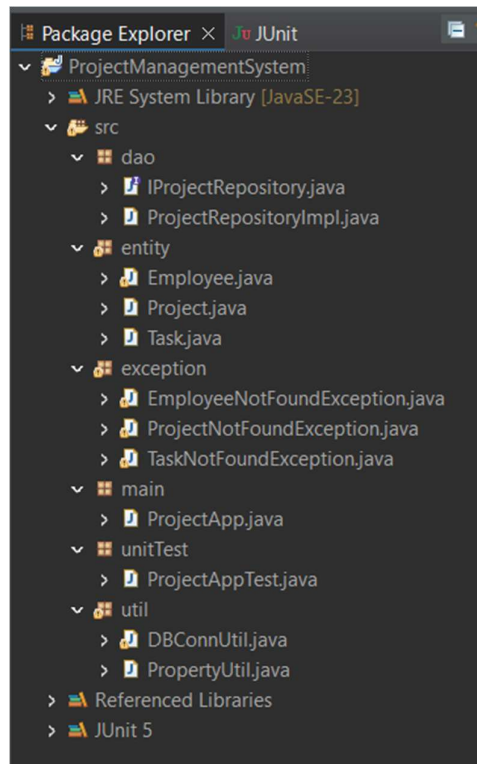


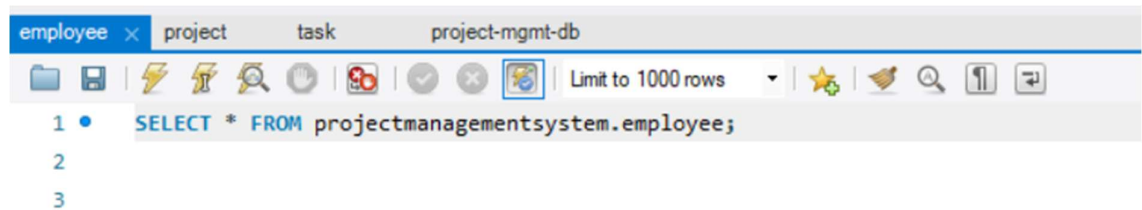
## **CASE STUDY DEVELOPMENT: PROJECT MANAGEMENT SYSTEM**

### 1. Directory Structure:



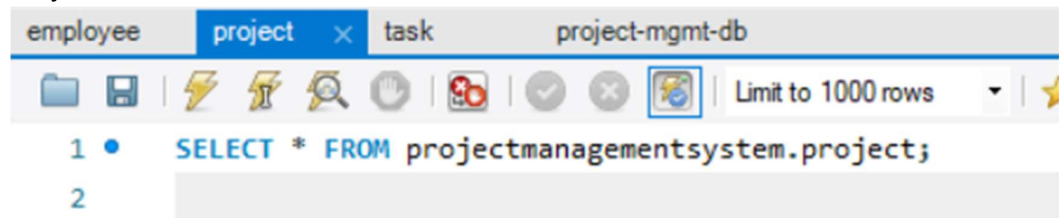
### 2. Database Schema:

- Employee table:



Result Grid						
Filter Rows:						
	employee_id	employee_name	employee_designation	employee_gender	employee_salary	project_id
▶	1	Mruganka	Developer	Female	500000	2
	2	Om	Tester	Male	540000	1
	3	Vaishnavi	Intern	Female	20000	1
	4	Sunny	Manager	Male	700000	2
*	NULL	NULL	NULL	NULL	NULL	NULL

- Project table:



Result Grid					
Filter Rows: <input type="text"/>					
	project_id	project_name	project_description	start_date	status
▶	1	Demo Project 1	testing and debugging	2025-04-04	started
	2	Demo Project 2	Developing application	2025-03-06	dev
*	NULL	NULL	NULL	NULL	NULL

- Task table:

The screenshot shows a database management interface with tabs for 'employee', 'project', and 'task'. The 'task' tab is active. Below the tabs is a toolbar with various icons. The SQL query editor contains the following text:

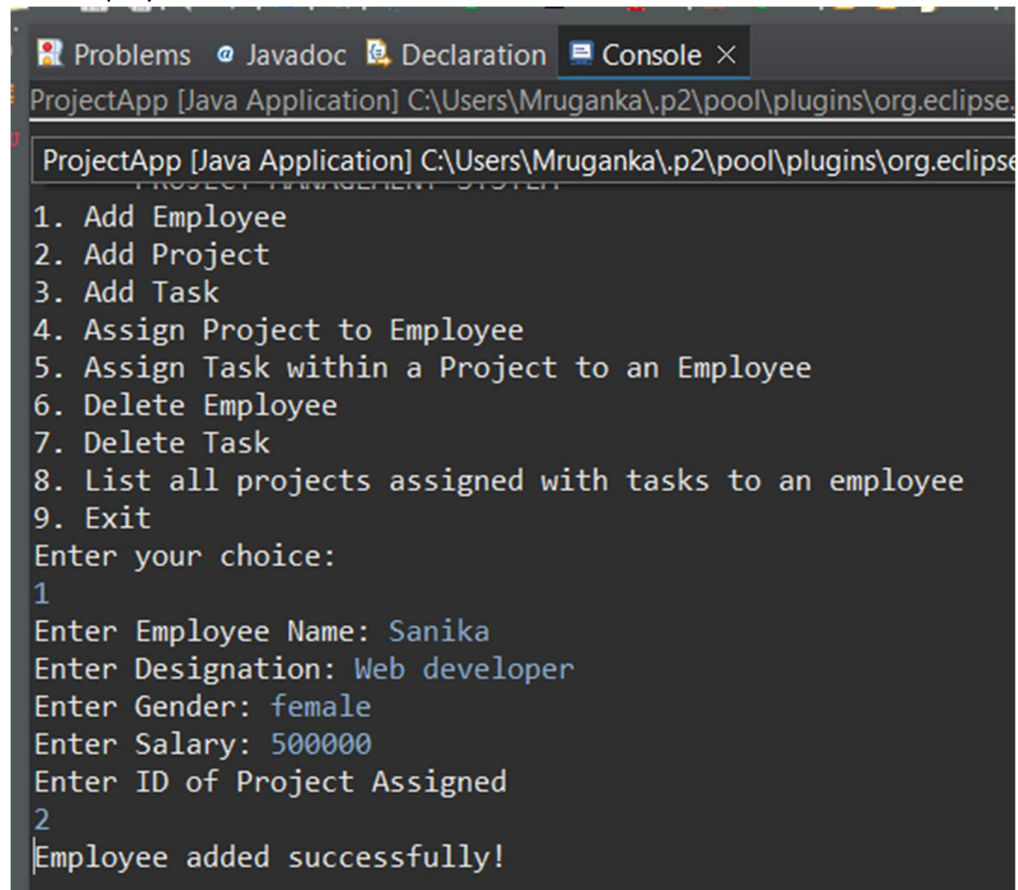
```
1 • SELECT * FROM projectmanagementsystem.task;
2
```

Below the query editor is a 'Result Grid' showing the data for the 'task' table:

Result Grid					
Filter Rows: <input type="text"/>					
	task_id	task_name	project_id	employee_id	status
▶	1	Developing application	2	1	assigned
	2	Testing application	1	2	started
	3	Research	1	3	assigned
	4	Management	2	4	completed
*	NULL	NULL	NULL	NULL	NULL

