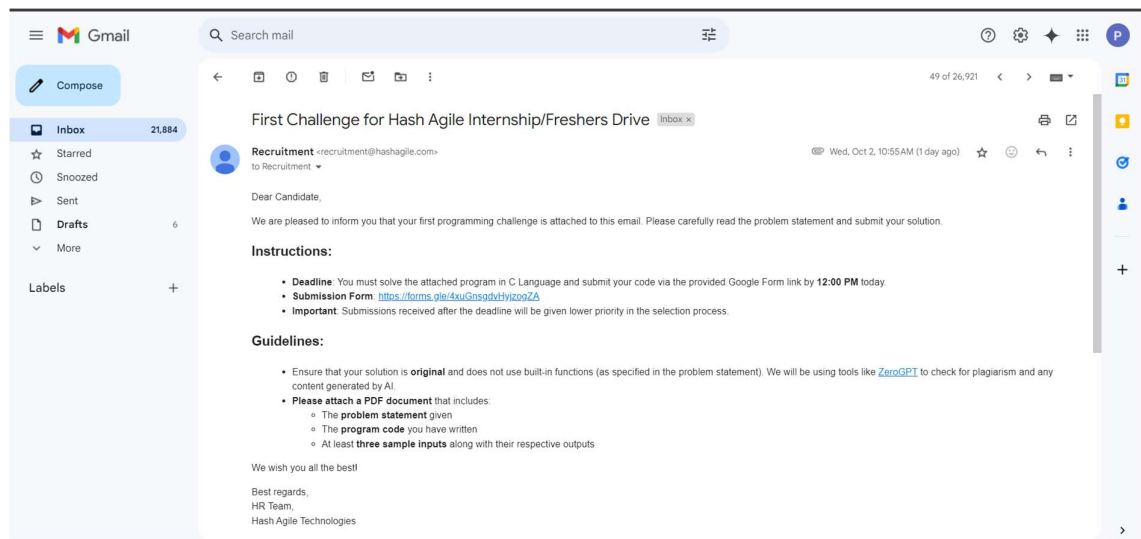


Submission:+

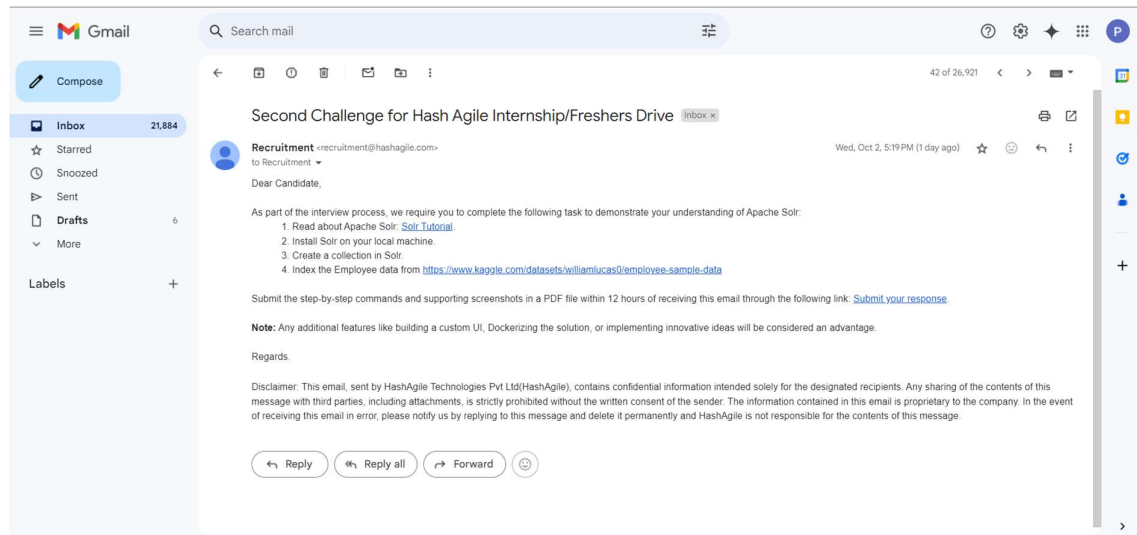
- Full Name: **Poulomi Bhattacharya**
- Email: poulomibhattacharya03@gmail.com
- Selfie:



