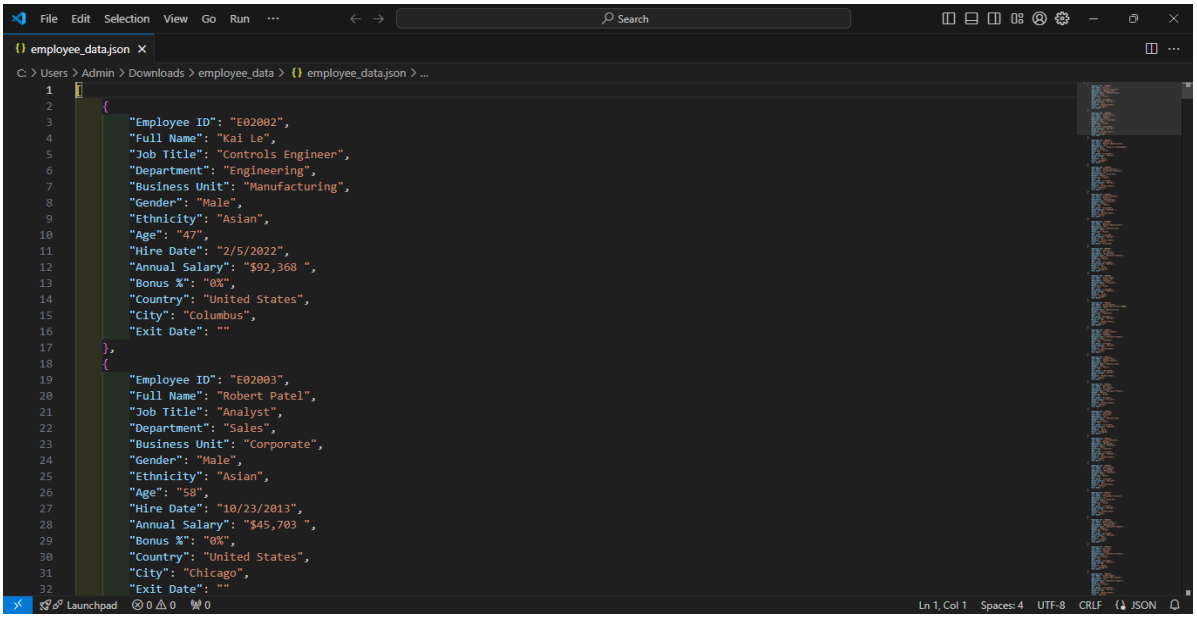
```
import csv
import json
# Input and output file paths
csv_file = 'Employee Sample Data 1.csv'  # Name of the csv file
json_file = 'employee_data.json'
# Read CSV and convert to JSON
data = []
with open(csv_file, newline='', encoding='ISO-8859-1') as csvfile:  # Change encoding here
    reader = csv.DictReader(csvfile)
    for row in reader:
        data.append(row)
# Write JSON data to file
with open(json_file, 'w', encoding='utf-8') as jsonfile:
    json.dump(data, jsonfile, indent=4)
```

8) EXECUTION

Executed this code as .py file using pythonIDLE ,in the path where the excel file [Employee Test Data 1] is located. Which was downloaded from Kaggle using the provided link.

