Database Management System
(UE10CS301)
Assignment - 4 Report

Team: 10
Team Members:

Name:	SRN:
Aliyah Kabeer	PES2UG19CS031
Anwesha Kelkar	PES2UG19CS058
Arunav Dey	PES2uG19CS066

1. Language choice for the front end design: Python

1. Easy to Use and Read
2. Asynchronous Coding
3. Less-Limited Programming Approach
4. Enterprise Application Integration
5. Its Use In Scientific and Numeric Applications
6. Use in Prototyping
7. Portability and Interactivity

2. PostgreSQL database adapter used for python: psycopg2 Python module

The psycopg2 is a Python module which is used to work with the PostgreSQL database. Its main features are the complete implementation of the Python DB API 2.0 specification and the thread safety (several threads can share the same connection). It was designed for heavily multi-threaded applications that create and destroy lots of cursors and make a large number of concurrent “INSERT”s or “UPDATE”s.Many Python types are supported and adapted to match the PostgreSQL data types.

3. Screenshots for the statements executed from the front end

```
                Online Food Ordering System
 1. Select all employees
 2. Select all restaurants
 3. Select restaurant details whose rating is 5
 4. Select the employees who work in restaurants with the least rating
 5. Update KFC rating
 6. Select employees whose salary is more than average
 7. Create trigger to accept a valid salary
 8. Group restaurants and their average rating by their name
 9. Update all employees salary under the manager with managerid=2
 0. Exit
```

```
1
Designation:  Cook
Staff ID:  1
Salary:  20000
Name:  Hari
Manager ID:  2
Restaurant ID:  1
-----------------------
Designation:  Cook
Staff ID:  2
Salary:  10000
Name:  Harita
Manager ID:  1
Restaurant ID:  2
-----------------------
Designation:  Cook
Staff ID:  3
Salary:  50000
Name:  Haria
Manager ID:  2
Restaurant ID:  3
-----------------------
Designation:  Cook
Staff ID:  4
Salary:  70000
Name:  Arita
Manager ID:  1
Restaurant ID:  4
-----------------------
Designation:  Cook
Staff ID:  5
Salary:  90000
Name:  Hart
Manager ID:  2
Restaurant ID:  5
-----------------------
Designation:  Cook
Staff ID:  6
Salary:  15000
Name:  Rita
Manager ID:  1
Restaurant ID:  6
-----------------------
Read command successfully
```

```
2
Registration Number:  1
Address:  Indiranagar, Bengaluru
Rating:  4.0
----------------------
Registration Number:  2
Address:  Koramangala, Bengaluru
Rating:  5.0
----------------------
Registration Number:  3
Address:  Indiranagar, Bengaluru
Rating:  3.0
----------------------
Registration Number:  4
Address:  Koramangala, Bengaluru
Rating:  2.0
----------------------
Registration Number:  5
Address:  Hennur, Bengaluru
Rating:  4.0
----------------------
Registration Number:  6
Address:  Electronic City, Bengaluru
Rating:  1.0
----------------------
Read command successfully

3
Enter valid rating: -1
Invalid number. Rating must be between 0 and 5
Enter valid rating: 4
Contact No.:  9876543210
Website:  mcdonalds.com
----------------------
Contact No.:  9976543310
Website:  pizzahut.com
----------------------

4
Name:  Rita
----------------------
Read command successfully
```

```
5
Enter the Restaurant ID : 2
Enter a new valid rating (between 0 and 5) : 3
Updated successfully
```

```
6
Name:  Haria
Designation:  Cook
-----------------------
Name:  Arita
Designation:  Cook
-----------------------
Name:  Hart
Designation:  Cook
-----------------------
Read command successfully
```

```
7
Trigger created successfully
```

```
8
Name:  KFC
Average Rating:  4.0
-----------------------
Name:  Pizza Hut
Average Rating:  4.0
-----------------------
Name:  McDonalds
Average Rating:  4.0
-----------------------
Name:  Dominos
Average Rating:  1.0
-----------------------
Name:  Leon Grill
Average Rating:  3.0
-----------------------
Name:  Burger King
Average Rating:  2.0
-----------------------
Read command successfully
```

```
9
Enter the Manager ID :
2
Enter the salary change (%) :
20
Updated successfully
```

