Group 13

Name(s): Terry Andi Marala, Tobias Clements, Tytus Clements, Sabirin Mohamed, Syed Shuja Mukhtar, Sultan Abuhijleh

Semester: Spring 2025

Test Case 1: Update Employee Data

Description: Verify the system is updating employee data correctly when provided with employee ID and updated values.

Test Inputs:

Employee ID: 534670
Updated Values: {"JobTitle": "HR Manager", "Address": "7364 Crossing Street"}

Expected Output (Oracle):

Employee 534670 should now have job title "HR Manager".
Address "7364 Crossing Street" should be updated under Employee 534670 as well.
Updated Values should be saved in the database with no errors.

Initialization:

Verify employee 534670 exists in the database.
Run a pre-check of current values JobTitle and Address for verification.

Dependencies:

MYSQL database should be connected and running.
Employee 534670 is existent in the system.
Database is active, working effectively, and allows updates.
updateEmployee() should be working.

Test Steps:

1. Call updateEmployee("534670", updatedValues)
2. Search for Employee 534670.
3. Verify that the job title is "HR Manager".
4. Verify the address as being "7364 Crossing Street".
5. Confirm no errors have occurred.

Tear Down:

Reset updated values back to original state if needed.
Log out of database system.
Clear cookies and session data.

Test Case 2: Search Employee

Description: Verify that an employee is searchable using their SSN.

Test Inputs:

Search Type: ssn
Search Value: 7904893920

Expected Output (Oracle):
Employee info linked to SSN 7904893920 such as name, empID, salary, job title, and division should appear.
Initialization:
Check to verify if an employee with SSN 7904893920 exists in the database.
Dependencies:
MYSQL database should be connected and running.
Valid employee data needs to exist within the system.
searchEmployee() should be working and return accurate information.
Test Steps:
1. Call searchEmployee("7904893920, "ssn").
2. Verify employee information that is returned.
3. Verify the information is accurate.
4. Confirm all fields such as name, empID, salary, job title, and division are correct.
5. Confirm that no information is inaccurate or missing.
Tear Down:
Clear cookies and session data.

Test Case 3: Update Salary

Description: Verify all employees within a salary range receive raises based on set percentage.
Test Inputs:
Percentage Increase: 4.5
Min Salary: 50000
Max Salary: 100000
Expected Output (Oracle):
Employees within the min and max range salaries should receive a 4.5% raise.
Number of employees affected should be returned.
The new salaries received should be updated into the database.
Initialization:
Verify employees with salaries between the range exist.
Record the current number of employees whose salaries are between $50,000 and $100,000.
Dependencies:
Connection to the MYSQL database should be working and stable.
Salaries must be stored as numeric data.
updateSalaryByRange() should handle decimal rounding and return the number of records affected.
Test Steps:
1. Search for employees whose salaries are between the ranges of $50,000 - $100,000

2. Record employee salaries before new salary update.
3. Call updateSalaryByRange(4.5, 50000, 100000)
4. Confirm new salaries = previous salary * 1.045
5. Verify and check that the number of employees that received a raise is accurate.

Tear Down:

Reset salaries to their original state if needed.