

Assessment 3

Java Code:

```
import org.apache.solr.client.solrj.SolrClient;

import org.apache.solr.client.solrj.SolrQuery;

import org.apache.solr.client.solrj.impl.HttpSolrClient;

import org.apache.solr.client.solrj.response.QueryResponse;

import org.apache.solr.client.solrj.response.UpdateResponse;

import org.apache.solr.common.SolrInputDocument;

import org.apache.solr.client.solrj.response.FacetField;

import java.util.List;

public class SolrEmployeeManager

{

    private SolrClient solrClient;

    public SolrEmployeeManager(String collectionUrl) {

        solrClient = new HttpSolrClient.Builder(collectionUrl).build();

    }

    public void indexData(String excludeColumn) throws Exception {
```

```
String[][] employeeData = {  
  
    {"E02001", "John", "IT", "Male"},  
  
        {"E02002", "Jane", "HR", "Female"},  
  
        {"E02003", "Doe", "Finance", "Male"},  
  
        {"E02004", "Alice", "IT", "Female"}  
  
};  
  
for (String[] employee : employeeData) {  
  
    SolrInputDocument doc = new SolrInputDocument();  
  
    doc.addField("EmployeeID", employee[0]);  
  
    doc.addField("FullName", employee[1]);  
  
    if (!excludeColumn.equals("Department")) {  
  
        doc.addField("Department", employee[2]);  
  
    }  
  
    if (!excludeColumn.equals("Gender")) {  
  
        doc.addField("Gender", employee[3]);  
  
    }  
  
    solrClient.add(doc);  
  
}  
  
solrClient.commit();
```

```
System.out.println("Indexed data into Solr, excluding column: " +  
excludeColumn);
```

```
}
```

```
public List searchByColumn(String columnName, String columnValue)  
throws Exception {
```

```
SolrQuery query = new SolrQuery();
```

```
query.setQuery(columnName + ":" + columnValue);
```

```
QueryResponse response = solrClient.query(query);
```

```
return response.getResults();
```

```
}
```

```
public long getEmpCount() throws Exception {
```

```
SolrQuery query = new SolrQuery("*:*"); QueryResponse response =  
solrClient.query(query);
```

```
return response.getResults().getNumFound();
```

```
}
```

```
public void delEmpById(String employeeId) throws Exception {
```

```
UpdateResponse response = solrClient.deleteById(employeeId);
```

```
solrClient.commit();
```

```
System.out.println("Deleted employee with ID: " + employeeId);
```

```
}
```

```
public void getDepFacet() throws Exception {

    SolrQuery query = new SolrQuery("*:*");

    query.setFacet(true);

    query.addFacetField("Department");

    QueryResponse response = solrClient.query(query);

    List<FacetField> facets = response.getFacetFields();

    for (FacetField facet : facets) {

        for (FacetField.Count count : facet.getValues()) {

            System.out.println("Department: " + count.getName() + ", Count: " +
            count.getCount());

        }

    }

}

public void close() throws Exception {

    solrClient.close();

}

public static void main(String[] args) {

    try {

        String nameCollectionUrl = "http://localhost:8983/solr/Hash_Tanvi";

        String phoneCollectionUrl = "http://localhost:8983/solr/Hash_1234";
```

```
SolrEmployeeManager managerName = new
SolrEmployeeManager(nameCollectionUrl);

managerName.indexData("Department");

long countBeforeDeletion = managerName.getEmpCount();

System.out.println("Employee count (before deletion): " +
countBeforeDeletion);

managerName.delEmpById("E02003");

long countAfterDeletion = managerName.getEmpCount();

System.out.println("Employee count (after deletion): " + countAfterDeletion);

List results = managerName.searchByColumn("Department", "IT");

System.out.println("Search results for Department=IT: " + results.size());

managerName.getDepFacet(); managerName.close(); SolrEmployeeManager
managerPhone = new SolrEmployeeManager(phoneCollectionUrl);

managerPhone.indexData("Gender"); managerPhone.getDepFacet();

managerPhone.close();

}

catch (Exception e) {

e.printStackTrace();

}

}

}
```

OUTPUT

Indexed data into Solr, excluding column: Department

Employee count (before deletion): 4

Deleted employee with ID: E02003

Employee count (after deletion): 3

Search results for Department=IT: 1

Department: IT, Count: 2

Department: HR, Count: 1