| Name: Pelagio, Ronian T. | Date: April 14, 2025 |
|---|---|
| Subject: MIT 502 - ADVANCED DATABASE MANAGEMENT SYSTEM | |

## ACTIVITY # 2 & 3: ADVANCED SQL & CRUD Function

### Activity # 2: ADVANCED SQL

**INSTRUCTION:** Answer the following questions based on the given tables, and paste the query include the result *(Screenshots)* after each question.

Table Name: EmployeeInfo

| Eid | Name | Position | Salary | Age | Address | DeptCode |
|---|---|---|---|---|---|---|
| 1 | Juan Santos | Manager | 20000 | 35 | San Pablo | BPD |
| 2 | Miguel Lopez | Secretary | 14000 | 30 | San Pablo | CRD |
| 3 | Jude King | Sales | 12000 | 34 | Calauan | SD |
| 4 | Pedro Lao | Manager | 20000 | 28 | Rizal | SD |
| 5 | Jamar Perez | Sales | 12000 | 30 | Rizal | CRD |

Table Name: Loan

| Eid | LoanAmount | Date |
|---|---|---|
| 1 | 5000 | 10/10/2022 |
| 2 | 2000 | 09/10/2022 |
| 1 | 4000 | 12/31/2021 |
| 5 | 1500 | 11/18/2021 |
| 5 | 7000 | 10/10/2021 |

Table Name: DepartmentInfo

| DeptCode | DeptDescription |
|---|---|
| BPD | BODY AND PAINT DEPARTMENT |
| CRD | CUSTOMER RELATION DEPARTMENT |
| SD | SALES DEPARTMENT |

1. Write a SQL query that retrieves the Name, Position, DeptCode, and DeptDescription of every employee.

```sql
act2.sql
1. SELECT
    EmployeeInfo.Name,
    EmployeeInfo.Position,
    EmployeeInfo.DeptCode,
    DepartmentInfo.DeptDescription
  FROM
    EmployeeInfo, DepartmentInfo
  WHERE
    EmployeeInfo.DeptCode = DepartmentInfo.DeptCode;
```

| Name | Position | DeptCode | DeptDescription |
|---|---|---|---|
| Juan Santos | Manager | BPD | BODY AND PAINT DEPARTMENT |
| Miguel Lopez | Secretary | CRD | CUSTOMER RELATION DEPARTMENT |
| Jude King | Sales | SD | SALES DEPARTMENT |
| Pedro Lao | Manager | SD | SALES DEPARTMENT |
| Jamar Perez | Sales | CRD | CUSTOMER RELATION DEPARTMENT |

2. Write a SQL query that returns the Name, Position, and total loan amount of every employee who has a loan.

```sql
act2.sql
2. SELECT
    EmployeeInfo.Name,
    EmployeeInfo.Position,
    SUM(Loan.LoanAmount) AS TotalLoan
  FROM
    EmployeeInfo, Loan
  WHERE
    EmployeeInfo.Eid = Loan.Eid
  GROUP BY
    EmployeeInfo.Name, EmployeeInfo.Position;
```

| | Name | Position | totalloan |
|---|---|---|---|
| Edit Copy Delete | Jamar Perez | Sales | 22000.00 |
| Edit Copy Delete | Juan Santos | Manager | 9000.00 |
| Edit Copy Delete | Miguel Lopez | Secretary | 2000.00 |

Click the drop-down arrow to toggle column's visibility.

3. Write a SQL query that returns the Name, Position, and Age of employees who don't have loans.

```sql
-- act2.sql
3. SELECT
    Name,
    Position,
    Age
FROM
    EmployeeInfo
WHERE
    Eid NOT IN (SELECT Eid FROM Loan);
```

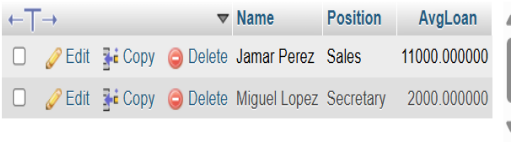

4. Write a SQL query that returns the DeptDescription and the total number of employees in each department.

```sql
-- act2.sql
4. SELECT
    DepartmentInfo.DeptDescription,
    COUNT(EmployeeInfo.Eid) AS TotalEmployees
FROM
    EmployeeInfo, DepartmentInfo
WHERE
    EmployeeInfo.DeptCode = DepartmentInfo.DeptCode
GROUP BY
    DepartmentInfo.DeptDescription;
```

| DeptDescription | TotalEmployees |
|---|---|
| BODY AND PAINT DEPARTMENT | 1 |
| CUSTOMER RELATION DEPARTMENT | 2 |
| SALES DEPARTMENT | 2 |