After executing this script the employee_data.json file will be created



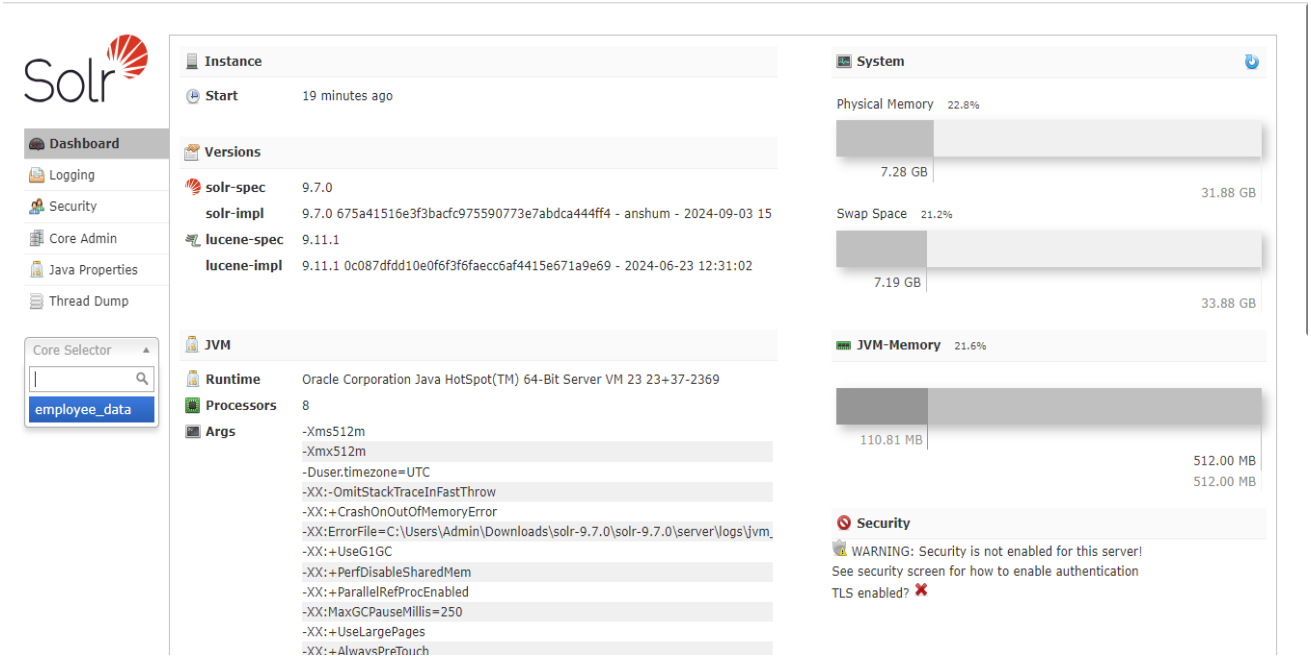


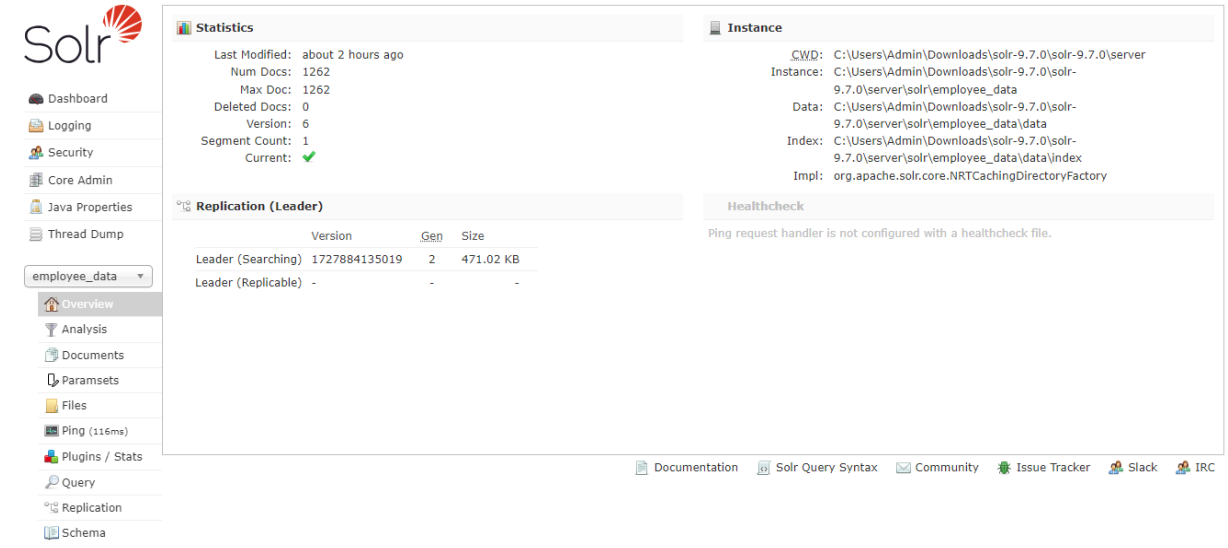
9) POSTING THE JSONFILE TO COLLECTION

Using the **solr.cmd**: to post the JSON file in to the collection

Command :