- **First Task Email Screenshot:**



- **Second Task Email Screenshot:**



- GitHub URL for Round 1: <https://github.com/poulomi-03/hashagile-assignment-poulomi-bhattacharya.git>
- GitHub URL for Assignment:

Second round challenge

Problem Statement:

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>

Step 01: Since Apache Solr requires Java 1.8 or higher, it's important to verify if Java is installed and set up correctly.

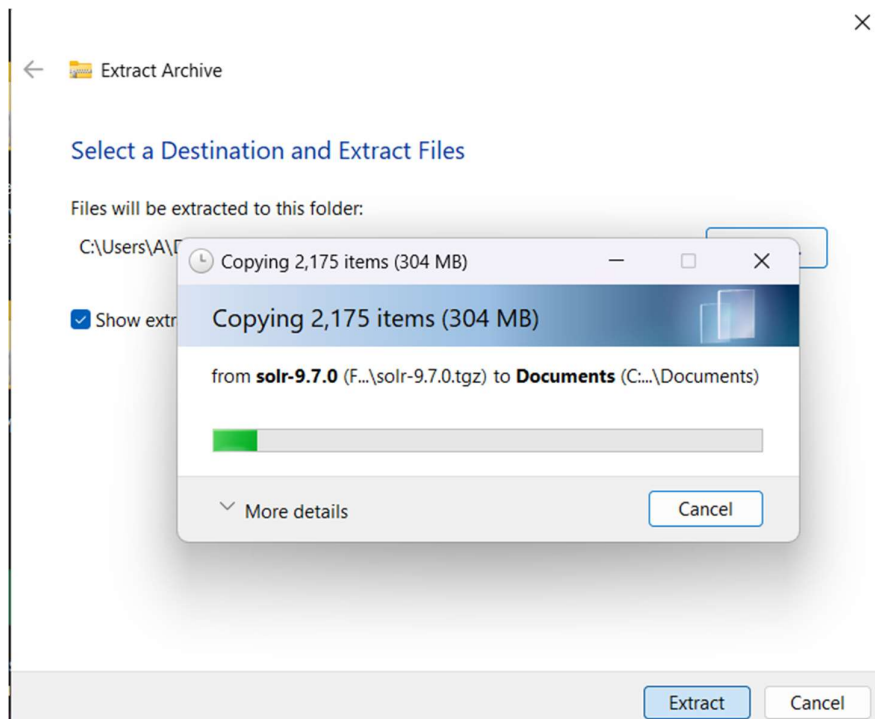
- Open a command prompt (Windows) or terminal (Linux/macOS) and run the following command:

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.

C:\Users\A>java -version
java version "20.0.1" 2023-04-18
Java(TM) SE Runtime Environment (build 20.0.1+9-29)
Java HotSpot(TM) 64-Bit Server VM (build 20.0.1+9-29, mixed mode, sharing)

C:\Users\A>
```

Step02: Download and Extract Solr



Step 03: Navigate to the Solr bin directory where you extracted Solr.

- **Windows (PowerShell)/Command Line**

```
Command Prompt
C:\Users\A\cd Documents
C:\Users\A\Documents>cd solr-9.7.0
C:\Users\A\Documents\solr-9.7.0>cd bi
The system cannot find the path specified.
C:\Users\A\Documents\solr-9.7.0>cd bin
C:\Users\A\Documents\solr-9.7.0\bin>solr start
```

Step 04: Create a Collection in Solr

To create a collection in Solr, use the create command. This will create a new collection named `employee_data`.

- Run the following command in your terminal or PowerShell/CMD:

```
Command Prompt
C:\Users\A\cd Documents
C:\Users\A\Documents>cd solr-9.7.0
C:\Users\A\Documents\solr-9.7.0>cd bi
The system cannot find the path specified.
C:\Users\A\Documents\solr-9.7.0>cd bin
C:\Users\A\Documents\solr-9.7.0\bin>solr start

ERROR: Process 28576 is already listening on port 8983. If this is Solr, please stop it first before starting (or use restart). If this is not Solr, then please choose a different port using -p PORT

C:\Users\A\Documents\solr-9.7.0\bin>solr create -c employee_data
WARNING: Using default configset with data driven schema functionality. NOT RECOMMENDED for production use.
To turn off: bin\solr config -c employee_data -p 8983 -action set-user-property -property update.autoCreateFields -value false
Option '-solrurl', '-solrurl': Deprecated for removal since 9.7: Use --solr-url instead
WARNING: URIs provided to this tool needn't include Solr's context-root (e.g. "/solr/"). Such URIs are deprecated and support for them will be removed in a future release. Correcting from [http://localhost:8983/solr] to [http://localhost:8983/].
Option '-confdir', '-confdir': Deprecated for removal since 9.7: Use --conf-dir instead
Option '-confdir', '-confdir': Deprecated for removal since 9.7: Use --conf-dir instead
Option '-confdir', '-confdir': Deprecated for removal since 9.7: Use --conf-dir instead
Option '-confdir', '-confdir': Deprecated for removal since 9.7: Use --conf-dir instead
Option '-confname', '-confname': Deprecated for removal since 9.7: Use --conf-name instead
Created new core 'employee_data'
C:\Users\A\Documents\solr-9.7.0\bin>post -c employee_data C:\Users\A\Documents\data\employee_data.csv
'post' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\A\Documents\solr-9.7.0\bin>java -jar post.jar -c employee_data C:\Users\A\Documents\data\employee_data.csv
Error: Unable to access jarfile post.jar
C:\Users\A\Documents\solr-9.7.0\bin>solr post -c employee_data C:\Users\A\Documents\data\employee_data.csv
Posting files to [base] url http://localhost:8983/solr/employee_data/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
Posttool: WARNING: No files or directories matching C:\Users\A\Documents\data\employee_data.csv
0 files indexed.
COMMITTING Solr index changes to http://localhost:8983/solr/employee_data/update...
WARNING: URIs provided to this tool needn't include Solr's context-root (e.g. "/solr/"). Such URIs are deprecated and support for them will be removed in a future release. Correcting from [http://localhost:8983/solr] to [http://localhost:8983/].
Time spent: 0:00:00.472
C:\Users\A\Documents\solr-9.7.0\bin>solr post -c employee_data C:\Users\A\Documents\data\employee_data.csv
Posting files to [base] url http://localhost:8983/solr/employee_data/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
```

Step 5: Index Employee Data from Kaggle

1. Download the employee dataset from [Kaggle](#). Convert it into a JSON or CSV format if necessary.
2. Upload/Index the data to Solr:

Solr provides a tool called post to upload data easily. Assuming your data is in a CSV file named `employee_data.csv`, run the following command to index it:

`./post -c employee_data /path/to/employee_data.csv`

```
Command Prompt

C:\Users\A\cd Documents
C:\Users\A\Documents>cd solr-9.7.0
C:\Users\A\Documents\solr-9.7.0>cd hl
The system cannot find the path specified.
C:\Users\A\Documents\solr-9.7.0>cd bin
C:\Users\A\Documents\solr-9.7.0\bin>soler start
ERROR: Process 28576 is already listening on port 8983. If this is Solr, please stop it first before starting (or use restart). If this is not Solr, then please choose a different port using -p PORT