4. Screenshots for the schema change statements.

```
    #Altering the Schema
    alter_query = """ALTER SCHEMA foodordering RENAME TO Food_Ordering;"""
    cursor.execute(alter_query)
    print("Successfully altered schema name\n")
    alter_query = """ALTER SCHEMA Food_Ordering OWNER TO Aliyah;"""
    cursor.execute(alter_query)
    print("Successfully altered schema owner\n")

    #Altering constraints
    alter_query = """ALTER TABLE Drinks
RENAME COLUMN name TO beverage;"""
    cursor.execute(alter_query)
    print("Successfully altered column name\n")

    alter_query = """ALTER TABLE Desserts
ADD COLUMN Eggless varchar;"""
    cursor.execute(alter_query)
    print("Successfully added new column to schema \n")

    alter_query = """ALTER TABLE Desserts
ALTER COLUMN Eggless
SET DEFAULT 'Not specified';"""
    cursor.execute(alter_query)
    print("Successfully set default value to existing column\n")
```

```
restaurants=# ALTER SCHEMA foodordering RENAME TO Food_Ordering;
NOTICE:  snitch: ddl_command_start ALTER SCHEMA
ALTER SCHEMA
restaurants=# ALTER SCHEMA Food_Ordering OWNER TO Aliyah;
NOTICE:  snitch: ddl_command_start ALTER SCHEMA
ALTER SCHEMA
restaurants=# ALTER TABLE Drinks
restaurants-# RENAME COLUMN name TO beverage;
NOTICE:  snitch: ddl_command_start ALTER TABLE
ALTER TABLE
restaurants=# ALTER TABLE Desserts
restaurants-# ADD COLUMN Eggless varchar;
NOTICE:  snitch: ddl_command_start ALTER TABLE
ALTER TABLE
restaurants=# ALTER TABLE Desserts
restaurants-# ALTER COLUMN Eggless
restaurants-# SET DEFAULT 'Not specified';
NOTICE:  snitch: ddl_command_start ALTER TABLE
ALTER TABLE
restaurants=# \d Desserts
                                  Table "public.desserts"
   Column   |         Type          | Collation | Nullable |              Default
------------+-----------------------+-----------+----------+-----------------------------------
 dessertsid | integer               |           | not null |
 name       | character varying(32) |           |          |
 menuid     | integer               |           |          |
 eggless    | character varying     |           |          | 'Not specified'::character varying
```

5. Write up about the changes in Business/Applications changes/expansion - that might lead to

1. **schema changes:** A schema change is an alteration made to a collection of logical structures (or schema objects) in a database.When the data requirements of an organization change, the databases used to store the data must also change. If the data is not reliable and available, the system does not serve the business—rather, it threatens the health of the business. So, we need infallible techniques to manage database changes—and we need techniques that are not just fail-safe but also automated, efficient, and easy to use.

2. **constraint changes:**

- Prevent bad data from being entered into tables of interest.
- Enforce business logic at the database-level.
- Documentation of important database rules.
- Enforce relational integrity between any number of tables.
- Improve database performance.
- Enforce uniqueness.

3. **DBMS migration (from SQL based to No-SQL):** If there are existing RDBMS databases that are storing content, documents, files, or have unstructured data, then there are significant advantages in moving such databases to NoSQL databases. Benefits include cost benefits, performance, scalability, future proof for changes, reducing conversion jobs, and extensive supportability for analytics.

6. With the existing design of your database, if you have to migrate to any No-SQL variety, then which one will be your choice? Why?
No-SQL Database choice: MongoDB

- Flexible document schemas
- Code-native data access
- Change-friendly design

- Powerful querying and analytics
- Easy horizontal scale-out

7. Indicate contribution of each member clearly and approximate time spent in hours for the respective activity.

Team Member:	Contribution:
Aliyah Kabeer	Frontend, triggers & cursors
Anwesha Kelkar	Simple queries & reports
Arunav Dey	Complex queries & change in constraints and schema

- Each member contributed equally towards every aspect of this project.

- Our team took approximately 2 and a half hours to complete this assignment.