C:\Users\Admin\Downloads\solr-9.7.0\solr-9.7.0\bin\solr.cmd post -c employee_dataC:\Users\Admin\Downloads\employee_data\employee_data.json



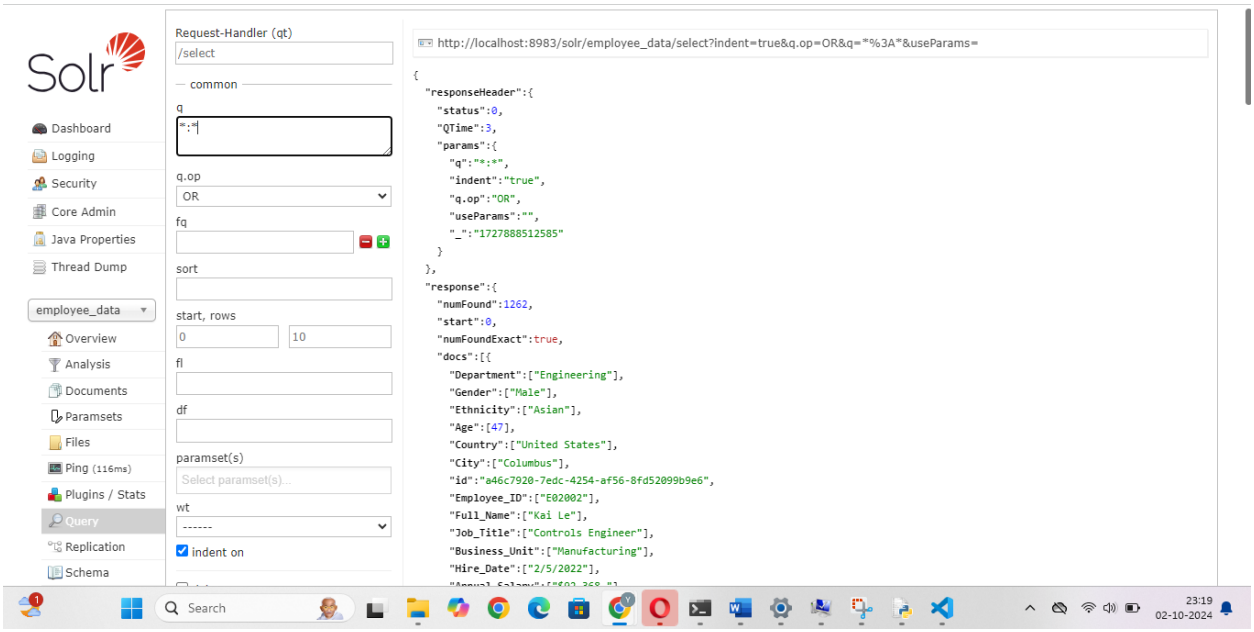