5. Write a SQL query that returns the Name, Position, and average loan amount of every employee from the CUSTOMER RELATION DEPARTMENT

```sql
-- act2.sql
5. SELECT
    EmployeeInfo.Name,
    EmployeeInfo.Position,
    AVG(Loan.LoanAmount) AS AvgLoan
FROM
    EmployeeInfo, DepartmentInfo, Loan
WHERE
    EmployeeInfo.DeptCode = DepartmentInfo.DeptCode
    AND EmployeeInfo.Eid = Loan.Eid
    AND DepartmentInfo.DeptDescription = 'CUSTOMER RELATION DEPARTMENT'
GROUP BY
    EmployeeInfo.Name, EmployeeInfo.Position;
```




6. Write SQL statements to create a new table named "PositionInfo" with attributes PositionID (primary key & auto_increment), Position, and Salary. Then, copy the content of the "EmployeeInfo" table (Position & Salary) into the "PositionInfo" table.

```sql
act2.sql
54  6.
55  CREATE TABLE PositionInfo (
56      PositionID INT AUTO_INCREMENT PRIMARY KEY,
57      Position VARCHAR(50),
58      Salary INT
59  );
60
61  INSERT INTO PositionInfo (Position, Salary)
62  SELECT DISTINCT Position, Salary
63  FROM EmployeeInfo;
64
```

| | PositionID | Position | Salary |
|---|---|---|---|
| Edit Copy Delete | 1 | Manager | 20000 |
| Edit Copy Delete | 2 | Secretary | 14000 |
| Edit Copy Delete | 3 | Sales | 12000 |

7. Create a SQL function that categorizes the age of an employee into brackets: 1-12 (Young), 13-19 (Teen), 20-49 (Adult), and 50 and above (Senior).
   **Sample query: SELECT CategorizeAge(Age) FROM EmployeeInfo;**

```sql
act2.sql
65  7. DELIMITER //
66
67  CREATE FUNCTION CategorizeAge(age INT)
68  RETURNS VARCHAR(20)
69  BEGIN
70    IF age BETWEEN 1 AND 12 THEN
71      RETURN 'Young';
72    ELSEIF age BETWEEN 13 AND 19 THEN
73      RETURN 'Teen';
74    ELSEIF age BETWEEN 20 AND 49 THEN
75      RETURN 'Adult';
76    ELSE
77      RETURN 'Senior';
78    END IF;
79  END //
80
81  DELIMITER ;
82
```

| | Name | Type | Returns | |
|---|---|---|---|---|
| | CategorizeAge | FUNCTION | varchar(20) | Edit Execute Export D |

8. Create a SQL procedure that retrieves the Name, Position, and DeptDescription of employees using their Employee ID. **Sample query: CALL GetEmployeeInfo(1)**;

```sql
act2.sql
83  8. DELIMITER //
84
85  CREATE PROCEDURE GetEmployeeInfo(IN empId INT)
86  BEGIN
87    SELECT
88      EmployeeInfo.Name,
89      EmployeeInfo.Position,
90      DepartmentInfo.DeptDescription
91    FROM
92      EmployeeInfo, DepartmentInfo
93    WHERE
94      EmployeeInfo.DeptCode = DepartmentInfo.DeptCode
95      AND EmployeeInfo.Eid = empId;
96  END //
97
```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0039 seconds.)

```
CREATE PROCEDURE GetEmployeeInfo(IN empId INT) BEGIN SELECT
EmployeeInfo.Name, EmployeeInfo.Position,
DepartmentInfo.DeptDescription FROM EmployeeInfo, DepartmentInfo
WHERE EmployeeInfo.DeptCode = DepartmentInfo.DeptCode AND
EmployeeInfo.Eid = empId; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

9. Create a SQL view that returns all the information from the "EmployeeInfo" table.

act2.sql

```
100   9. CREATE VIEW EmployeeInfoView AS
101   SELECT * FROM EmployeeInfo;
102
103   10. SELECT
```


10. Write a SQL query that returns the Name, DeptDescription, and total loans (including employees with no loans) of every employee.

act2.sql

