**Project Report: Software House Database**

**Overview of Project**

The software house database is designed to manage various aspects of a software development company, including client and employee information, project management, and team organization. It aims to streamline operations, improve communication, and enhance efficiency within the organization.

**Conceptual Design and tables:**

**Clients:** Stores information about clients who engage with the software house for services.

**Customers:** Similar to clients, this table may contain additional details about customers who utilize the software or services provided.

**Developers:** Contains data about developers employed by the software house, including their skills and contact information.

**Employees:** General table to store information about all employees, including developers, managers, and other staff.

**Managers:** Holds details about managerial staff within the organization.

**Projects:** Stores information related to ongoing projects, including project names, descriptions, and deadlines.

**Project\_Clients:** A relational table linking projects with their respective clients.

**Project\_Employees:** Associates employees with specific projects, tracking their roles and contributions.

**Task\_Comments:** Stores comments or notes related to specific tasks within projects.

**Tasks**: Contains details about individual tasks within projects, including task descriptions, deadlines, and assigned employees.

**Team\_Employees:** A relational table connecting employees with their respective teams.

**Technical\_Teams**: Stores information about technical teams within the organization, including team names and descriptions.

**Conceptual tables:**

|  |
| --- |
| **Clients** |
| Client\_id  Name  Contact info  Company  ….. |

|  |
| --- |
| **Customers** |
| Customer\_id  Name  Contact info  …. |

|  |
| --- |
| **Developers** |
| Developer\_id  Name  Contact info  Skills  ….. |

|  |
| --- |
| **Employees** |
| Employee\_id  Name  Contact info  Position  …. |

|  |
| --- |
| **Manager** |
| Manager\_id  Name  Contact info  Department  …… |

|  |
| --- |
| **Projects** |
| Project\_id  Name  Description  Deadline  Status  …… |

|  |
| --- |
| **Project\_clients** |
| Project\_Client\_id  Project\_id  Client\_id  ….. |

|  |
| --- |
| **Project\_employees** |
| Project\_employee\_id  Project\_id  Employee\_id  Role  …… |

|  |
| --- |
| **Task\_comments** |
| Task\_id  Project\_id  Description  Deadline  Status  …… |

|  |
| --- |
| **Team\_employees** |
| Team\_employee\_id  Team\_id  Employee\_id  …… |

|  |
| --- |
| **Technical\_teams** |
| Team\_id  Name  Description  …… |

**Logical Design**

The logical design of the database involves defining the structure and relationships between tables, as well as implementing constraints and indexes for data integrity and performance optimization. Here are some considerations:

**Primary Keys**: Each table should have a primary key to uniquely identify records.

**Foreign Keys**: Foreign keys establish relationships between tables, ensuring referential integrity.

**Indexes:** Indexes can be applied to columns frequently used in queries to improve search performance.

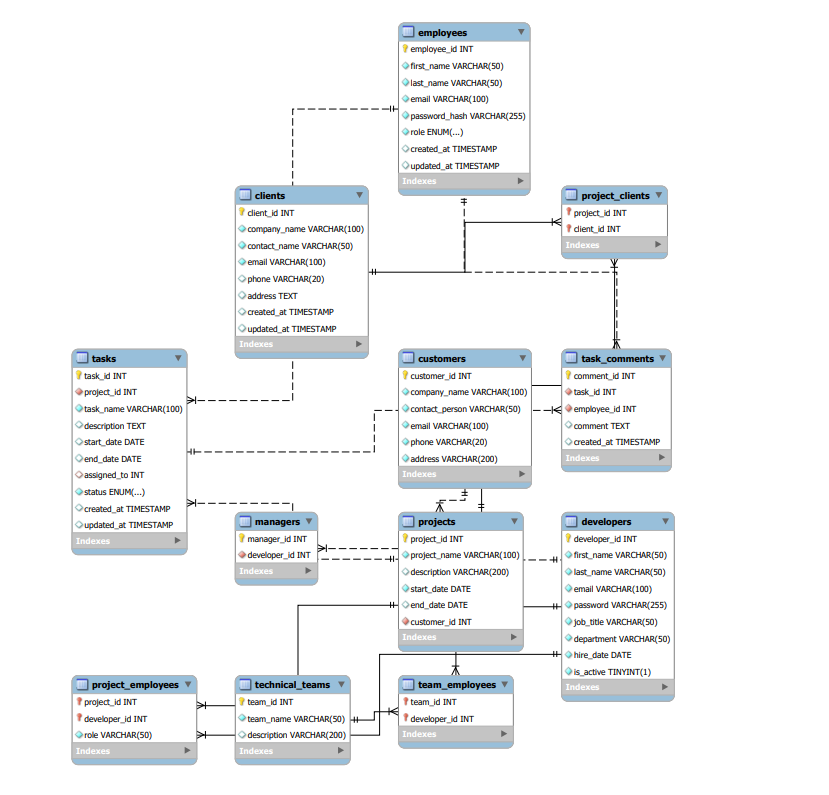
**Normalization:** The database should be normalized to minimize redundancy and dependency issues.

**Constraints:** Constraints such as NOT NULL, UNIQUE, and CHECK can be applied to enforce data integrity rules.

**Views:** Views can be created to simplify complex queries or provide customized data perspectives.

**Stored Procedures and Triggers**: These can be utilized for implementing business logic and automating certain tasks within the database.