9) EXECUTING QUERIES

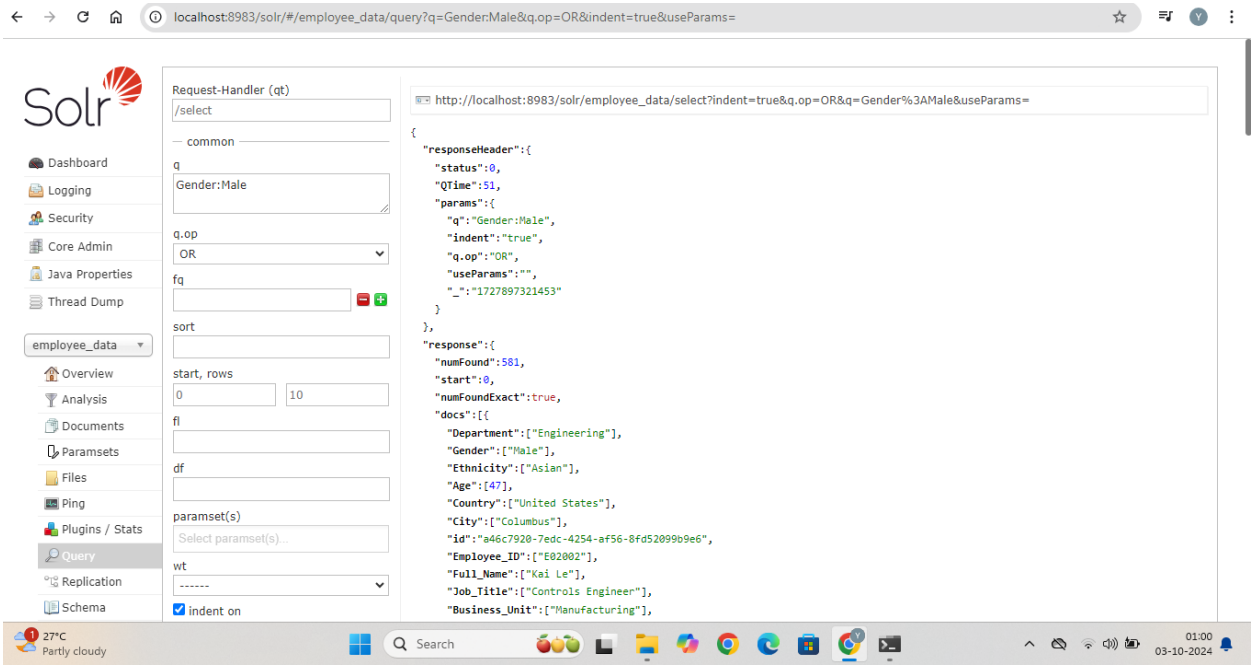
Getting all the values in json format

Query:

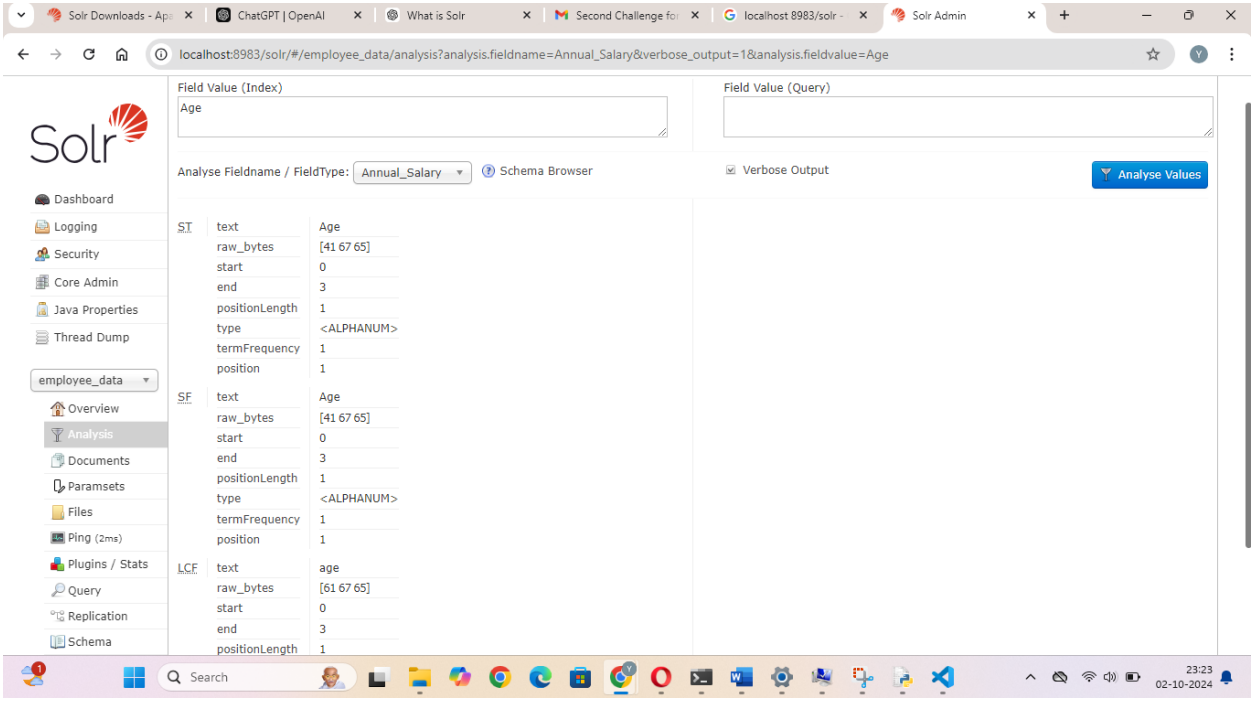
:



Query to obtain male employees - **Gender:Male**

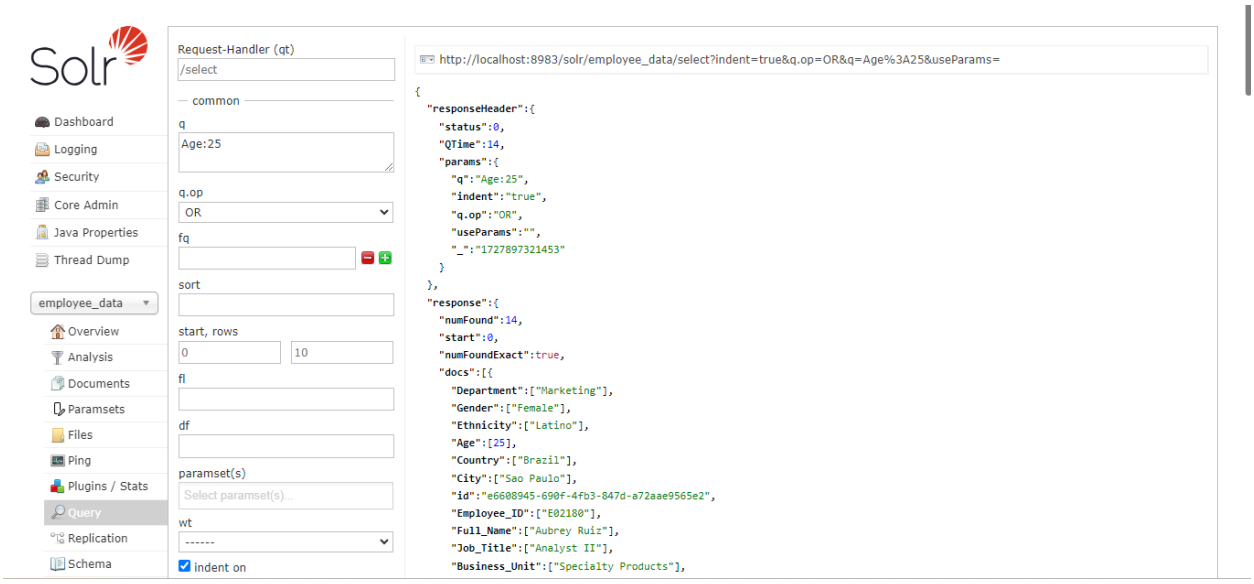


Analysis of Age Column



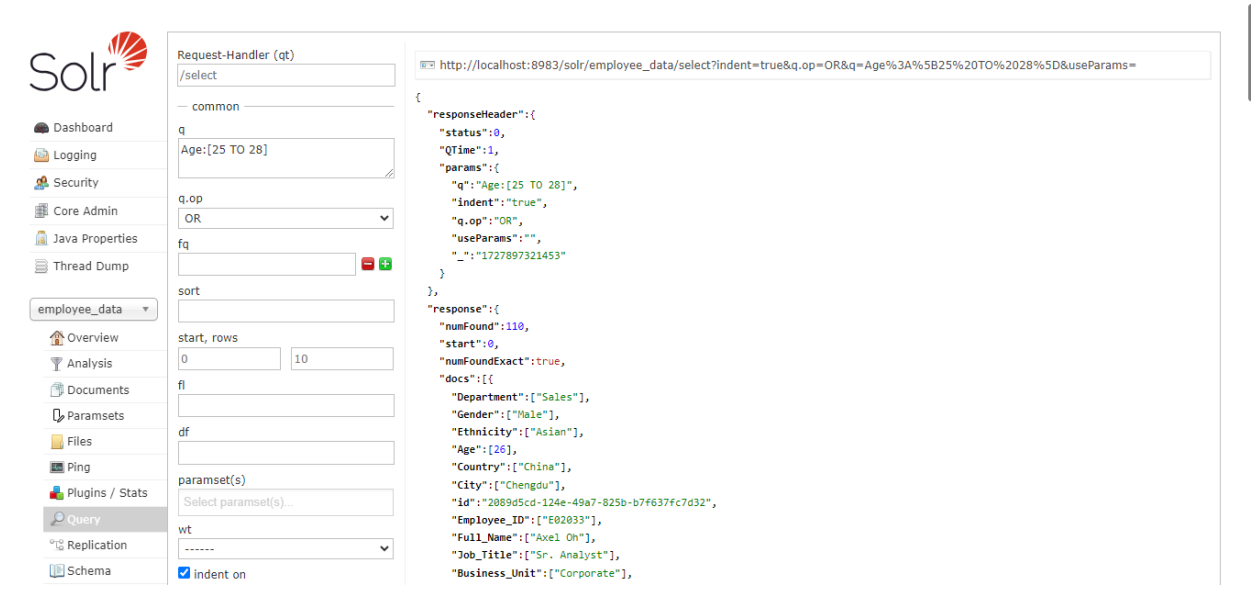
