Database: CompanyManagementSystem

Employee		
Column Name	**Data Type**	**Description**
Employee_Id	INT	Primary key, unique identifier
First_Name	VARCHAR(50)	First name of the employee
Last_Name	VARCHAR(50)	Last name of the employee
Email	VARCHAR(100)	Email address of the employee
Phone_Number	VARCHAR(15)	Phone number of the employee
Hire_Date	DATE	Date when the employee was hired
Job_Title	VARCHAR(50)	Job title of the employee
Salary	DECIMAL(10, 2)	Salary of the employee
Department_Id	INT	Foreign key to Department table

Department		
Column Name	**Data Type**	**Description**
Department_Id	INT	Primary key, unique identifier
Department_Name	VARCHAR(100)	Name of the department

Project		
Column Name	**Data Type**	**Description**
Project_Id	INT	Primary key, unique identifier
Project_Name	VARCHAR(100)	Name of the project
Start_Date	DATE	Start date of the project
End_Date	DATE	End date of the project
Budget	DECIMAL(12, 2)	Budget allocated for the project

EmployeeProject		
Column Name	**Data Type**	**Description**
EmployeeProject_Id	INT	Primary key, unique identifier
Employee_Id	INT	Foreign key to Employee table
Project_Id	INT	Foreign key to Project table

Prepare Stored Procedures as Below:

1. Write a stored procedure to insert a new employee into the `Employee` table. Include parameters for first name, last name, email, phone number, hire date, job title, salary, and department ID.
2. Implement a stored procedure to retrieve an employee's details from the `Employee` table based on the employee ID provided as a parameter.
3. Develop a stored procedure that updates an existing employee's information in the `Employee` table. Include parameters for employee ID, first name, last name, email, phone number, hire date, job title, salary, and department ID.
4. Create a stored procedure to delete an employee from the `Employee` table based on the employee ID provided as a parameter.
5. Write a stored procedure to fetch all employees belonging to a specific department from the `Employee` table. Use the department ID as a parameter.
6. Develop a stored procedure to retrieve all projects assigned to a specific employee from the `Project` table. Use the employee ID as a parameter.
7. Implement a stored procedure to fetch all employees who are assigned to a particular project from the `Employee` table. Use the project ID as a parameter.
8. Create a stored procedure to fetch details of a specific department from the `Department` table along with the count of employees in that department. Use the department ID as a parameter.
9. Write a stored procedure to calculate and return the total budget allocated to projects within a specific department from the `Project` table. Use the department ID as a parameter.
10. Write a stored procedure to retrieve all employees with their Department and Project information.