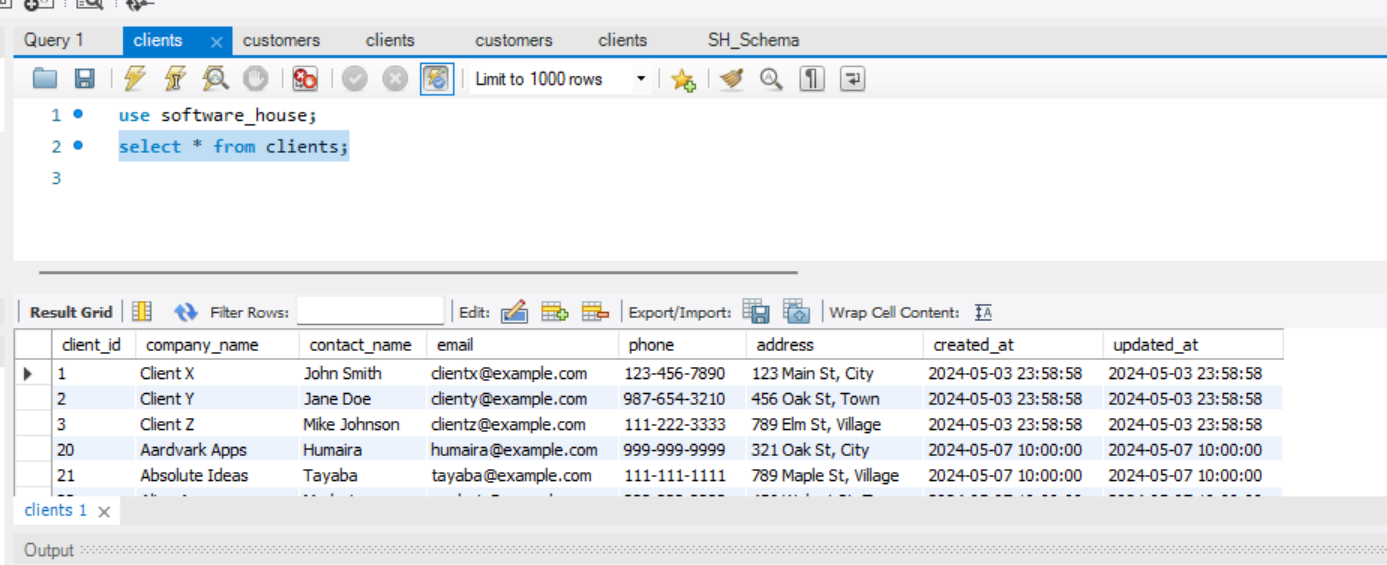
**ER Diagram of Software House Schema**



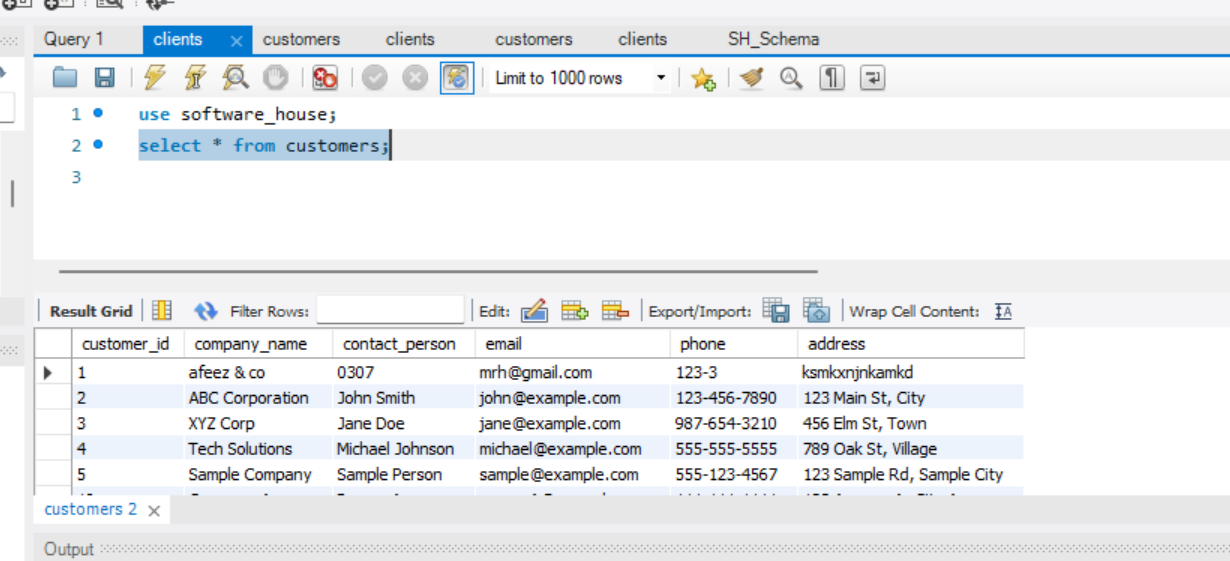
**Conclusion**

The software house database schema provides a solid foundation for managing various aspects of the organization, including clients, employees, projects, and tasks. By following sound database design principles, such as normalization and establishing appropriate relationships between entities, the database ensures data integrity and facilitates efficient operations within the software house.

**Some query**

****

2)



3) 