```
103   10. SELECT
104     EmployeeInfo.Name,
105     DepartmentInfo.DeptDescription,
106     IFNULL(SUM(Loan.LoanAmount), 0) AS TotalLoan
107   FROM
108     EmployeeInfo
109   LEFT JOIN Loan ON EmployeeInfo.Eid = Loan.Eid
110   JOIN DepartmentInfo ON EmployeeInfo.DeptCode = DepartmentInfo.DeptCode
111   GROUP BY
112     EmployeeInfo.Name, DepartmentInfo.DeptDescription;
113
114
115
```

| Name | DeptDescription | TotalLoan |
|---|---|---|
| Jamar Perez | CUSTOMER RELATION DEPARTMENT | 22000.00 |
| Juan Santos | BODY AND PAINT DEPARTMENT | 9000.00 |
| Jude King | SALES DEPARTMENT | 0.00 |
| Miguel Lopez | CUSTOMER RELATION DEPARTMENT | 2000.00 |
| Pedro Lao | SALES DEPARTMENT | 0.00 |

**TABLE : tbl_artist**

| aid | name | talent | salary | age | program |
|---|---|---|---|---|---|
| 1 | Juan Santos | Acting | 21000 | 35 | Ang Probinsyano |
| 2 | Miguel Lopez | Singing | 75000 | 30 | The Voice Ph |
| 3 | Jude King | Dancing | 48000 | 34 | Eat Bulaga |
| 4 | Pedro Lao | Dancing | 27000 | 28 | Family Feud |
| 5 | Jamar Perez | Dancing | 120000 | 30 | Matang Lawin |
| 6 | Sandra Kim | Acting | 83000 | 21 | Ang Probinsyano |
| 7 | Manny Fork | Singing | 58000 | 40 | The Voice Ph |
| 8 | Mary Tan | Acting | 58000 | 30 | Eat Bulaga |
| 9 | Ken Kio | Singing | 63000 | 41 | Family Feud |
| 10 | Sam Jim | Acting | 53000 | 41 | Wish Ko Lang |

1. Write a query that will return the artist's name starting with the letter "J" Program.

act2.sql

```
1   SELECT name, program
2   FROM tbl_artist
3   WHERE name LIKE 'J%';
4
```

| | name | program |
|---|---|---|
| Edit Copy Delete | Juan Santos | Ang Probinsyano |
| Edit Copy Delete | Jude King | Eat Bulaga |
| Edit Copy Delete | Jamar Perez | Matang Lawin |

2. Write a query that will return the name and program of the highest-paid artist belonging "to Eat Bulaga".

```
5   SELECT name, program
6   FROM tbl_artist
7   WHERE program = 'Eat Bulaga'
8   ORDER BY salary DESC
9   LIMIT 1;
10
```

| | name | program |
|---|---|---|
| Edit Copy Delete | Mary Tan | Eat Bulaga |

3. Write a query that will return the total number of artists aged 60 or below.

```sql
SELECT COUNT(*) AS TotalArtists
FROM tbl_artist
WHERE age <= 60;
```

Your SQL query has been executed successfully.

```
SELECT COUNT(*) AS TotalArtists FROM tbl_artist WHERE age <= 60;
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Extra options

| TotalArtists |
| --- |
| 10 |

4. Write a query that will return the total number of artists belonging to the “Ang Probisyano” program.

```sql
SELECT COUNT(*) AS TotalArtists
FROM tbl_artist
WHERE program = 'Ang Probinsyano';
```

Your SQL query has been executed successfully.

```
SELECT COUNT(*) AS TotalArtists FROM tbl_artist WHERE program = 'Ang Probinsyano';
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Extra options

| TotalArtists |
| --- |
| 2 |

5. Write a query that will return the name of the artist in lowercase and the talent in lowercase format.

```sql
SELECT
  LOWER(name) AS name_lowercase,
  LOWER(talent) AS talent_lowercase
FROM tbl_artist;
```

| name_lowercase | talent_lowercase |
| --- | --- |
| juan santos | acting |
| miguel lopez | singing |
| jude king | dancing |
| pedro lao | dancing |
| jamar perez | dancing |
| sandra kim | acting |
| manny fork | singing |
| mary tan | acting |
| ken kio | singing |
| sam jim | acting |

6. Write a query that will return the name and age of artists who are 25 years old and below.