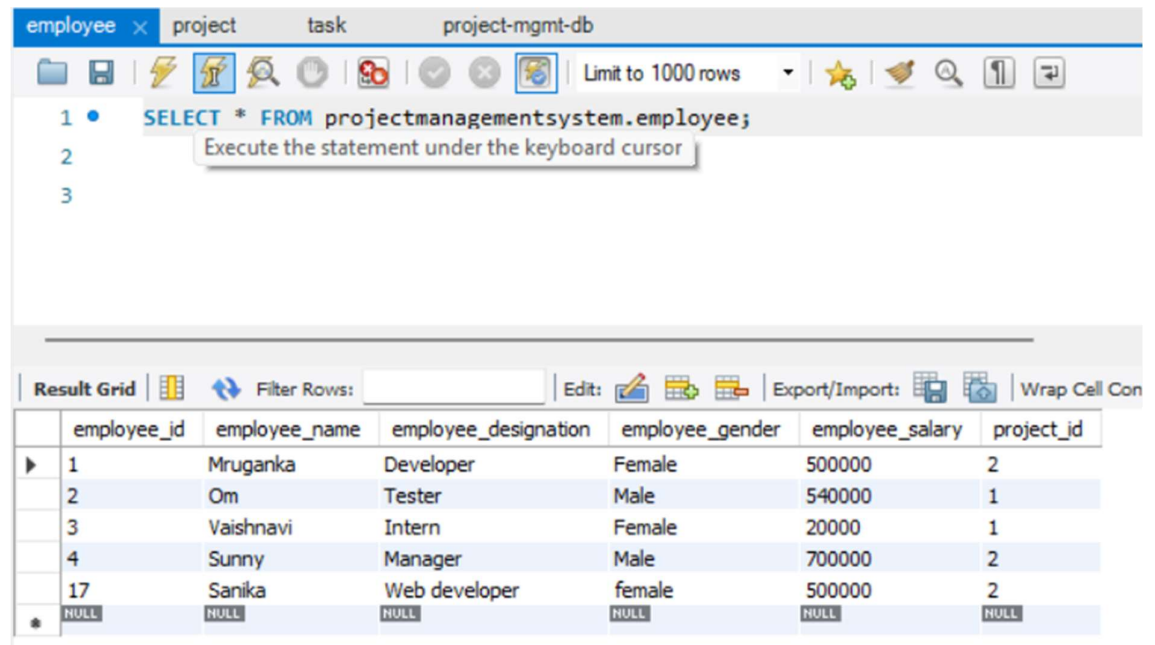
3. Output of mentioned methods:

(a) Add Employee:



```
ProjectApp [Java Application] C:\Users\Mruganka\.p2\pool\plugins\org.eclipse...
ProjectApp [Java Application] C:\Users\Mruganka\.p2\pool\plugins\org.eclipse...

1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
1
Enter Employee Name: Sanika
Enter Designation: Web developer
Enter Gender: female
Enter Salary: 500000
Enter ID of Project Assigned
2
Employee added successfully!
```



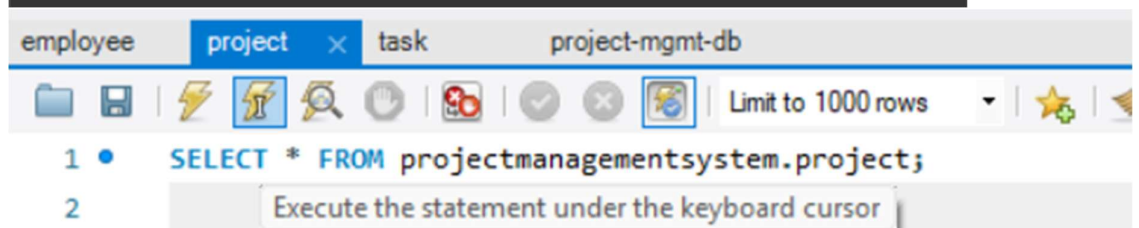
employee_id	employee_name	employee_designation	employee_gender	employee_salary	project_id
1	Mruganka	Developer	Female	500000	2
2	Om	Tester	Male	540000	1
3	Vaishnavi	Intern	Female	20000	1
4	Sunny	Manager	Male	700000	2
17	Sanika	Web developer	female	500000	2
NULL	NULL	NULL	NULL	NULL	NULL

(b) Add Project:

```

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
2
Enter Project Name: Demo project 3
Enter Project Description: JUnit Testing
Enter Start Date (YYYY-MM-DD): 2025-02-15
Enter Status (started/dev/build/test/deployed): build
Project added successfully!