Query to get data's where age=25

Age:25



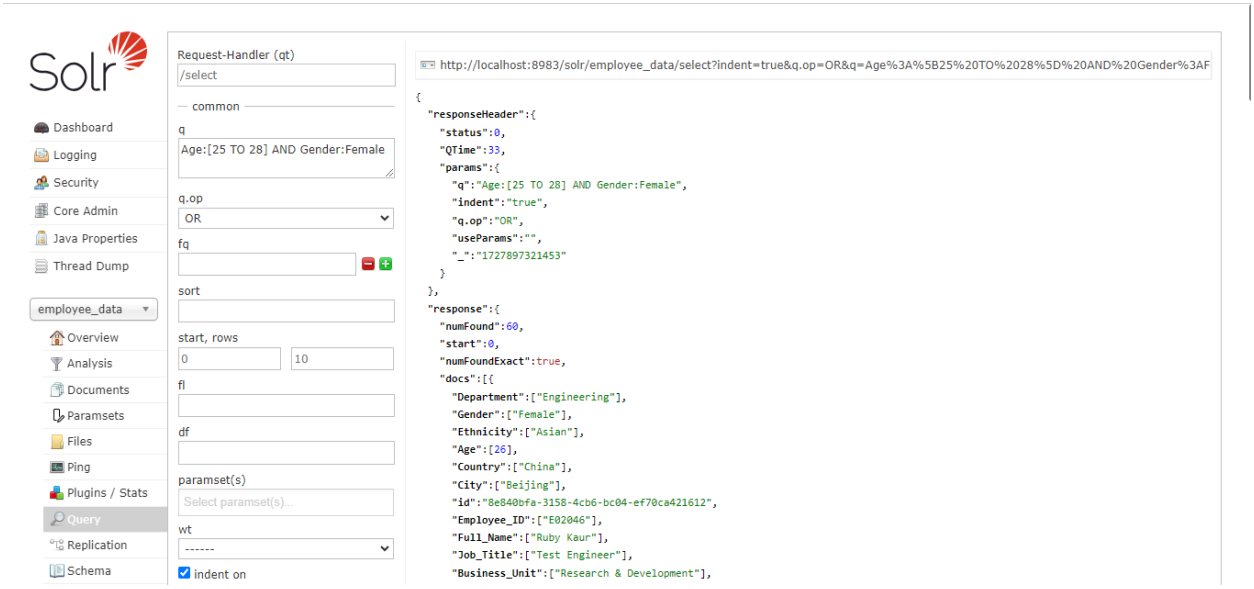
Query to obtain data between Age 25 to 28