C:\Users\A\Documents\solr-9.7.0\bin>soler create -c employee_data
WARNING: Using default configset with data driven schema functionality. NOT RECOMMENDED for production use.
To turn off: bin\soler config -c employee_data -p 8983 -action set-user-property -property update.autoCreateFields -value false
Option '-solrhl', '-solrhl': Depreciated for removal since 9.7; Use --solr-url instead
WARNING: URIs provided to this tool needn't include Solr's context-root (e.g. "/solr/"). Such URIs are deprecated and support for them will be removed in a future release. Correcting from [http://localhost:8983/solr] to [http://localhost:8983/].
Option '-confdir', '-confdir': Depreciated for removal since 9.7; Use --conf-dir instead
Option '-confdir', '-confdir': Depreciated for removal since 9.7; Use --conf-dir instead
Option '-confdir', '-confdir': Depreciated for removal since 9.7; Use --conf-dir instead
Option '-confname', '-confname': Depreciated for removal since 9.7; Use --conf-name instead
Created new core 'employee_data'

C:\Users\A\Documents\solr-9.7.0\bin>post -c employee_data C:\Users\A\Documents\data\employee_data.csv
'post' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\A\Documents\solr-9.7.0\bin>java -jar post.jar -c employee_data C:\Users\A\Documents\data\employee_data.csv
Error: Unable to access jarfile post.jar

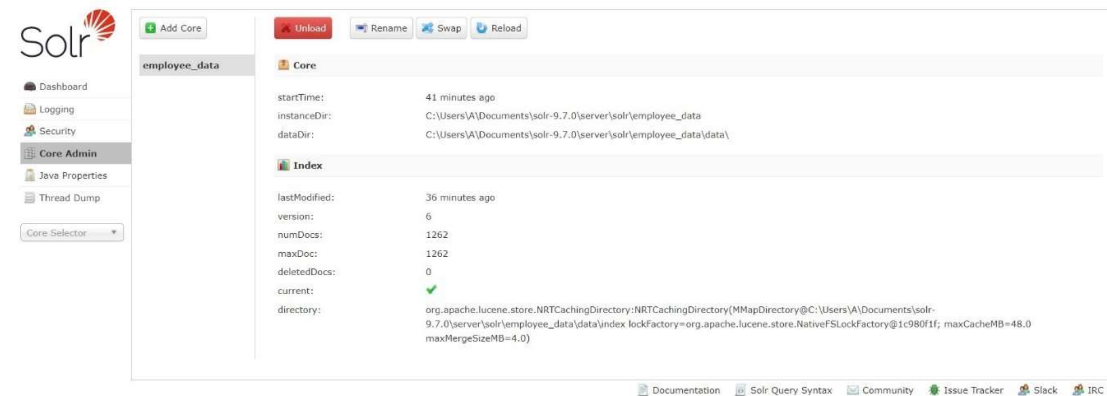
C:\Users\A\Documents\solr-9.7.0\bin>soler post -c employee_data C:\Users\A\Documents\data\employee_data.csv
Posting files to [base] url http://localhost:8983/solr/employee_data/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ots,rtf,htm,html,txt,log
PostTool: WARNING: No files or directories matching C:\Users\A\Documents\data\employee_data.csv
0 files indexed.
COMMITTING Solr index changes to http://localhost:8983/solr/employee_data/update...
WARNING: URIs provided to this tool needn't include Solr's context-root (e.g. "/solr/"). Such URIs are deprecated and support for them will be removed in a future release. Correcting from [http://localhost:8983/solr] to [http://localhost:8983/].
Time spent: 0:00:00.471


C:\Users\A\Documents\solr-9.7.0\bin>soler post -c employee_data C:\Users\A\Documents\data\employee_data.csv
Posting files to [base] url http://localhost:8983/solr/employee_data/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
```

Step 05: Verify the data is indexed:

- Go to the Solr Admin UI (<http://localhost:8983/solr/>).
- Select the employee_data collection from the left sidebar.
- Navigate to the Query section.
- Perform a query to fetch all indexed documents by entering `*:*` and clicking Execute Query.

You should see a list of documents returned from the indexed data.





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

Request-Handler (qt)

/select

common

q

:

q.op

OR

fq

sort

start, rows

0 10

fl

df

paramset(s)

Select paramset(s)

wt

indent on

debugQuery

defType

http://localhost:8983/solr/employee_data/select?indent=true&q.op=OR&q=%3A*&useParams=

