Chapter Three

Relational Database Modeling

Building Blocks(1)

- **Entities**: Real world physical or logical object.
- **Attributes**: Properties used to describe each Entity or real world object.
- **Relationship**: The association between the real world objects (i.e. Entities.)
- **Constraints**: Rules that should be obeyed or followed while manipulating the data.

Building Blocks(2)

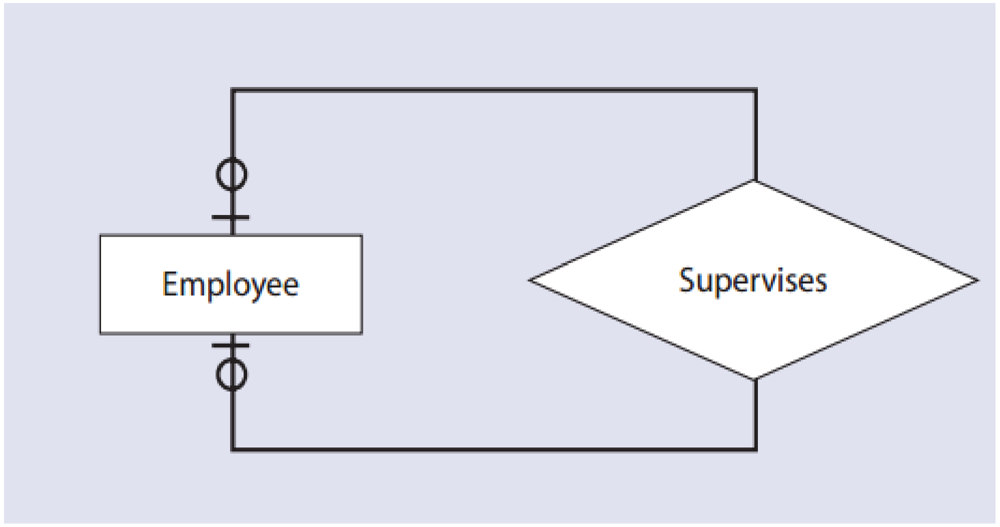
- The name given to an entity should always be a singular noun descriptive of each item to be stored in it.
- **Attributes** - The items of information which characterize and describe these entities
- Attributes are pieces of information about entities
- Attribute name: Should be explanatory words or phrases.
- In any business processing one object may be associated with another object due to some event. Such kind of association is what we call a relationship between entity objects.