Age [25 TO 28]



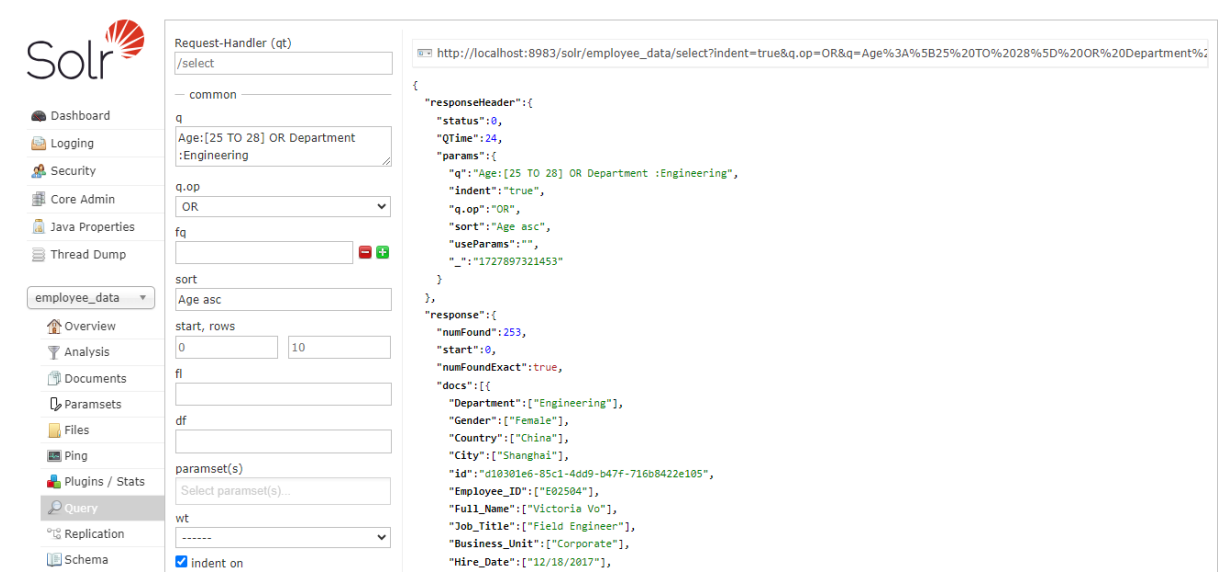
AND

Age:[25 TO 28] AND Gender:Female



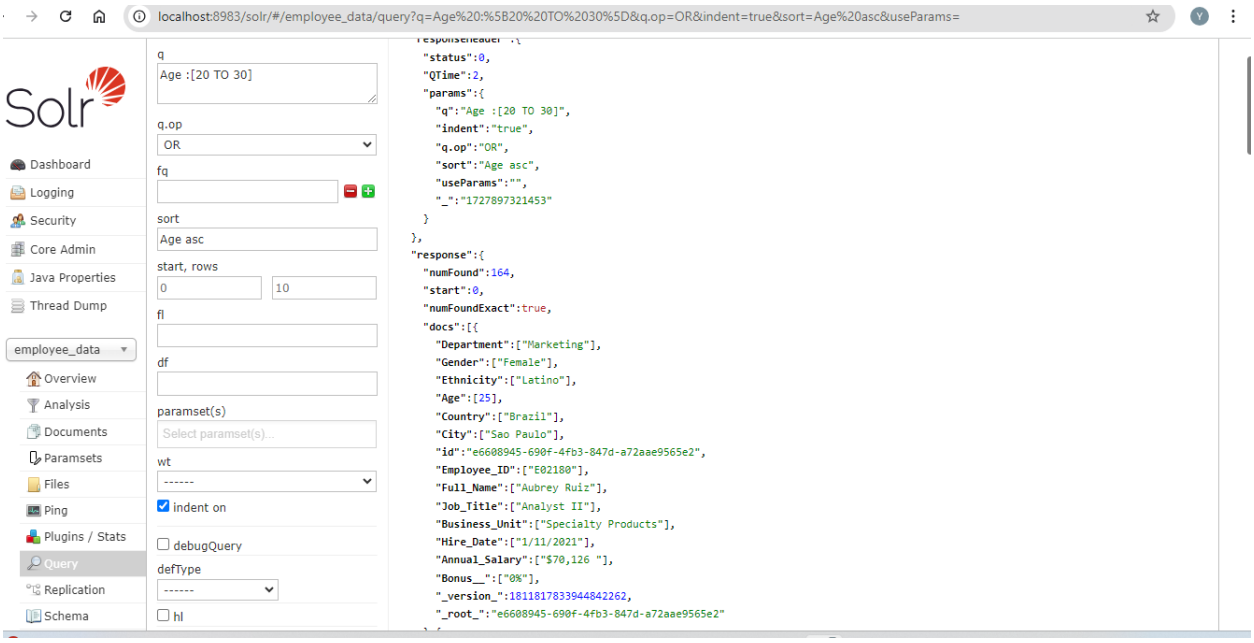
OR

Age:[25 TO 28] OR Department :Engineering


Solr
Request-Handler (qt)
/select
common
q
Age:[25 TO 28] OR Department :Engineering
q.op
OR
fq
sort
Age asc
start, rows
0
10
fl
df
paramset(s)
Select paramset(s)...
wt
------
indent on
Dashboard
Logging
Security
Core Admin
Java Properties
Thread Dump
employee_data
Overview
Analysis
Documents
Paramsets
Files
Ping
Plugins / Stats
Query
Replication
Schema
http://localhost:8983/solr/employee_data/select?indent=true&q.op=OR&q=Age%3A%5B25%20TO%2028%5D%20OR%20Department%2
{
"responseHeader":{
"status":0,
"QTime":24,
"params":{
"q":"Age:[25 TO 28] OR Department :Engineering",
"indent":"true",
"q.op":"OR",
"sort":"Age asc",
"useParams":"",
"_":"1727897321453"
}
},
"response":{
"numFound":253,
"start":0,
"numFoundExact":true,
"docs":[{
"Department":["Engineering"],
"Gender":["Female"],
"Country":["China"],
"City":["Shanghai"],
"id":"d10301e6-85c1-4dd9-b47f-716b8422e105",
"Employee_ID":["E02504"],
"Full_Name":["Victoria Vo"],
"Job_Title":["Field Engineer"],
"Business_Unit":["Corporate"],
