**ST. XAVIER’S COLLEGE**

Maitighar, Kathmandu



**Database Management System**

**Lab Assignment #3**

**Submitted by**

Shaurabh Chapagain

013BSCSIT037

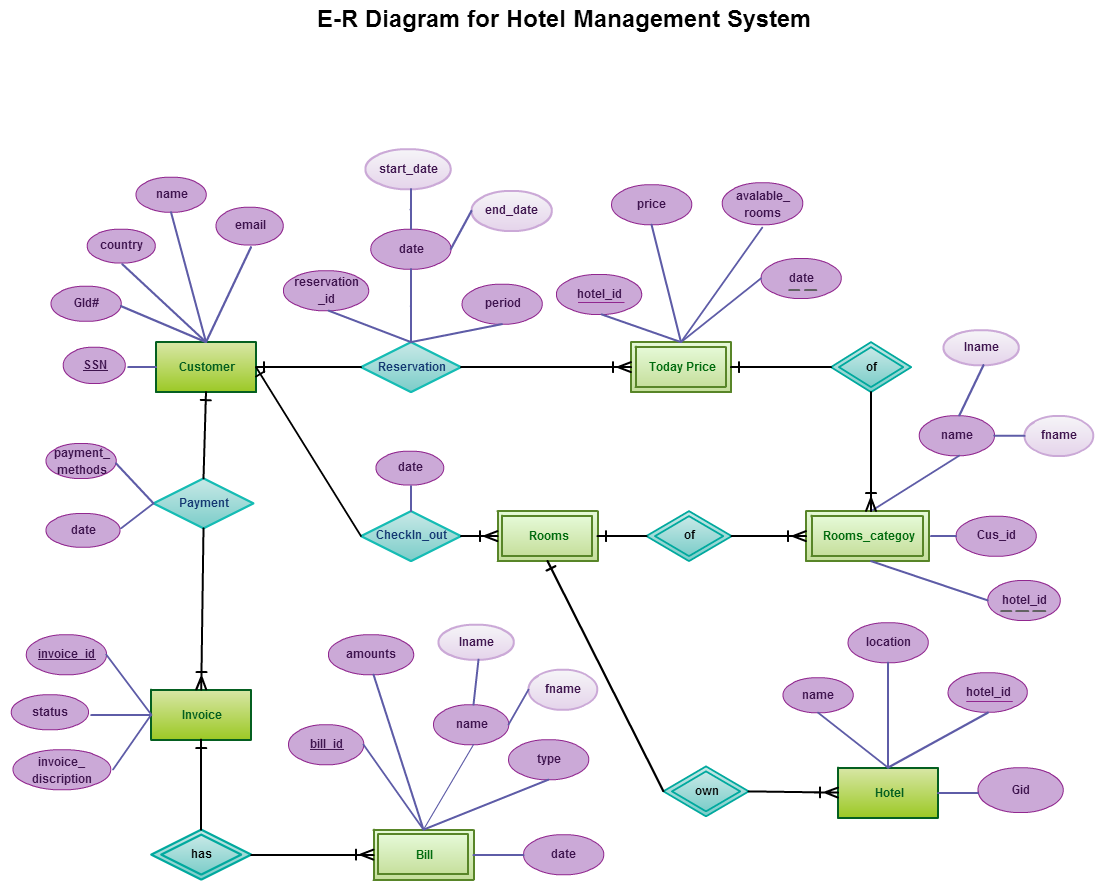
**Submitted to**

**Er. Sanjay Kumar Yadav**

Lecturer

Dept. of Computer Science

St. Xavier’s Colleg



**Design**

Design is the process of producing a detailed model of something. In database this refers to how data are constructed into the database.

Some design approaches are given below:

**Functional design**

Functional Design is a paradigm used to simplify the design of hardware and software devices such as computer software and increasingly, 3D models. A functional design assures that each modular part of a device has only one responsibility and performs that responsibility with the minimum of side effects on other parts.

**Database design**

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database.