```sql
SELECT name, age
FROM tbl_artist
WHERE age <= 25;
```

| ←T→ | name | age |
| --- | --- | --- |
| Edit Copy Delete | Sandra Kim | 21 |

7. Write a query that will return the average salary of artists per program.

```sql
SELECT program, AVG(salary) AS average_salary
FROM tbl_artist
GROUP BY program;
```

| ←T→ | program | average_salary |
| --- | --- | --- |
| Edit Copy Delete | Ang Probinsyano | 52000.0000 |
| Edit Copy Delete | Eat Bulaga | 53000.0000 |
| Edit Copy Delete | Family Feud | 45000.0000 |
| Edit Copy Delete | Matang Lawin | 120000.0000 |
| Edit Copy Delete | The Voice Ph | 66500.0000 |
| Edit Copy Delete | Wish Ko Lang | 53000.0000 |

8. Write a query that will return the information of every artist in descending order by name.

```
SELECT *
FROM tbl_artist
ORDER BY name DESC;
```

| | aid | name | talent | salary | age | program |
|---|---|---|---|---|---|---|
| Edit Copy Delete | 6 | Sandra Kim | Acting | 83000 | 21 | Ang Probinsyano |
| Edit Copy Delete | 10 | Sam Jim | Acting | 53000 | 41 | Wish Ko Lang |
| Edit Copy Delete | 4 | Pedro Lao | Dancing | 27000 | 28 | Family Feud |
| Edit Copy Delete | 2 | Miguel Lopez | Singing | 75000 | 30 | The Voice Ph |
| Edit Copy Delete | 8 | Mary Tan | Acting | 58000 | 30 | Eat Bulaga |
| Edit Copy Delete | 7 | Manny Fork | Singing | 58000 | 40 | The Voice Ph |
| Edit Copy Delete | 9 | Ken Kio | Singing | 63000 | 41 | Family Feud |
| Edit Copy Delete | 3 | Jude King | Dancing | 48000 | 34 | Eat Bulaga |
| Edit Copy Delete | 1 | Juan Santos | Acting | 21000 | 35 | Ang Probinsyano |
| Edit Copy Delete | 5 | Jamar Perez | Dancing | 120000 | 30 | Matang Lawin |

9. Write a query that will return the name and salary of artists where their salary is above the average.

