HASHAGILE ROUND CODING:

```
import pandas as pd
from pymongo import MongoClient

# Load the CSV data into a pandas DataFrame
df = pd.read_csv("Employee Sample Data 1.csv")

# Initialize MongoDB Client
client = MongoClient("mongodb://localhost:27017/")  # Connect to local MongoDB
db = client["employee_database"]  # Create or connect to a database

# a) Create Collection
def createCollection(p_collection_name):
    collection = db[p_collection_name]  # Create or access the specified collection
    print(f"Collection '{p_collection_name}' created or accessed.")
```

```
    return collection


# b) Index Data
def indexData(p_collection_name,
p_exclude_column=None):
    collection =
createCollection(p_collection_name)
    # Remove the specified column from the
DataFrame if provided
    if p_exclude_column:
        data_to_index =
df.drop(columns=[p_exclude_column])
    else:
        data_to_index = df
    # Convert DataFrame to dictionary records and
insert into MongoDB
    records = data_to_index.to_dict("records")
    collection.insert_many(records)
    print(f"Indexed {len(records)} documents into
'{p_collection_name}' collection.")
```

```
# c) Search by Column
def searchByColumn(p_collection_name, p_column_name, p_column_value):
    collection = db[p_collection_name]
    query = {p_column_name: p_column_value}
    results = list(collection.find(query))
    print(f"Found {len(results)} records where {p_column_name} = {p_column_value}")
    return results

# d) Get Employee Count
def getEmpCount(p_collection_name):
    collection = db[p_collection_name]
    count = collection.count_documents({})
    print(f"Total employee count in '{p_collection_name}' collection: {count}")
    return count
```

```
# e) Delete Employee by ID
def delEmpById(p_collection_name, p_employee_id):
    collection = db[p_collection_name]
    result = collection.delete_one({"Employee ID": p_employee_id})
    if result.deleted_count > 0:
        print(f"Deleted employee with ID {p_employee_id} from '{p_collection_name}' collection.")
    else:
        print(f"No employee found with ID {p_employee_id}.")

# f) Get Department Facet
def getDepFacet(p_collection_name):
    collection = db[p_collection_name]
    pipeline = [
        {"$group": {"_id": "$Department", "count": {"$sum": 1}}}
```

```
    ]
    department_counts = list(collection.aggregate(pipeline))
    print("Employee count grouped by department:")
    for dept in department_counts:
        print(f"{dept['_id']}: {dept['count']}")
    return department_counts

# Example Usage
collection_name = "employees"  # Define collection name

# Run each function in sequence
createCollection(collection_name)  # a) Create Collection
indexData(collection_name, p_exclude_column="Exit Date")  # b) Index Data (excluding "Exit Date" column)
```

print(searchByColumn(collection_name, "Department", "IT")) # c) Search by Column

print(getEmpCount(collection_name)) # d) Get Employee Count

delEmpById(collection_name, "E02001") # e) Delete Employee by ID

print(getDepFacet(collection_name)) # f) Get Department Facet

---

SAMPLE INPUT:

a) Var v_nameCollection = ‘Hash_<Your Name>’

b) Var v_phoneCollection =’Hash_<Your Phone last four digits>’

c) createCollection(v_nameCollection)

d) createCollection(v_phoneCollection)

e) getEmpCount(v_nameCollection)

f) indexData(v_nameCollection,’Department’)

g) indexData(v_ phoneCollection, ‘Gender’)

h) getEmpCount(v_nameCollection)

i) delEmpById (v_ nameCollection ,‘E02003’)

j) getEmpCount(v_nameCollection)

k) searchByColumn(v_nameCollection,’Department’,’IT’)

l) searchByColumn(v_nameCollection,’Gender’ ,’Male’)

m) searchByColumn(v_ phoneCollection,’Department’,’IT’)

n) getDepFacet(v_ nameCollection)

o) getDepFacet(v_ phoneCollection)

SAMPLE OUTPUT:

Define collection names based on your name and last four phone digits

v_nameCollection = "Hash_kaviya"

v_phoneCollection = "Hash_1234"

a) Create Collection for name-based and phone-based collections

createCollection(v_nameCollection) # Output: Collection 'Hash_Sharmila' created or accessed.

createCollection(v_phoneCollection) # Output: Collection 'Hash_1234' created or accessed.

b) Get initial employee count in v_nameCollection

print(getEmpCount(v_nameCollection)) # Expected Output: Total employee count in 'Hash_Sharmila' collection: <count>

c) Index data in v_nameCollection excluding 'Department' column

indexData(v_nameCollection, "Department") # Expected Output: Indexed <count> documents into 'Hash_Sharmila' collection.

d) Index data in v_phoneCollection excluding 'Gender' column

indexData(v_phoneCollection, "Gender") # Expected Output: Indexed <count> documents into 'Hash_1234' collection.

e) Get updated employee count in v_nameCollection

print(getEmpCount(v_nameCollection)) # Expected Output: Total employee count in 'Hash_Sharmila' collection: <count>

f) Delete an employee by ID in v_nameCollection

delEmpById(v_nameCollection, "E02003") # Expected Output: Deleted employee with ID E02003 from 'Hash_Sharmila' collection.

g) Get final employee count in v_nameCollection after deletion

print(getEmpCount(v_nameCollection)) # Expected Output: Total employee count in 'Hash_Sharmila' collection: <count after deletion>

h) Search by 'Department' in v_nameCollection for 'IT'

results_dept_it = searchByColumn(v_nameCollection, "Department", "IT")

print(results_dept_it) # Expected Output: Found <count> records where Department = IT.

i) Search by 'Gender' in v_nameCollection for 'Male'

results_gender_male = searchByColumn(v_nameCollection, "Gender", "Male")

print(results_gender_male) # Expected Output: Found <count> records where Gender = Male.

j) Search by 'Department' in v_phoneCollection for 'IT'

results_dept_it_phone = searchByColumn(v_phoneCollection, "Department", "IT")

print(results_dept_it_phone) # Expected Output: Found <count> records where Department = IT.

k) Get department facet in v_nameCollection

dept_facet_name = getDepFacet(v_nameCollection)

print(dept_facet_name) # Expected Output: Department grouping and counts in 'Hash_Sharmila' collection.

l) Get department facet in v_phoneCollection

dept_facet_phone = getDepFacet(v_phoneCollection)

print(dept_facet_phone) # Expected Output: Department grouping and counts in 'Hash_1234' collection.