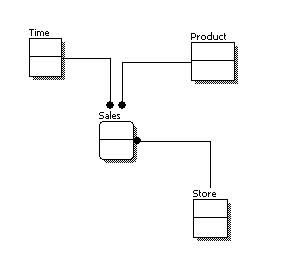
Database design concepts are as follow:

**Conceptual database design:**

A conceptual data model identifies the highest-level relationships between the different entities. Features of conceptual data model include:

* Includes the important entities and the relationships among them.
* No attribute is specified.
* No primary key is specified.

The figure below is an example of a conceptual data model.



From the figure above, we can see that the only information shown via the conceptual data model is the entities that describe the data and the relationships between those entities. No other information is shown through the conceptual data model.

**Logical database design**

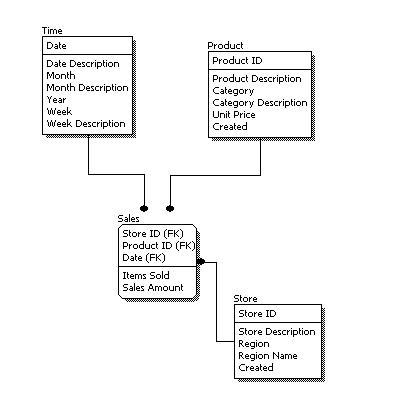
A logical data model describes the data in as much detail as possible, without regard to how they will be physical implemented in the database. Features of a logical data model include:

* Includes all entities and relationships among them.
* All attributes for each entity are specified.
* The primary key for each entity is specified.
* Foreign keys (keys identifying the relationship between different entities) are specified.
* Normalization occurs at this level.

The steps for designing the logical data model are as follows:

1. Specify primary keys for all entities.
2. Find the relationships between different entities.
3. Find all attributes for each entity.
4. Resolve many-to-many relationships.
5. Normalization.

The figure below is an example of a logical data model.



Comparing the logical data model shown above with the [conceptual data model](http://www.1keydata.com/datawarehousing/conceptual-data-model.html) diagram, we see the main differences between the two:

* In a logical data model, primary keys are present, whereas in a conceptual data model, no primary key is present.
* In a logical data model, all attributes are specified within an entity. No attributes are specified in a conceptual data model.
* Relationships between entities are specified using primary keys and foreign keys in a logical data model. In a conceptual data model, the relationships are simply stated, not specified, so we simply know that two entities are related, but we do not specify what attributes are used for this relationship.

**Physical database design**

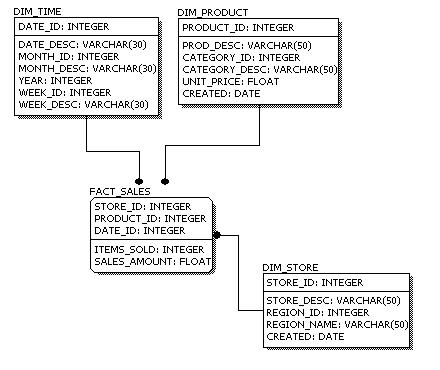
Physical data model represents how the model will be built in the database. A physical database model shows all table structures, including column name, column data type, column constraints, primary key, foreign key, and relationships between tables. Features of a physical data model include:

* Specification all tables and columns.
* Foreign keys are used to identify relationships between tables.
* De-normalization may occur based on user requirements.
* Physical considerations may cause the physical data model to be quite different from the logical data model.
* Physical data model will be different for different RDBMS. For example, data type for a column may be different between MySQL and SQL Server.

The steps for physical data model design are as follows:

1. Convert entities into tables.
2. Convert relationships into foreign keys.
3. Convert attributes into columns.
4. Modify the physical data model based on physical constraints / requirements.

The figure below is an example of a physical data model.



Comparing the physical data model shown above with the [logical data model](http://www.1keydata.com/datawarehousing/logical-data-model.html) diagram, we see the main differences between the two:

* Entity names are now table names.
* Attributes are now column names.
* Data type for each column is specified. Data types can be different depending on the actual database being used.

**Characteristics of relation**