```

{
  "responseHeader": {
    "status": 0,
    "QTime": 54,
    "params": {
      "q": "/*:*/*",
      "indent": "true",
      "q.op": "OR",
      "useParams": "true",
      "_:": "1727891845048"
    }
  },
  "response": {
    "numFound": 1262,
    "start": 0,
    "numFoundExact": true,
    "docs": [
      {
        "Department": ["Engineering"],
        "Gender": ["Male"],
        "Ethnicity": ["Asian"],
        "Age": [47],
        "Country": ["United States"],
        "City": ["Columbus"],
        "Id": "093e7cdd-e67a-41da-b80f-13f4d9300c3f",
        "Employee_ID": ["E000002"],
        "Full_Name": ["Kal Le"],
        "Job_Title": ["Controls Engineer"],
        "Business_Unit": ["Manufacturing"],
        "Hire_Date": ["2/5/2022"],
        "Annual_Salary": ["$92,368 "],
        "Bonus_": ["0%"],
        "_version": "1811825560222957568",
        "_root_": "893e7cdd-e67a-41da-b80f-13f4d9300c3f"
      },
      {
        "_version": "1811825560572133376",
        "_root_": "128e7c0b-8d24-4e0b-89ae-11fb2606386e"
      },
      {
        "Department": ["IT"],
        "Gender": ["Female"],
        "Ethnicity": ["Caucasian"],
        "Age": [38],
        "Country": ["United States"],
        "City": ["Seattle"],
        "Id": "578a262a-1c3b-4b0e-8e5e-59fe0ed09861",
        "Employee_ID": ["E02010"],
        "Full_Name": ["Gianna Holmes"],
        "Job_Title": ["System Administrator"],
        "Business_Unit": ["Manufacturing"],
        "Hire_Date": ["9/9/2011"],
        "Annual_Salary": ["$97,630 "],
        "Bonus_": ["0%"],
        "_version": "1811825560573181952",
        "_root_": "578a262a-1c3b-4b0e-8e5e-59fe0ed09861"
      },
      {
        "Department": ["Finance"],
        "Gender": ["Male"],
        "Ethnicity": ["Caucasian"],
        "Age": [52],
        "Country": ["United States"],
        "City": ["Miami"],
        "Id": "7ef508ce-4248-4459-8022-8626268db592",
        "Employee_ID": ["E02011"],
        "Full_Name": ["Jameson Thomas"],
        "Job_Title": ["Manager"],
        "Business_Unit": ["Specialty Products"],
        "Hire_Date": ["2/5/2015"],
        "Annual_Salary": ["$105,879 "],
        "Bonus_": ["10%"],
        "_version": "1811825560574230528",
        "_root_": "7ef508ce-4248-4459-8022-8626268db592"
      }
    ]
  }
}