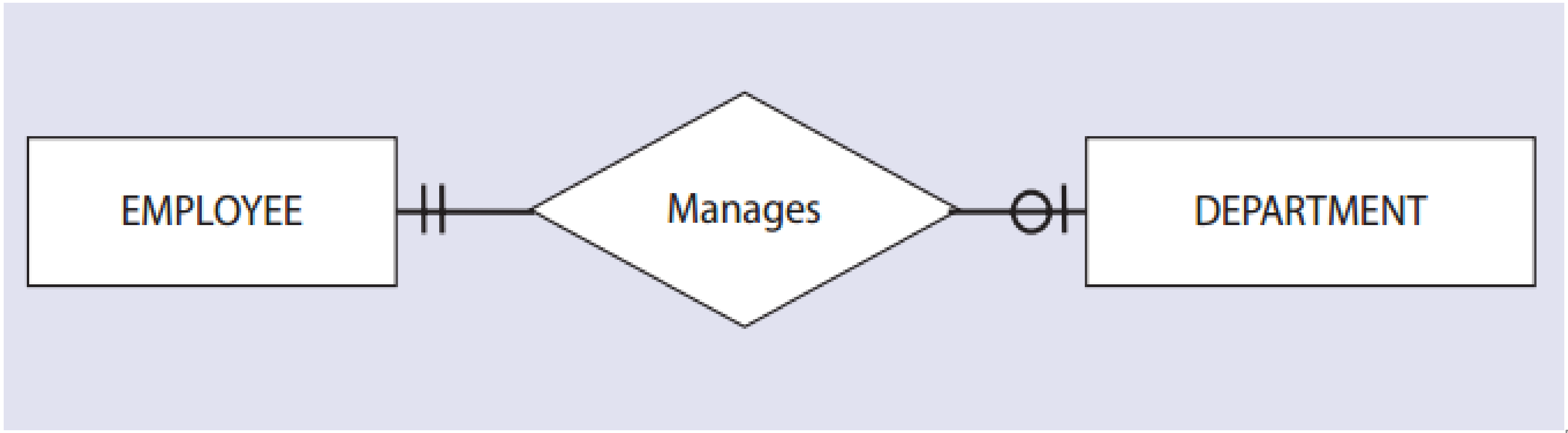
Types of Attributes

- Simple (atomic) Vs Composite attributes
  - Simple : Contains a single value (not divided into sub parts)
  - Composite: Divided into sub parts (composed of other attributes). E.g. Name, Hobbies
- Single-valued Vs multi-valued attributes
  - Single-valued : Have only single value
  - Multi-Valued: Type of attribute that can have more than one value at a time.
  - E.g. Address (email, phone, place of birth…)
- Stored vs. Derived Attributes
- Null Values

Degree of a Relationship

- Unary/recursive relationship
- Binary relationships
- Ternary relationship
- n-nary relationship

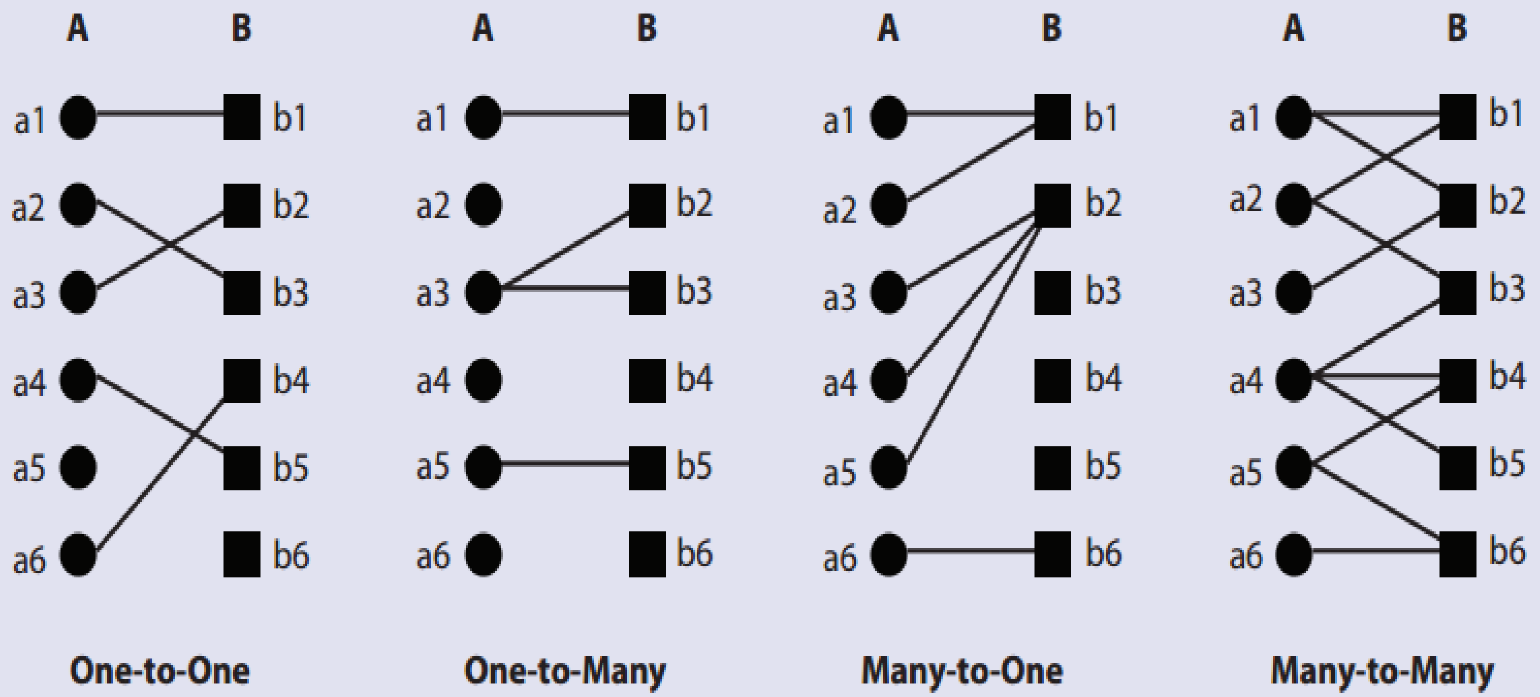




Cardinality of a Relationship (1)