```
SELECT name, salary
FROM tbl_artist
WHERE salary > (
  SELECT AVG(salary) FROM tbl_artist
);
```

| | name | salary |
|---|---|---|
| Edit Copy Delete | Miguel Lopez | 75000 |
| Edit Copy Delete | Jamar Perez | 120000 |
| Edit Copy Delete | Sandra Kim | 83000 |
| Edit Copy Delete | Ken Kio | 63000 |

10. Write a query that will return the average salary of employees from the parts department.

N/A- What parts department?. Bonus na po i2 sir hehe.

***BONUS ?***

**Activity # 3: CRUD Function**

**INSTRUCTION:** Improve the schema *(If necessary)* of the given tables above ***(Activity # 2 "EmployeeInfo, LoanAmount, and DepartmentInfo").*** Write a simple application with basic CRUD in any programming/ scripting language with the following functions.

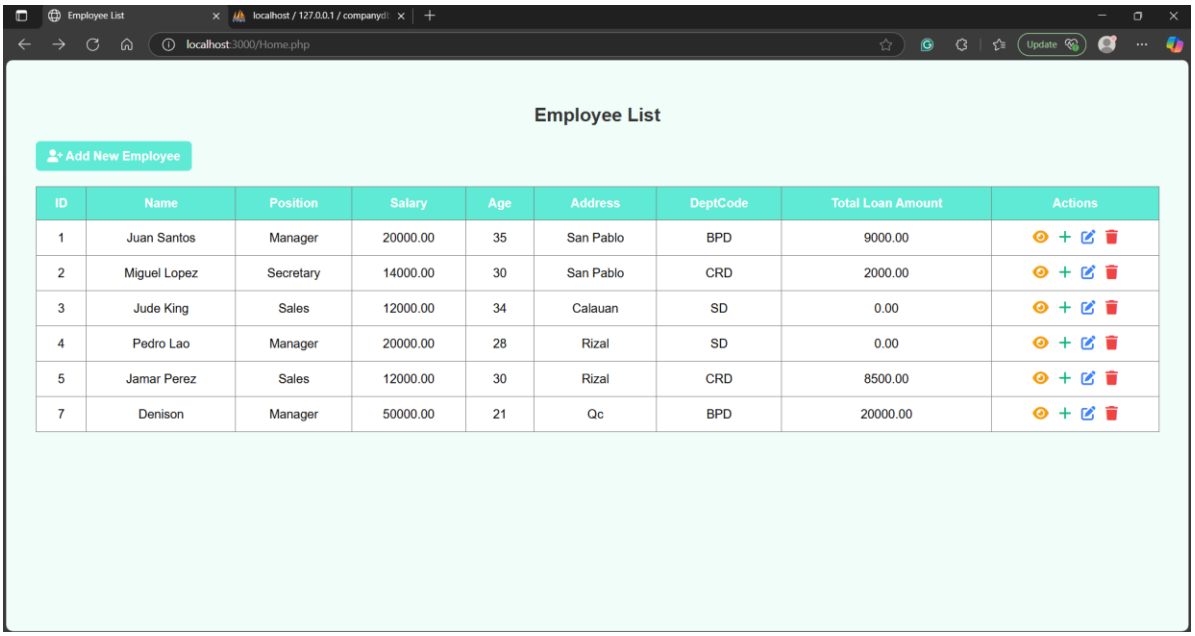