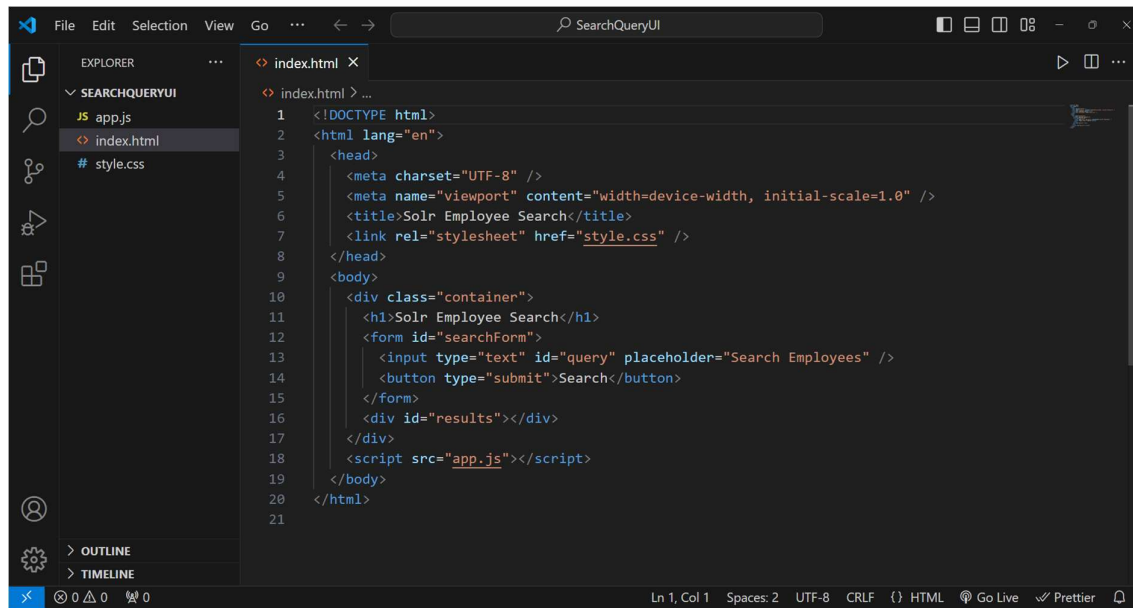
```

Additional Feature:

Apache Solr Search UI with HTML, CSS, and JavaScript

Building a user interface that interacts with an Apache Solr backend, allowing users to query a Solr collection and display results on the frontend.

Step 01: Create index.html



```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <title>Solr Employee Search</title>
7     <link rel="stylesheet" href="style.css" />
8   </head>
9   <body>
10    <div class="container">
11      <h1>Solr Employee Search</h1>
12      <form id="searchForm">
13        <input type="text" id="query" placeholder="Search Employees" />
14        <button type="submit">Search</button>
15      </form>
16      <div id="results"></div>
17    </div>
18    <script src="app.js"></script>
19  </body>
20 </html>
21
```

Step 02: Create style.css

```
* {
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}

body {
  font-family: Arial, sans-serif;
  background-color: #f4f4f4;
  padding: 20px;
}

.container {
  max-width: 600px;
  margin: 0 auto;
  background: #fff;
  padding: 20px;
  border-radius: 8px;
  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
}

h1 {
  text-align: center;
  margin-bottom: 20px;
}

form {
  display: flex;
  justify-content: center;
```

```

}

input {
  padding: 10px;
  width: 300px;
  margin-right: 10px;
  border-radius: 5px;
  border: 1px solid #ccc;
}

button {
  padding: 10px 20px;
  border: none;
  background-color: #007bff;
  color: white;
  border-radius: 5px;
  cursor: pointer;
}

button:hover {
  background-color: #0056b3;
}

#results {
  margin-top: 20px;
}

.result-item {
  background-color: #e9ecef;
  padding: 10px;
  margin-bottom: 10px;
  border-radius: 5px;
}

```

Step 03: Create app.js

```

document
  .getElementById("searchForm")
  .addEventListener("submit", async function (e) {
    e.preventDefault();

    // Get the query from the input field
    const query = document.getElementById("query").value;

    // Make sure there's a query entered
    if (query.trim() === "") {
      alert("Please enter a search term");
      return;
    }

    // Solr query URL
    const solrUrl = `http://localhost:8983/solr/employee_data/select?q=${query}&wt=json`;

    try {

```



```

// Fetch data from Solr
const response = await fetch(solrUrl);
const data = await response.json();

// Display results
displayResults(data.response.docs);
} catch (error) {
  console.error("Error fetching data:", error);
  alert("Failed to fetch data from Solr");
}
});

function displayResults(docs) {
  const resultsDiv = document.getElementById("results");
  resultsDiv.innerHTML = "";

  if (docs.length === 0) {
    resultsDiv.innerHTML = "<p>No results found</p>";
    return;
  }

  docs.forEach((doc) => {
    const resultItem = document.createElement("div");
    resultItem.classList.add("result-item");