- One-to-One:
- One-to-Many
- Many-to-One
- Many-to-Many

Cardinality of a Relationship (2)



Relational Integrity

- **Domain integrity**: No value of the attribute should be beyond the allowable limits.
- **Entity integrity**: In a base relation, no attribute of a Primary Key can assume a value of NULL.
- **Referential integrity**: If a Foreign Key exists in a relation, either the Foreign Key value must match a Candidate Key value in its home relation or the Foreign Key value must be NULL.
- **Enterprise integrity**: Additional rules specified by the users or database administrators of a database are incorporated.

Types of Keys - Super key

- A set of one or more attributes that in group (collectively) can identify an entity uniquely from the entity set.
- An attribute or set of attributes that uniquely identifies a tuple within a relation.
- Eg. **Employee**(**EmpID**, **EmployeeName**, SSN, DeptID, DOB)
  - {EmpID + EmployeeName}, {EmpID, DOB}, {SSN}, {EmpID}
- Super key stands for superset of a key.
- A Super Key is a set of one or more attributes that are taken collectively and can identify all other attributes uniquely

Types of Keys - Candidate Key (1)

- The candidate key is the sufficient and the necessary set of attributes to distinguish an entity set.
- Are individual columns in a table that qualifies for uniqueness of each row/tuple.
  - ✓Employee(EmpID, EmpName, SSN, DeptIDDOB}= EmpID, SSN
- For a Primary Key and thus are Candidate keys.
- Are super keys for which no proper subset is a super key
- In other words candidate keys are minimal super keys

Types of Keys- Candidate Key (2)

- **Candidate key** - an attribute or set of attributes that uniquely identifies individual occurrences of an entity type or tuple within a relation.
- A candidate key has two properties:
  - ✓ Uniqueness
  - ✓ Irreducibility

Types of Keys – Alternative key

- Candidate column other than the Primary column, like if EmployeeID is set for a PK then SSN would be the Alternate key.
- Employee(EmpID, EmpName, SSN, DeptID, DOB}= SSN

Types of Keys - Composite key

- **Composite key:** A candidate key that consists of two or more attributes.
- If a table do have a single column that qualifies for a Candidate key, then you have to select 2 or more columns to make a row unique.
- Like if there is no EmployeeID or SSN columns, then you can make EmployeeName + DOB as Composite Primary Key.
- But still there can be a narrow chance of duplicate rows
  - ✓ Employee(EmpID, <u>EmpName</u>, SSN, DeptID, <u>DOB</u>}

Types of Keys - Primary key (1)

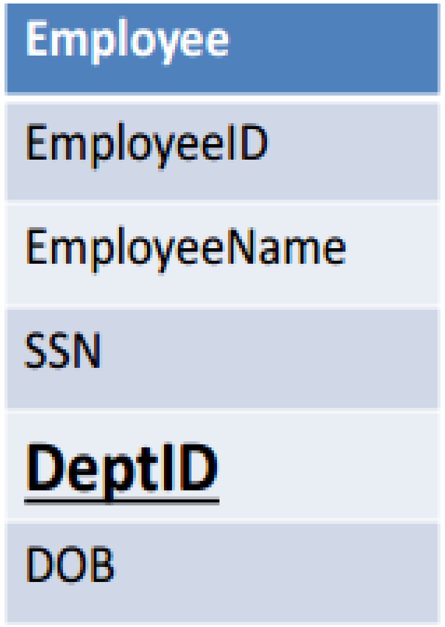
- **Primary key:** the candidate key that is selected to identify tuples uniquely within the relation.
- It is a candidate key that is chosen by the database designer to identify entities with in an entity set.
- Ideally a primary key is composed of only a single attribute.
- But it is possible to have a primary key composed of more than one attribute.
- Is the column you choose to maintain uniqueness in a table at row level

