# ST.XAVIER’S COLLEGE

# MAITIGHAR, KATHMANDU

****

**ASSIGNMENT #4**

**Database Management System**

**Submitted By:**

Pratibha Panta

012BSCCSIT028

4th Sem, 2nd Year

**Submitted To:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

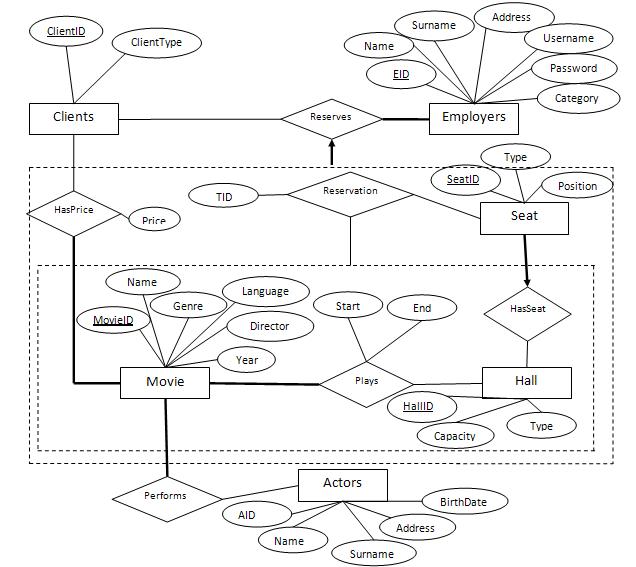
Er. Sanjay Kumar Yadav

Department of Computer Science

Lecturer

**E-R diagram with one case study**

**Eg. Case study of Cinema Hall**



Here, clients ID, EID, Movie ID and Hall ID are the primary keys.

A **primary key**, also called a primary keyword, is a key in a relational database that is unique for each record. It is a unique identifier, such as a driver license number, telephone number (including area code), or vehicle identification number (VIN). A relational database must always have one and only one primary key.

Items in the ellipse represent the attributes.

An **attribute** is a characteristic. In a database management system (DBMS), an attribute refers to a database component, such a table. It also may refer to a database field. Attributes describe the instances in the row of a database.

Items in the rectangles represent the entity sets.

An **entity** is a thing or object of importance about which data must be captured. All things aren't entities—only those about which information should be captured. Information about an entity is captured in the form of attributes and/or relationships. If something is a candidate for being an entity and it has no attributes or relationships, it isn't an entity.

Relationship sets are represented with the diamonds.

A **relationship** is an association among two or more entities. A relationship set is a set of relationship of same type.

**Lines** link the attributes to entity set and entity set to the relationship sets.

Designing a database

**Functional design**: A database which uses a functional language as its query language.

The functional database model is used to support analytics applications such as Financial Planning and Performance Management. The functional database model, or the functional model for short, is different from but complementary to, the relational model. The functional model is also distinct from other similarly named concepts, including the DAPLEX functional database model, [1] and functional language databases.

**Conceptual Database design**: Once all the requirements have been collected and analyzed, the next step is to create a conceptual schema for the database, using a high level conceptual data model. This phase is called conceptual design.

The result of this phase is an Entity-Relationship (ER) diagram or UML class diagram. It is a high-level data model of the specific application area. It describes how different entities (objects, items) are related to each other. It also describes what attributes (features) each entity has. It includes the definitions of all the concepts (entities, attributes) of the application area.

During or after the conceptual schema design, the basic data model operations can be used to specify the high-level user operations identified during the functional analysis. This also serves to confirm that the conceptual schema meets all the undefined functional requirements.[2]

**Logical Database design**: The result of the logical design phase (or data model mapping phase) is a set of relation schemas. The ER diagram or class diagram is the basis for these relation schemas.

To create the relation schemas is quite a mechanical operation. There are rules how the ER model or class diagram is transferred to relation schemas.

The relation schemas are the basis for table definitions. In this phase (if not done in previous phase) the primary keys and foreign keys are defined.

**Physical database design:** The goal of the last phase of database design, physical design, is to implement the database. At this phase one must know which database management system (DBMS) is used. For example, different DBMS's have different names for data types and have different data types.

**Characteristics of Relation**

* Data within a column must be accessible by specifying the table name, the column name, and the value of the primary key of the row.
* The DBMS must support missing and inapplicable information in a systematic way, distinct from regular values and independent of data type.
* The DBMS must support an active on-line catalogue.
* The DBMS must support at least one language that can be used independently and from within programs, and supports data definition operations, data manipulation, constraints, and transaction management.
* Views must be updatable by the system.
* The DBMS must support insert, update, and delete operations on sets.
* The DBMS must support logical data independence.
* The DBMS must support physical data independence.
* Integrity constraints must be stored within the catalogue, separate from the application.
* The DBMS must support distribution independence. The existing application should run when the existing data is redistributed or when the DBMS is redistributed.

**E-R to relational Mapping Algorithm**

**Step 1: Mapping of regular entity types**

For each regular entity type E in the ER schema,

– create a relation R that includes all the simple attributes of E

– include only simple component attributes of composite attribute

– choose one of the key attributes of E as primary key for R

• Ex. Employee, Department, Project relations– primary key

**Step 2: Mapping of weak entity types:**

• For each weak entity type W with owner entity type E:

–create a relation R that includes all the simple attributes of W includes a foreign key attributes of R the primary key attributes of E primary key of R.

**Step 3: Mapping of binary relation(1:1) types:**

For each binary 1:1 relationship type R in the ER schema :identify the relations S and T that correspond to the entity types participating in R–choose the relation S corresponding to entity type with total participation in R– includes foreign key in S the primary key of T– include all the simple attributes of the R as attributes of S

•E.x. MANAGES–DEPARTMENT=>total participation, the role of S–We include primary key S S N of EMPLOYEE as foreign key MGRSSN in the DEPT.–We also include Start Date of the MANAGES in the DEPT. and rename it MGRSTARTDATE.

**Step 4: Mapping of binary (1:N) relation types:**

**Reference**

[1] <https://en.wikipedia.org/wiki/Functional_Database_Model>

[2] <http://www2.amk.fi/>