```



Result Grid					
Filter Rows: <input type="text"/>					
Edit:					
Export/Import:					
	project_id	project_name	project_description	start_date	status
▶	1	Demo Project 1	testing and debugging	2025-04-04	started
	2	Demo Project 2	Developing application	2025-03-06	dev
	14	Demo project 3	JUnit Testing	2025-02-15	build
*	NULL	NULL	NULL	NULL	NULL

(c) Add Task

```

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
3
Enter Task Name: JUnit test cases
Enter Project ID: 14
Enter Employee ID: 1
Enter Task Status (Assigned/Started/Completed): completed
Task added successfully!

```

employee project task x project-mgmt-db

Limit to 1000 rows

1 • SELECT \* FROM projectmanagementsystem.task;

2 Execute the statement under the keyboard cursor

Result Grid					
Filter Rows:					
Edit:					
Export/					
	task_id	task_name	project_id	employee_id	status
▶	1	Developing application	2	1	assigned
	2	Testing application	1	2	started
	3	Research	1	3	assigned
	4	Management	2	4	completed
	12	JUnit test cases	14	1	completed
*	NULL	NULL	NULL	NULL	NULL

(d) Assign project to employee:



```
Problems Javadoc Declaration Console X
ProjectApp [Java Application] C:\Users\Mruganka\p2\pool\plugins\org.eclipse

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
4
Enter Project ID: 14
Enter Employee ID: 2
Project assigned to employee successfully!
```

employee x project task project-mgmt-db

Limit to 1000 rows

```
1 • SELECT * FROM projectmanagementsystem.employee;
2
3
```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	employee_id	employee_name	employee_designation	employee_gender	employee_salary	project_id
▶	1	Mruganka	Developer	Female	500000	2
	2	Om	Tester	Male	540000	14
	3	Vaishnavi	Intern	Female	20000	1
	4	Sunny	Manager	Male	700000	2
	17	Sanika	Web developer	female	500000	2
*	NULL	NULL	NULL	NULL	NULL	NULL

(e) Assign task within a project to employee:

Problems Javadoc Declaration Console ×

ProjectApp [Java Application] C:\Users\Mruganka\.p2\pool\plugins\org.eclipse.j

```

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
5
Enter Task ID: 12
Enter Project ID: 14
Enter Employee ID: 1
Task assigned successfully!

```

employee project task × project-mgmt-db

Limit to 1000 rows

```

1 • SELECT * FROM projectmanagementsystem.task;
2

```

Result Grid					
Filter Rows:					
Edit: Export/Import:					
	task_id	task_name	project_id	employee_id	status
▶	1	Developing application	2	1	assigned
	2	Testing application	1	2	started
	3	Research	1	3	assigned
	4	Management	2	4	completed
	12	JUnit test cases	14	1	completed
*	NULL	NULL	NULL	NULL	NULL

(f) Delete employee:

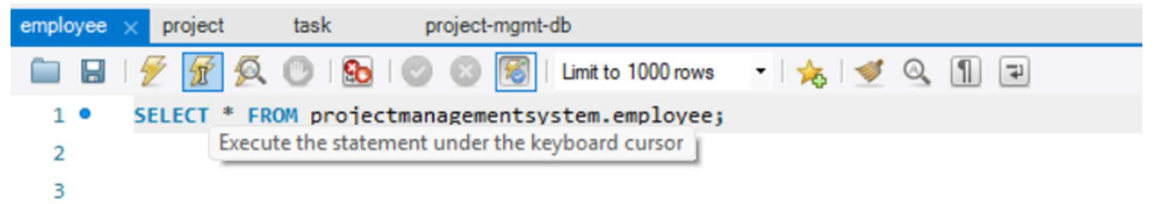
```
Problems Javadoc Declaration Console X
ProjectApp [Java Application] C:\Users\Mruganka\.p2\pool\plugins\org.eclipse

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
6
Enter Employee ID to delete: 23
Error: Employee with id 23 not found
```

