Finals Lab Task 3 – MySQL Basics

Multi-Level Company Database

1. SQL Statements (Tasks 1–5)

```
CREATE TABLE employees (
    employee_id INT AUTO_INCREMENT PRIMARY KEY,
    employee_name VARCHAR(255) NOT NULL,
    manager_id INT,
    FOREIGN KEY (manager_id) REFERENCES employees(employee_id)
);

CREATE TABLE departments (
    department_id INT AUTO_INCREMENT PRIMARY KEY,
    department_name VARCHAR(255) NOT NULL
);

CREATE TABLE employee_departments (
    employee_id INT,
    department_id INT,
    PRIMARY KEY (employee_id, department_id),
    FOREIGN KEY (employee_id) REFERENCES employees(employee_id),
    FOREIGN KEY (department_id) REFERENCES departments(department_id)
);

CREATE TABLE employee_projects (
    employee_id INT,
    project_name VARCHAR(255) NOT NULL,
    FOREIGN KEY (employee_id) REFERENCES employees(employee_id)
);

CREATE TABLE managers (
    manager_id INT AUTO_INCREMENT PRIMARY KEY,
    employee_id INT,
    FOREIGN KEY (employee_id) REFERENCES employees(employee_id)
);
```

2. Table Structures

```
EMPLOYEES
- employee_id (INT, PK, AUTO_INCREMENT)
- employee_name (VARCHAR 255, NOT NULL)
- manager_id (INT, FK → employees.employee_id)

DEPARTMENTS
- department_id (INT, PK, AUTO_INCREMENT)
- department_name (VARCHAR 255, NOT NULL)

EMPLOYEE_DEPARTMENTS
- employee_id (INT, FK → employees.employee_id)
- department_id (INT, FK → departments.department_id)
- PRIMARY KEY (employee_id, department_id)

EMPLOYEE_PROJECTS
- employee_id (INT, FK → employees.employee_id)
- project_name (VARCHAR 255, NOT NULL)

MANAGERS
- manager_id (INT, PK, AUTO_INCREMENT)
- employee_id (INT, FK → employees.employee_id)
```

3. Relational Schema (Text-Based ER Diagram)

```
EMPLOYEES (employee_id PK) ← MANAGERS (employee_id FK)
        ↑
        ■
EMPLOYEE_DEPARTMENTS (employee_id FK, department_id FK)
```

DEPARTMENTS (department_id PK)

EMPLOYEES (employee_id FK) → EMPLOYEE_PROJECTS (employee_id FK)

4. SQL Copy of Database

See attached SQL file: finals_lab_task3.sql