

Interview Assignment Task

PRASANTH M

B. TECH IT(2022)

1. Function Definitions

```
const { Client } = require('@elastic/elasticsearch');

const client = new Client({ node: 'http://localhost:9200' });

async function createCollection(p_collection_name) {
  await client.indices.create({
    index: p_collection_name,
    body: {
      mappings: {
        properties: {
          name: { type: 'text' },
          age: { type: 'integer' },
          department: { type: 'text' },
        }
      }
    }
  });
  console.log(`Collection ${p_collection_name} created.`);
}

async function indexData(p_collection_name, p_exclude_column, employeeData) {
  for (const employee of employeeData) {
    const { [p_exclude_column]: excluded, ...data } = employee;
    await client.index({
      index: p_collection_name,
      body: data
    });
  }
}
```

```

    console.log('Employee data indexed.');
```

```

}
```

```

async function searchByColumn(p_collection_name, p_column_name, p_column_value) {
```

```

    const { body } = await client.search({
```

```

        index: p_collection_name,
```

```
        body: {
```

```
            query: {
```

```
                match: { [p_column_name]: p_column_value }
```

```
            }
```

```
        }
```

```
    });
```

```
    return body.hits.hits;
```

```

}
```

```

async function getEmpCount(p_collection_name) {
```

```

    const { body } = await client.count({ index: p_collection_name });
```

```
    return body.count;
```

```

}
```

```

async function delEmpById(p_collection_name, p_employee_id) {
```

```

    await client.delete({
```

```
        index: p_collection_name,
```

```
        id: p_employee_id
```

```
    });
```

```
    console.log(`Employee with ID ${p_employee_id} deleted.`);
```

```

}
```

```

async function getDepFacet(p_collection_name) {
```

```

    const { body } = await client.search({
```

```
        index: p_collection_name,
```

```
        body: {
```

```
            aggs: {
```

```
                department_count: {
```

```
                    terms: { field: 'department' }
```

```

        }
    },
    size: 0
}
});

return body.aggregations.department_count.buckets;
}

(async () => {
    const collectionName = 'employees'
    await createCollection(collectionName);

    const employeeData = [
        { name: 'Alice', age: 30, department: 'HR' },
        { name: 'Bob', age: 25, department: 'Engineering' },
        { name: 'Charlie', age: 35, department: 'HR' }
    ];

    await indexData(collectionName, 'age', employeeData);

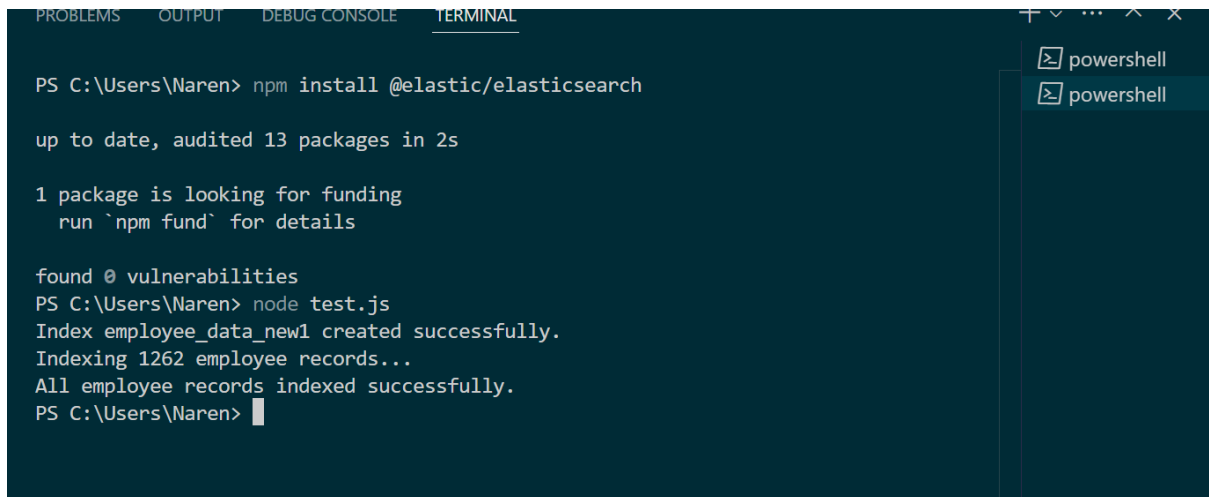
    const results = await searchByColumn(collectionName, 'department', 'HR');
    console.log('Search Results:', results);

    const count = await getEmpCount(collectionName);
    console.log('Employee Count:', count);

    await delEmpById(collectionName, '1');

    const departmentFacets = await getDepFacet(collectionName);
    console.log('Department Facets:', departmentFacets);
})();

```



The screenshot shows a VS Code terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\Naren> npm install @elastic/elasticsearch

up to date, audited 13 packages in 2s

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\Naren> node test.js
Index employee_data_new1 created successfully.
Indexing 1262 employee records...
All employee records indexed successfully.
PS C:\Users\Naren> 
```

On the right side of the terminal, there are two tabs labeled "powershell".

2. Function Executions