    // Extract fields from the document, ensuring to access the first item in arrays
    resultItem.innerHTML = `
    <p><strong>Employee ID:</strong> ${doc.Employee_ID[0]}</p>
    <p><strong>Full Name:</strong> ${doc.Full_Name[0]}</p>
    <p><strong>Job Title:</strong> ${doc.Job_Title[0]}</p>
    <p><strong>Department:</strong> ${doc.Department[0]}</p>
    <p><strong>Gender:</strong> ${doc.Gender[0]}</p>
    <p><strong>Ethnicity:</strong> ${doc.Ethnicity[0]}</p>
    <p><strong>Age:</strong> ${doc.Age[0]}</p>
    <p><strong>Country:</strong> ${doc.Country[0]}</p>
    <p><strong>City:</strong> ${doc.City[0]}</p>
    <p><strong>Hire Date:</strong> ${doc.Hire_Date[0]}</p>
    <p><strong>Annual Salary:</strong> ${doc.Annual_Salary[0]}</p>
    <p><strong>Bonus:</strong> ${doc.Bonus__[0]}</p>
    `;

    resultsDiv.appendChild(resultItem);
  });
}

```

Result:

Telegram WebMeet - atwChatGPTChatGPTInbox (2/18/1)Employee SamSolr EmployeeSolr Admin

127.0.0.1:5500

Solr Employee Search

Search

Employee ID: E02002

Full Name: Kai Le

Job Title: Controls Engineer

Department: Engineering

Gender: Male

Ethnicity: Asian

Age: 47

Country: United States

City: Columbus

Hire Date: 2/5/2022

Annual Salary: \$92,368

Bonus: 0%

Employee ID: E02003

Full Name: Robert Patel

Job Title: Analyst

Department: Sales

Gender: Male

Ethnicity: Asian

Age: 58

Country: United States

City: Chicago

Hire Date: 10/23/2013

Annual Salary: \$45,703

Bonus: 0%

Employee ID: E02004

Full Name: Cameron Lo

Job Title: Network Administrator

Department: IT

Search

01:0103-10-2024

Solr Employee Search

Search

Employee ID: E02002

Full Name: Kai Le

Job Title: Controls Engineer

Department: Engineering

Gender: Male

Ethnicity: Asian

Age: 47

Country: United States

City: Columbus

Hire Date: 2/5/2022

Annual Salary: \$92,368

Bonus: 0%

Employee ID: E02003

Full Name: Robert Patel

Job Title: Analyst

Department: Sales

Gender: Male

Ethnicity: Asian

Age: 58

Country: United States

City: Chicago

Hire Date: 10/23/2013

Annual Salary: \$45,703

Bonus: 0%

Employee ID: E02004

Full Name: Cameron Lo

Job Title: Network Administrator

Department: IT

Gender: Male

Ethnicity: Asian

Age: 34

Country: China

City: Shanghai

Hire Date: 3/24/2019

Annual Salary: \$63,576

Bonus: 0%

Employee ID: E02005

Full Name: Harper Castillo

Job Title: IT Systems Architect

Department: IT

Gender: Female

Ethnicity: Latino

Age: 39

Country: United States

City: Seattle

Hire Date: 4/7/2018

Annual Salary: \$98,062

Bonus: 0%

Employee ID: E02006

Full Name: Harper Dominguez

Job Title: Director

Department: Engineering

Gender: Female

Ethnicity: Latino

Age: 42

Country: United States

City: Austin

Hire Date: 6/18/2005

Annual Salary: \$175,391

Bonus: 24%

Function Executions:

1. Create a Collection: createCollection(v_nameCollection)

```
createCollection('Hash_PoulomiBhattacharya')
```

Output:

```
Collection 'Hash_PoulomiBhattacharya' created successfully.
```

2. Create a Collection: createCollection(v_phoneCollection)

```
createCollection('Hash_1234')
```

Output:

```
Collection 'Hash_1234' created successfully.
```

3. Get Employee Count: getEmpCount(v_nameCollection)

```
getEmpCount('Hash_PoulomiBhattacharya')
```

Output:

```
Total number of employees in 'Hash_PoulomiBhattacharya': 1262
```

4. Index Employee Data Excluding Department: indexData(v_nameCollection, 'Department')