1. Retrieve employee information
2. Calculate the total loan amount for each employee.
3. Retrieve department information for a given employee.
4. Apply new loan.

- Upload the application in any paid/free web hosting service *(If the application is not web based upload the whole project in your personal Google drive account and share the link).*
- Attach the URL and screenshots of your project upon submission.

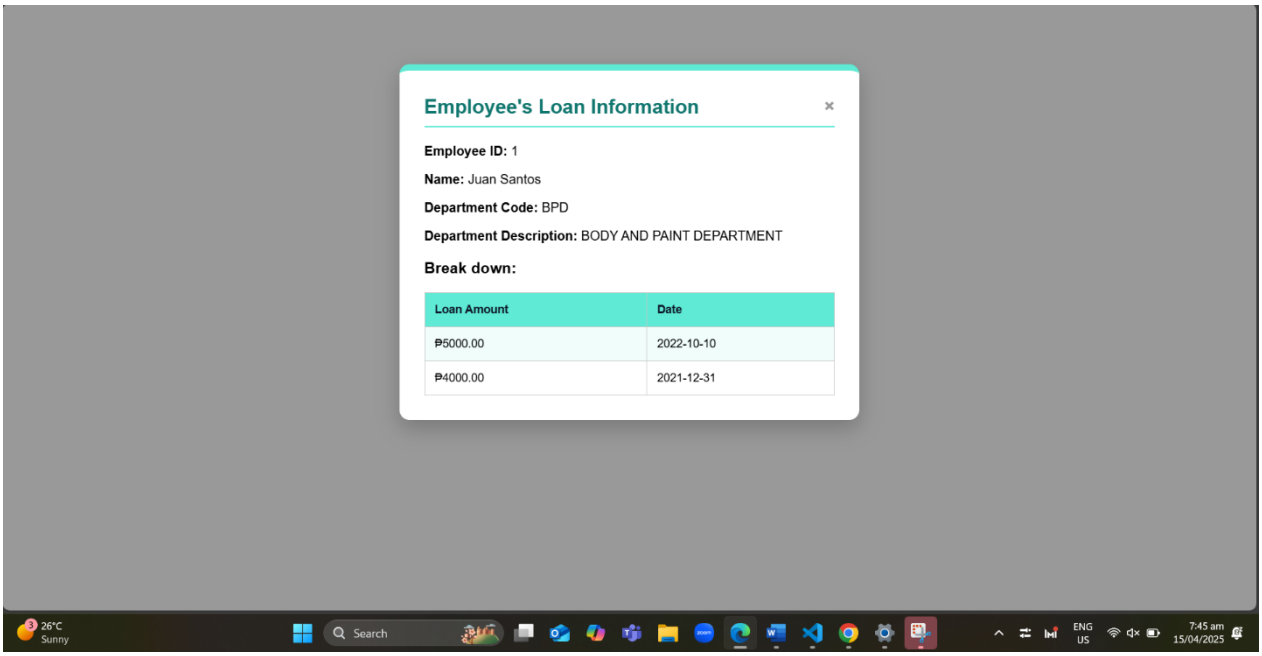
**CRUD Activity**

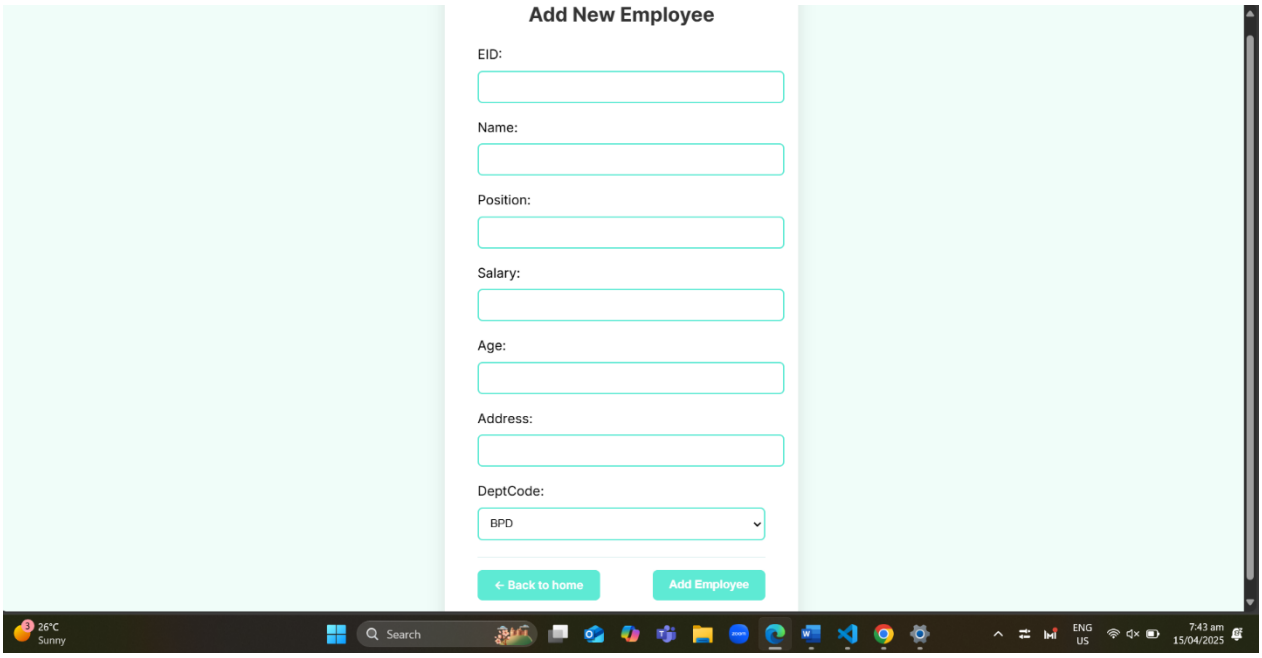
**URL: Employee List or localhost:3000/Home.php**

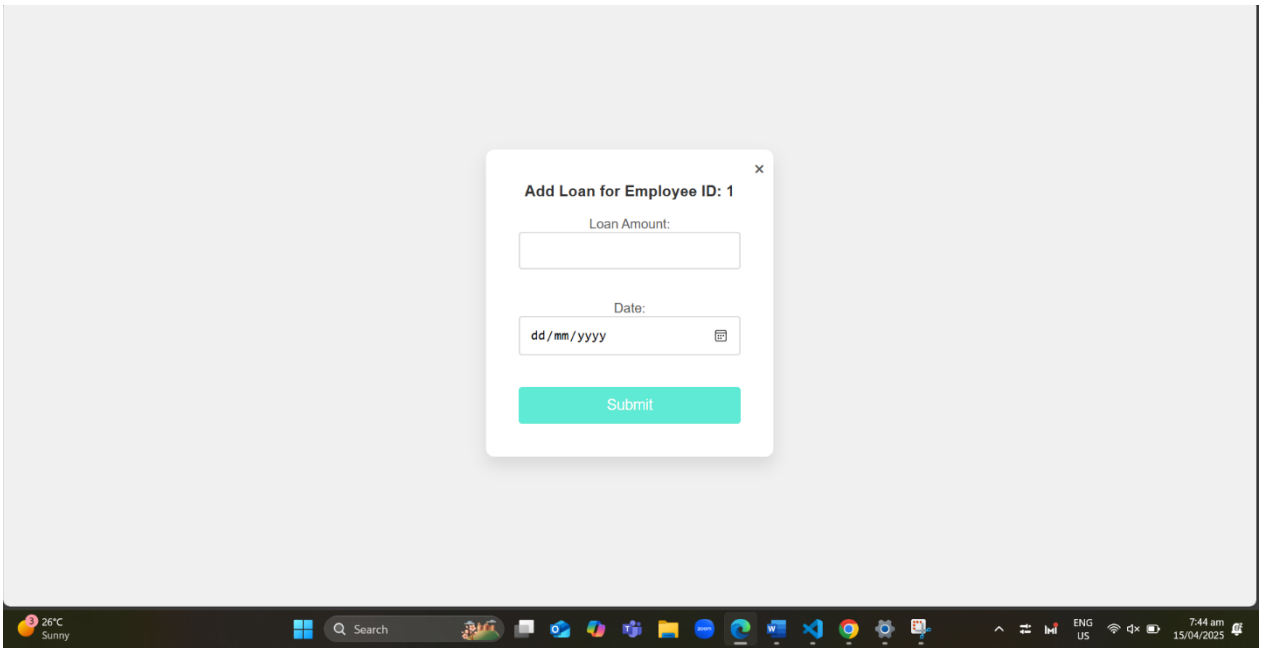