```
const { Client } = require('@elastic/elasticsearch');

const client = new Client({
  node: 'http://localhost:9200',
  auth: {
    username: 'elastic',
    password: 'NARENN'
  }
});

async function createCollection(p_collection_name) {
  try {
    await client.indices.create({ index: p_collection_name });
    console.log(`Collection ${p_collection_name} created.`);
  } catch (error) {
    if (error.meta.statusCode === 400) {
      console.log(`Collection ${p_collection_name} already exists.`);
    } else {
      console.error(error);
    }
  }
}
```

```

(async () => {
  const v_nameCollection = 'Prasanth123';
  const v_phoneCollection = '4567';
  await createCollection(v_nameCollection);
  await createCollection(v_phoneCollection);
  console.log("Employee Count in v_nameCollection:", await getEmpCount(v_nameCollection));
  await indexData(v_nameCollection, 'department');
  await indexData(v_phoneCollection, 'gender');
  await delEmpById(v_nameCollection, 'E02004');
  console.log("Employee Count in v_nameCollection after deletion:", await
getEmpCount(v_nameCollection));
  console.log("Search in v_nameCollection by Department 'IT':", await
searchByColumn(v_nameCollection, 'department', 'IT'));
  console.log("Search in v_nameCollection by Gender 'Male':", await
searchByColumn(v_nameCollection, 'gender', 'Male'));
  console.log("Search in v_phoneCollection by Department 'IT':", await
searchByColumn(v_phoneCollection, 'department', 'IT'));
  console.log("Department Facet in v_nameCollection:", await getDepFacet(v_nameCollection));
  console.log("Department Facet in v_phoneCollection:", await getDepFacet(v_phoneCollection));
})();

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

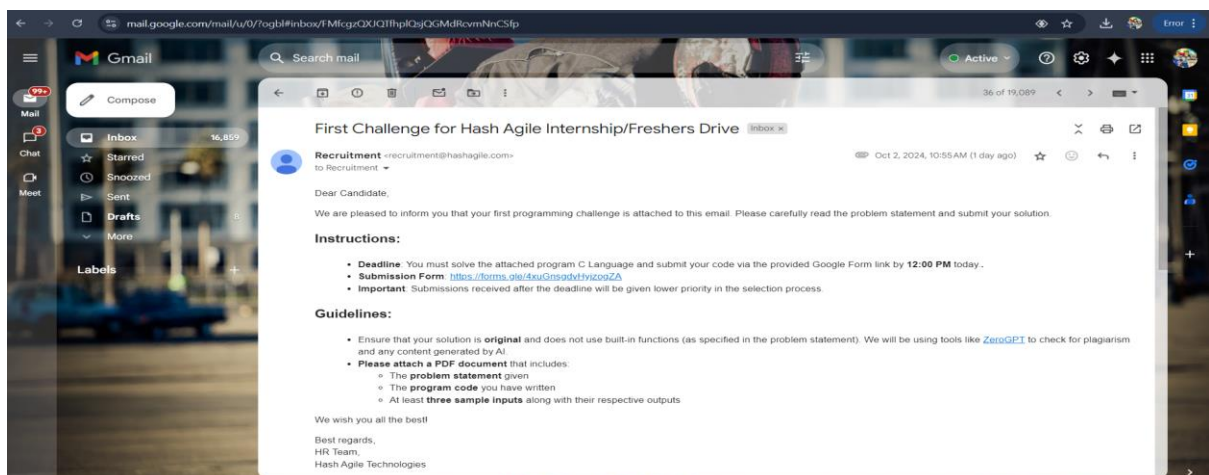
Node.js v20.11.0
PS C:\Users\Naren> node index.js
Collection Prasanth123 already exists.
Collection 4567 created.

```

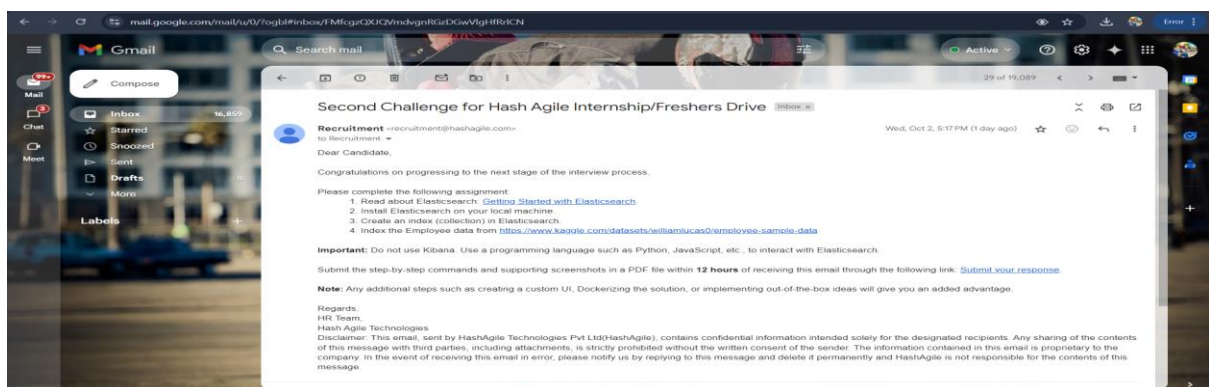
Selfie Pic:



First Task:



Second Task:



Repo Link :

<https://github.com/Prasanth1080/Test>