```
indexData('Hash_PoulomiBhattacharya', 'Department')
```

Output:

```
Employee data indexed into 'Hash_PoulomiBhattacharya', excluding column 'Department'
```

5. Index Employee Data Excluding Gender: indexData(v_phoneCollection, 'Gender')

```
indexData('Hash_1234', 'Gender')
```

Output:

```
Employee data indexed into 'Hash_1234', excluding column 'Gender'.
```

6. Delete Employee by ID: delEmpById(v_nameCollection, 'E02003')

```
delEmpById('Hash_PoulomiBhattacharya', 'E02003')
```

Output:

```
Employee with ID 'E02003' deleted from collection 'Hash_PoulomiBhattacharya'.
```

7. Get Employee Count Again: getEmpCount(v_nameCollection)

```
getEmpCount('Hash_PoulomiBhattacharya')
```

Output:

```
Total number of employees in 'Hash_PoulomiBhattacharya': 1261
```

8. Search by Department (IT): searchByColumn(v_nameCollection, 'Department', 'IT')

```
searchByColumn('Hash_PoulomiBhattacharya', 'Department', 'IT')
```

Output:

```
{
  "Department": "IT",
  "Gender": "Female",
  "Ethnicity": "Caucasian",
  "Age": 18,
  "Country": "United States",
  "City": "Seattle",
  "Employee_ID": "E02001",
  "Full_Name": "Gianna Holmes",
  "Job_Title": "Junior Administrator",
  "Business_Unit": "Manufacturing",
  "Hire_Date": "7/9/2017",
  "Annual_Salary": "$77,630.10",
  "Bonus": "$7,823.34"
}
```

9. Search by Gender (Male): searchByColumn(v_nameCollection, 'Gender', 'Male')

```
searchByColumn('Hash_PoulomiBhattacharya', 'Gender', 'Male')
```

Output:

```
No results found for gender 'Male'.
```

10. Search by Department in v_phoneCollection: searchByColumn(v_phoneCollection, 'Department', 'IT')

```
searchByColumn('Hash_1234', 'Department', 'IT')
```

Output:

```
{
  "Department": "IT",
  "Gender": "Female",
  "Ethnicity": "Caucasian",
  "Age": 18,
  "Country": "United States",
  "City": "Seattle",
  "Employee_ID": "E02001",
  "Full_Name": "Gianna Holmes",
  "Job_Title": "Junior Administrator",
  "Business_Unit": "Manufacturing",
  "Hire_Date": "7/9/2017",
  "Annual_Salary": "$77,630.10",
  "Bonus": "$7,823.34"
}
```

11. Get Department Faceting for v_nameCollection: getDepFacet(v_nameCollection)

```
getDepFacet('Hash_PoulomiBhattacharya')
```

Output:

```
{
  "IT": 1,
  "Finance": 1
}
```

12. Get Department Faceting for v_phoneCollection: getDepFacet(v_phoneCollection)

```
getDepFacet('Hash_1234')
```

Output:

```
{  
  "IT": 1,  
  "Finance": 1  
}
```

Search UI

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Solr Employee Search</title>  
</head>  
<body>  
  <h1>Search Employees by Column</h1>  
  <form id="search-form">  
    <label for="column">Column:</label>  
    <input type="text" id="column" name="column" required>  
    <label for="value">Value:</label>  
    <input type="text" id="value" name="value" required>  
    <button type="submit">Search</button>  
  </form>  
  
  <div id="results"></div>  
  
  <script>  
    document.getElementById('search-form').addEventListener('submit', function(e) {  
      e.preventDefault();  
      const column = document.getElementById('column').value;  
      const value = document.getElementById('value').value;  
  
      fetch(`http://localhost:8983/solr/Hash_PoulomiBhattacharya/select?q=${column}:${value}&  
wt=json`)  
        .then(response => response.json())  
        .then(data => {  
          const results = document.getElementById('results');  
          results.innerHTML = JSON.stringify(data.response.docs);  
        });  
    });  
  </script>  
</body>
```

</html>