"Hire_Date":["12/18/2017"],

SORT

Age asc


localhost:8983/solr/#/employee_data/query?q=Age%20:%5B20%20TO%2030%5D&q.op=OR&indent=true&sort=Age%20asc&useParams=
Solr
q
Age :[20 TO 30]
q.op
OR
fq
sort
Age asc
start, rows
0
10
fl
df
paramset(s)
Select paramset(s)...
wt
------
indent on
debugQuery
defType
------
hl
Dashboard
Logging
Security
Core Admin
Java Properties
Thread Dump
employee_data
Overview
Analysis
Documents
Paramsets
Files
Ping
Plugins / Stats
Query
Replication
Schema
"status":0,
"QTime":2,
"params":{
"q":"Age :[20 TO 30]",
"indent":"true",
"q.op":"OR",
"sort":"Age asc",
"useParams":"",
"_":"1727897321453"
}
},
"response":{
"numFound":164,
"start":0,
"numFoundExact":true,
"docs":[{
"Department":["Marketing"],
"Gender":["Female"],
"Ethnicity":["Latino"],
"Age":[25],
"Country":["Brazil"],
"City":["Sao Paulo"],
"id":"e6608945-690f-4fb3-847d-a72aae9565e2",
"Employee_ID":["E02180"],
"Full_Name":["Aubrey Ruiz"],
"Job_Title":["Analyst II"],
"Business_Unit":["Specialty Products"],
"Hire_Date":["1/11/2021"],
"Annual_Salary":["$70,126 "],
"Bonus__":["0%"],
"_version_":1811817833944842262,
"_root_":"e6608945-690f-4fb3-847d-a72aae9565e2"