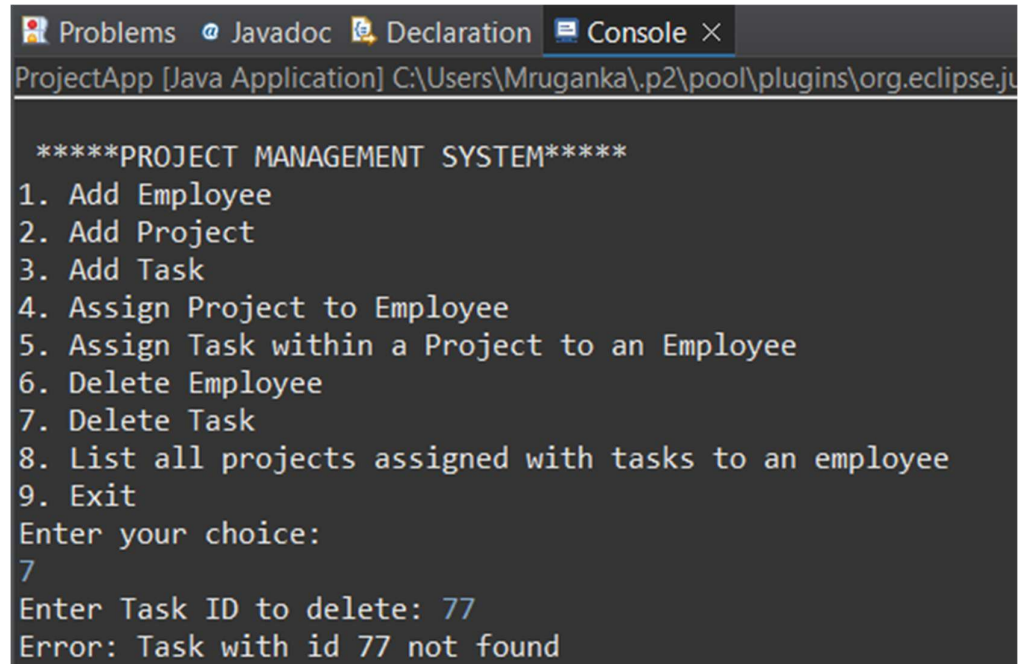
```
Problems Javadoc Declaration Console X
ProjectApp [Java Application] C:\Users\Mruganka\.p2\pool\plugins\org.eclipse.justj.c

*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
6
Enter Employee ID to delete: 17
Employee deleted successfully.
```





(g) Delete task:



Problems Javadoc Declaration Console ×

ProjectApp [Java Application] C:\Users\Mruganka\p2\pool\plugins\org.eclipse.justj.openj

```
*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
7
Enter Task ID to delete: 12
Task deleted successfully.
```

