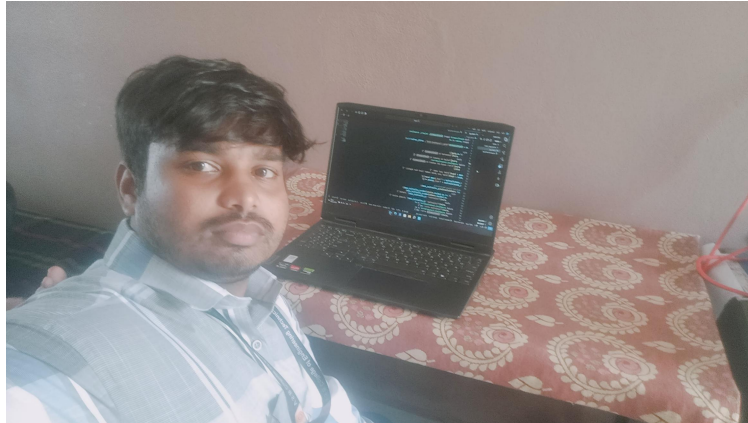


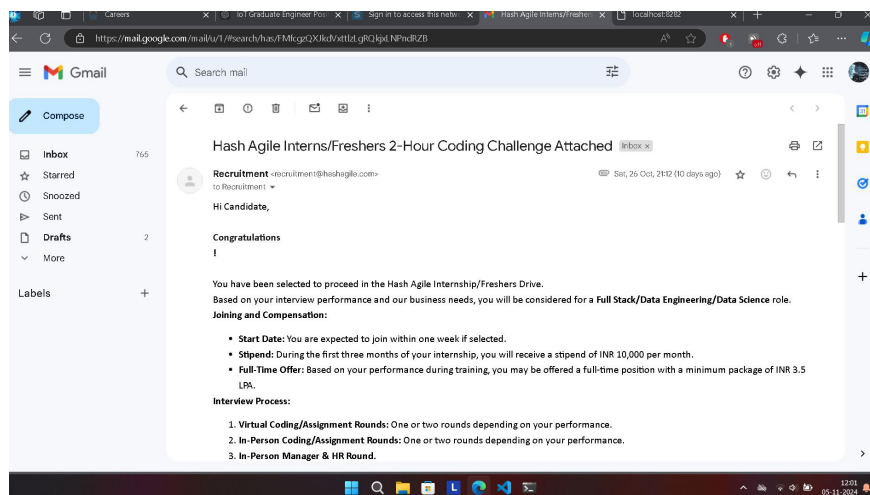
ROUND 2

Name: Sakthibala A

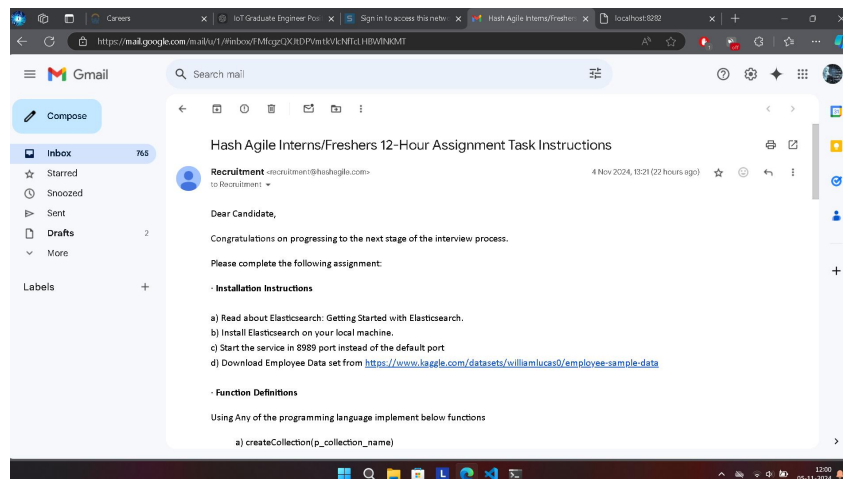
1. Selfie Pic:



2. 1 st Task Email Screenshot:



3. 2 nd Task Email:



4. Github URL For Round 1: [darkbala343/HASH_ROUND- 1](https://github.com/darkbala343/HASH_ROUND-1)
5. Github URL For Assesment: [darkbala343/HASH_ROUND-2](https://github.com/darkbala343/HASH_ROUND-2)
6. Function Execution Results:

```
1 from elasticsearch import Elasticsearch, helpers, exceptions
2 import pandas as pd
3
4 es = Elasticsearch("http://localhost:8282", verify_certs=False)
5
6 try:
7     if es.ping():
8         print("Connected to Elasticsearch.")
9     else:
10        print("Failed to connect to Elasticsearch.")
11 except exceptions.ConnectionError:
12    pass
13
14 Enter your name: Bala
15 Enter your phone number (last four digits): 1930
16 Collection 'hash_bala' already exists.
17 Collection 'hash_1930' created.
18 Employee count in 'hash_bala': 53
19 An error occurred: 407 document(s) failed to index.
20 An error occurred: 407 document(s) failed to index.
21 Employee count in 'hash_bala': 53
22 D:\hash\main.py:80: DeprecationWarning: Passing transport options in the API method is deprecated. Use 'Elasticsearch.options()' instead.
23     res = es.delete(index=p.collection_name, id=p.employee_id, ignore=[404])
24 Deleted employee ID E00803 from 'hash_bala'.
25 Employee count in 'hash_bala': 53
26 Search results for Department-IT in 'hash_bala':
27 []
```

```
ge': 27.0, 'Hire Date': '3/14/2022', 'Annual Salary': '$120,315', 'Bonus %': '0%', 'Country': 'United States', 'City': 'Miami',
'Exit Date': '7/17/2022'}, {'_index': 'hash_bala', '_id': 'E02113', '_score': 0.6747981, '_source': {'Employee ID': 'E02113',
'Full Name': 'Nolan Liang', 'Job Title': 'Sr. Analyst', 'Business Unit': 'Manufacturing', 'Gender': 'Male', 'Ethnicity': 'Asian',
'Age': 28.0, 'Hire Date': '03-04-2021', 'Annual Salary': '$85,206', 'Bonus %': '0%', 'Country': 'China', 'City': 'Chongqing',
'Exit Date': '07-08-2022'}}, {'_index': 'hash_bala', '_id': 'E02167', '_score': 0.6747981, '_source': {'Employee ID': 'E02167',
'Full Name': 'Nolan Howard', 'Job Title': 'Quality Engineer', 'Business Unit': 'Research and Development', 'Gender': 'Male',
'Ethnicity': 'Caucasian', 'Age': 35.0, 'Hire Date': '07-11-2016', 'Annual Salary': '$124,287', 'Bonus %': '0%', 'Country': 'United States', 'City': 'Austin', 'Exit Date': '6/23/2022'}}, {'_index': 'hash_bala', '_id': 'E02175', '_score': 0.6747981, '_source': {'Employee ID': 'E02175',
'Full Name': 'Christopher Robinson', 'Job Title': 'Vice President', 'Business Unit': 'Specialty Products', 'Gender': 'Male', 'Ethnicity': 'Caucasian', 'Age': 35.0, 'Hire Date': '07-11-2016', 'Annual Salary': '$124,287', 'Bonus %': '0%', 'Country': 'United States', 'City': 'Austin', 'Exit Date': '10/17/2022'}}, {'_index': 'hash_bala', '_id': 'E02186', '_score': 0.6747981, '_source': {'Employee ID': 'E02186',
'Full Name': 'Logan Reyes', 'Job Title': 'Quality Engineer', 'Business Unit': 'Corporate', 'Gender': 'Male', 'Ethnicity': 'Latino', 'Age': 62.0, 'Hire Date': '04-10-1995', 'Annual Salary': '$78,251', 'Bonus %': '0%', 'Country': 'Brazil', 'City': 'Sao Paulo', 'Exit Date': '3/19/2020'}}, {'_index': 'hash_bala', '_id': 'E02202', '_score': 0.6747981, '_source': {'Employee ID': 'E02202',
'Full Name': 'Charles Ramos', 'Job Title': 'Sr. Analyst', 'Business Unit': 'Corporate', 'Gender': 'Male', 'Ethnicity': 'Latino', 'Age': 37.0, 'Hire Date': '04-04-2010', 'Annual Salary': '$74,597', 'Bonus %': '0%', 'Country': 'United States', 'City': 'Austin', 'Exit Date': '1/30/2016'}}, {'_index': 'hash_bala', '_id': 'E02220', '_score': 0.6747981, '_source': {'Employee ID': 'E02220',
'Full Name': 'Thomas Rodriguez', 'Job Title': 'Analyst', 'Business Unit': 'Manufacturing', 'Gender': 'Male', 'Ethnicity': 'Latino', 'Age': 36.0, 'Hire Date': '10-09-2019', 'Annual Salary': '$51,669', 'Bonus %': '0%', 'Country': 'Brazil', 'City': 'Rio de Janeiro', 'Exit Date': '9/29/2022'}}, {'_index': 'hash_bala', '_id': 'E02230', '_score': 0.6747981, '_source': {'Employee ID': 'E02230',
'Full Name': 'Nolan Brooks', 'Job Title': 'Business Partner', 'Business Unit': 'Manufacturing', 'Gender': 'Male', 'Ethnicity': 'Caucasian', 'Age': 40.0, 'Hire Date': '11-08-2014', 'Annual Salary': '$49,342', 'Bonus %': '0%', 'Country': 'United States', 'City': 'Seattle', 'Exit Date': '8/27/2019'}}
Search results for Department-IT in 'hash_1930':
[]
Employee count grouped by department:
[]
```