***Screenshot(s)***

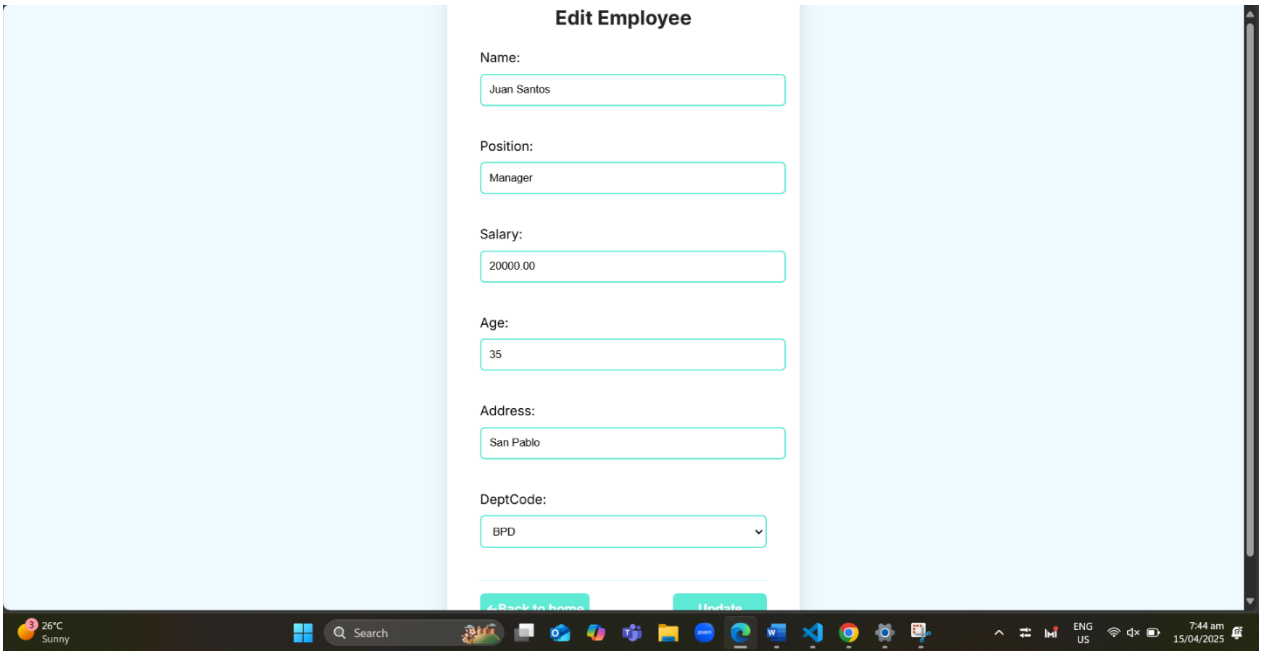


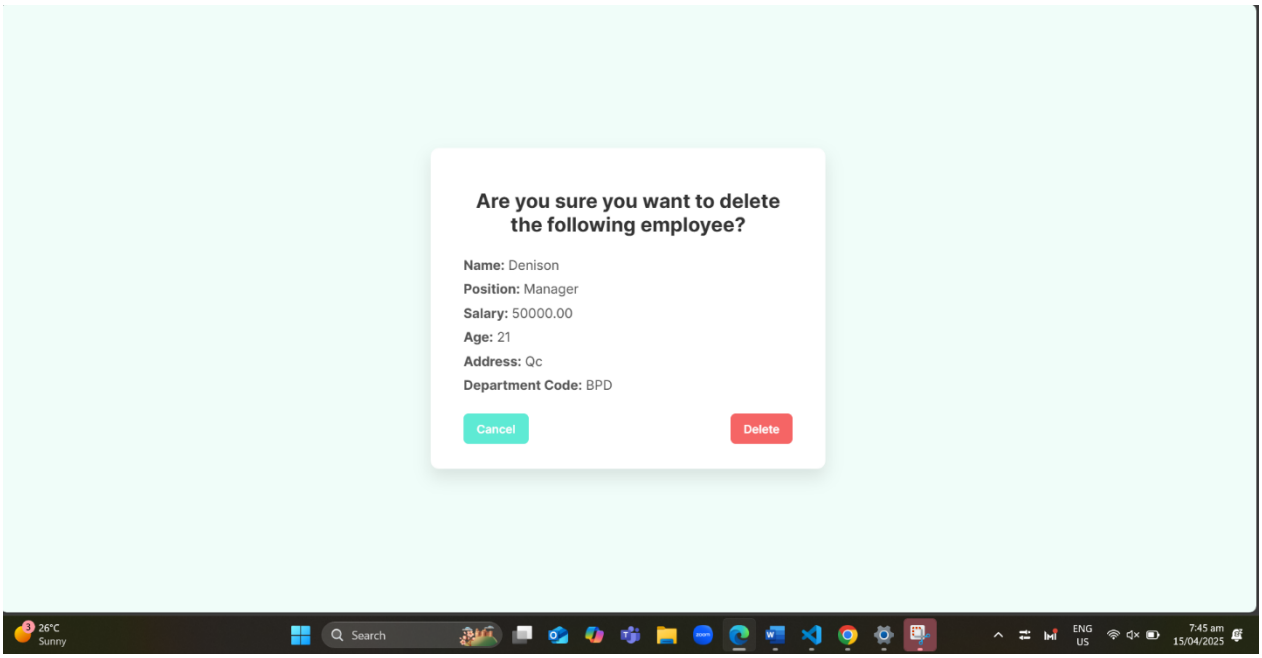

| ID | Name | Position | Salary | Age | Address | DeptCode | Total Loan Amount | Actions |
|---|---|---|---|---|---|---|---|---|
| 1 | Juan Santos | Manager | 20000.00 | 35 | San Pablo | BPD | 9000.00 | |
| 2 | Miguel Lopez | Secretary | 14000.00 | 30 | San Pablo | CRD | 2000.00 | |
| 3 | Jude King | Sales | 12000.00 | 34 | Calauan | SD | 0.00 | |
| 4 | Pedro Lao | Manager | 20000.00 | 28 | Rizal | SD | 0.00 | |
| 5 | Jamar Perez | Sales | 12000.00 | 30 | Rizal | CRD | 8500.00 | |
| 7 | Denison | Manager | 50000.00 | 21 | Qc | BPD | 20000.00 | |


Employee's Loan Information
Employee ID: 1
Name: Juan Santos
Department Code: BPD
Department Description: BODY AND PAINT DEPARTMENT
Break down:
Loan Amount
Date
₱5000.00
2022-10-10
₱4000.00
2021-12-31


Add New Employee
EID:
Name:
Position:
Salary:
Age:
Address:
DeptCode:
BPD
← Back to home
Add Employee


Add Loan for Employee ID: 1
Loan Amount:
Date:
dd/mm/yyyy
Submit
26°C
Sunny
Search
ENG
US
7:44 am
15/04/2025


Edit Employee
Name:
Juan Santos
Position:
Manager
Salary:
20000.00
Age:
35
Address:
San Pablo
DeptCode:
BPD
26°C
Sunny
Search
ENG
US
7:44 am
15/04/2025


Are you sure you want to delete the following employee?
Name: Denison
Position: Manager
Salary: 50000.00
Age: 21
Address: Qc
Department Code: BPD
Cancel
Delete
26°C
Sunny
Search
ENG
US
7:45 am
15/04/2025