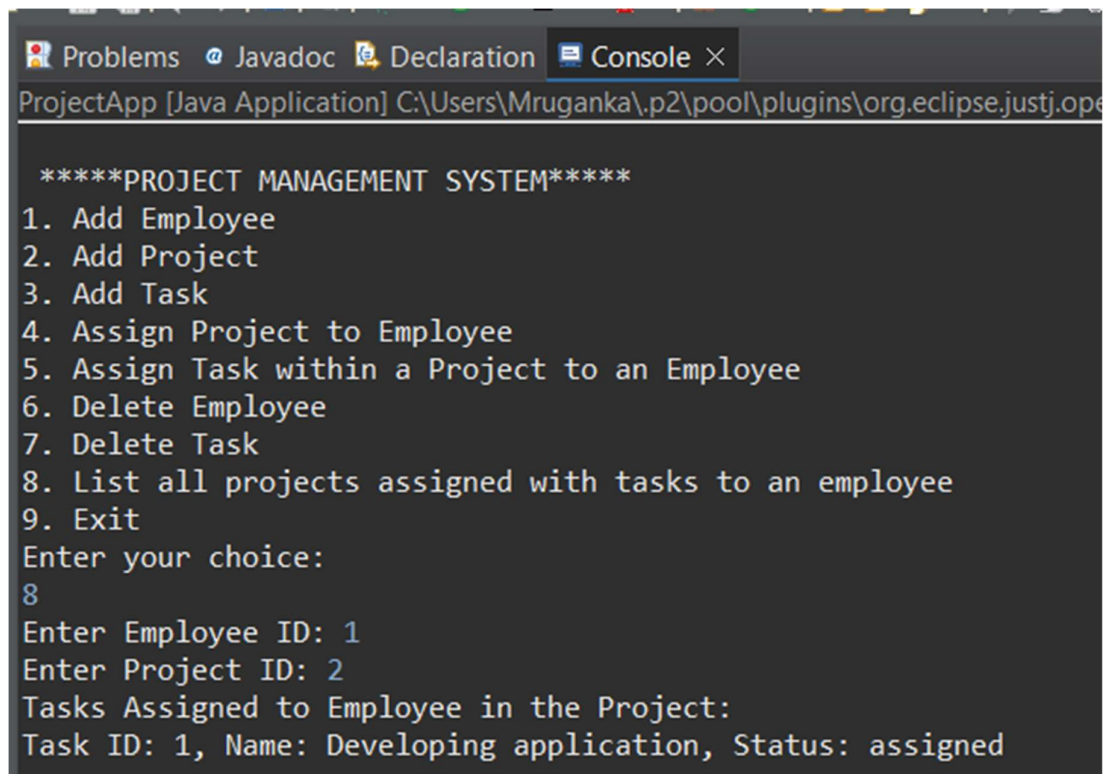
employee project task × project-mgmt-db

Limit to 1000 rows

```
1 • SELECT * FROM projectmanagementsystem.task;
2
```

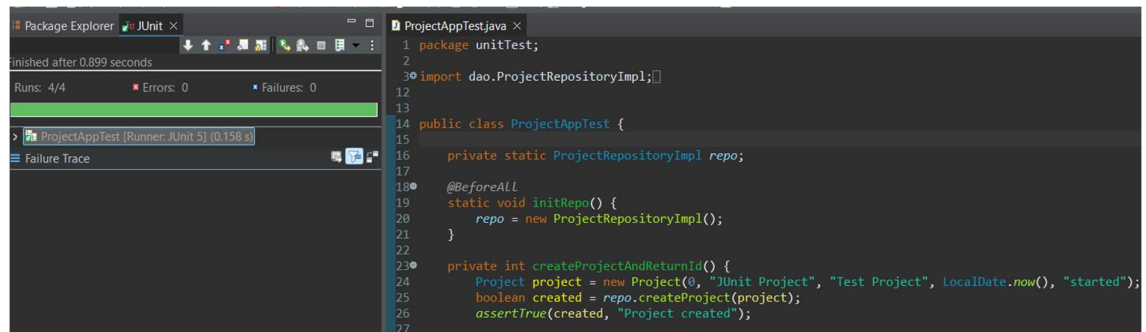
Result Grid					
Filter Rows:					
Edit:					
Export/Import					
	task_id	task_name	project_id	employee_id	status
▶	1	Developing application	2	1	assigned
	2	Testing application	1	2	started
	3	Research	1	3	assigned
	4	Management	2	4	completed
*	NULL	NULL	NULL	NULL	NULL

(h) List all projects assigned with tasks to an employee:



```
*****PROJECT MANAGEMENT SYSTEM*****
1. Add Employee
2. Add Project
3. Add Task
4. Assign Project to Employee
5. Assign Task within a Project to an Employee
6. Delete Employee
7. Delete Task
8. List all projects assigned with tasks to an employee
9. Exit
Enter your choice:
8
Enter Employee ID: 1
Enter Project ID: 2
Tasks Assigned to Employee in the Project:
Task ID: 1, Name: Developing application, Status: assigned
```

4. Junit Testing:  
All 4 test cases executed.



```
1 package unitTest;
2
3 import dao.ProjectRepositoryImpl;
4
5 public class ProjectAppTest {
6
7     private static ProjectRepositoryImpl repo;
8
9     @BeforeAll
10    static void initRepo() {
11        repo = new ProjectRepositoryImpl();
12    }
13
14    private int createProjectAndReturnId() {
15        Project project = new Project(0, "JUnit Project", "Test Project", LocalDate.now(), "started");
16        boolean created = repo.createProject(project);
17        assertTrue(created, "Project created");
18    }
19 }
```