Service running on port 8989:
<http://localhost:8989>

```
{
  "name": "DESKTOP-RC691PS",
  "cluster_name": "elasticsearch",
  "cluster_uuid": "d2q8p9ixqykb-GmQxIa",
  "version": {
    "number": "8.15.3",
    "build_flavor": "default",
    "build_type": "zip",
    "build_hash": "197332668b55c3a05c73a74c28a1f6d6923010",
    "build_date": "2024-10-09T22:08:00.328917561Z",
    "build_snapshot": false,
    "lucene_version": "9.11.1",
    "minimum_wire_compatibility_version": "7.17.0",
    "minimum_index_compatibility_version": "7.0.0"
  },
  "tagline": "You know, for Search"
}
```

Program:

```
name = input("Enter your name: ")
phone = input("Enter your phone number (last four digits): ")

v_nameCollection = f'hash_{name.lower()}'
v_phoneCollection = f'hash_{phone[-4:]}'

Qodo Gen: Options | Test this function
def createCollection(p_collection_name):
    try:
        if not es.indices.exists(index=p_collection_name):
            es.indices.create(index=p_collection_name)
            print(f"Collection '{p_collection_name}' created.")
        else:
            print(f"Collection '{p_collection_name}' already exists.")
    except exceptions.ConnectionError as e:
        print(f"Connection error: {e}")
    except exceptions.TransportError as e:
        print(f"Elasticsearch error: {e}")
```

```
def getDepFacet(p_collection_name):
    query = {
        "field": "Department.keyword"
    }
    res = es.search(index=p_collection_name, body=query)
    print("Employee count grouped by department:")
    return res['aggregations']['departments']['buckets']
except exceptions.ConnectionError as e:
    print(f"Connection error: {e}")

createCollection(v_nameCollection)
createCollection(v_phoneCollection)
getEmpCount(v_nameCollection)
indexData(v_nameCollection, 'Department')
indexData(v_phoneCollection, 'Gender')
getEmpCount(v_nameCollection)
delEmpById(v_nameCollection, 'E02003')
getEmpCount(v_nameCollection)
print(searchByColumn(v_nameCollection, 'Department', 'IT'))
print(searchByColumn(v_nameCollection, 'Gender', 'Male'))
print(searchByColumn(v_phoneCollection, 'Department', 'IT'))
print(getDepFacet(v_nameCollection))
print(getDepFacet(v_phoneCollection))
```