Types of Keys - Primary key (2)

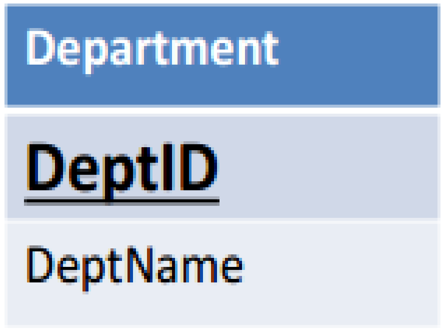
- No two rows can have the same primary key value
- Every row must have a primary key value
- The primary key field cannot be null
- Value in a primary key column can never be modified or updated, if any foreign key refers to that primary key.

Types of Keys – Foreign Key

- **Foreign key:** an attribute, or set of attributes, within one relation that matches the candidate key of some relation.
- A foreign key is a link between different relations to create relationship or view.



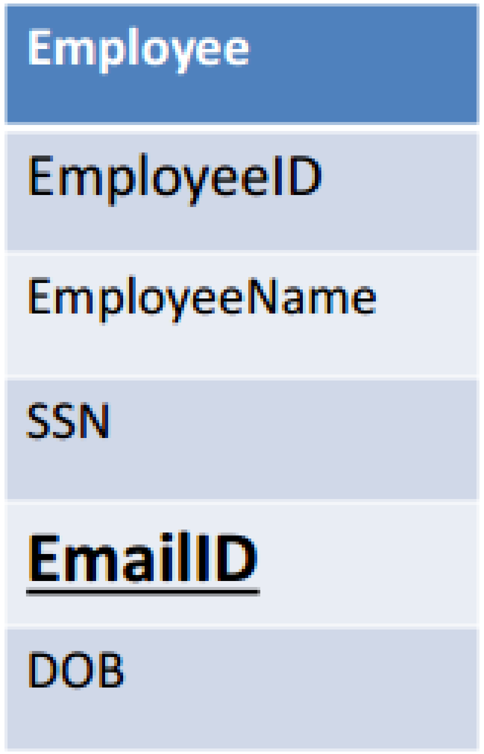
Employee
EmployeeID
EmployeeName
SSN
DeptID
DOB



Department
DeptID
DeptName

Types of Keys – Unique Key

- Unique key is same as primary with the difference being the existence of null.
- Unique key field allows one value as NULL value.



Employee
EmployeeID
EmployeeName
SSN
<u>**EmailID**</u>
DOB

Relational Views (1)

- Relations are perceived as a table from the users’ perspective.
- There are two kinds of relation in relational database.
  - ✓ Base (Named) and
  - ✓ View (Unnamed) Relations.
- The basic difference is on how the relation is created, used and updated:
- **Base Relation**: A named relation corresponding to an entity in the conceptual schema, whose tuples are physically stored in the database.

Relational Views (2)

- **View (Unnamed Relation)**: A View is the dynamic result of one or more relational operations operating on the base relations to produce another virtual relation that does not actually exist as presented.
- So a view is virtually derived relation that does not necessarily exist in the database but can be produced upon request by a particular user at the time of request.

Schema

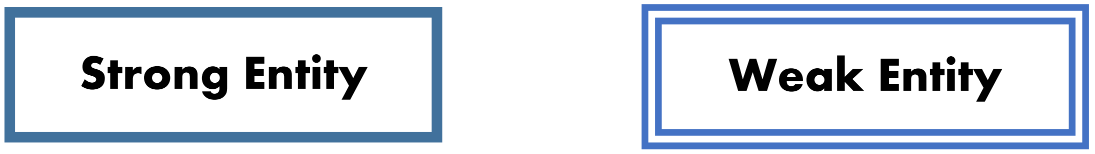
- **Database Schema (Intension):** specifies name of relation and the collection of the attributes (specifically the Name of attributes).
  - ✓Refer to a description of database (or intention)
  - ✓Specified during database design
  - ✓Should not be changed unless during maintenance
- **Schema Diagrams:** convention to display some aspect of a schema visually.
- **Schema Construct:** refers to each object in the schema (e.g. STUDENT) E.g. STUNEDT (FName,LName,Id,Year,Dept,Sex)

Instances

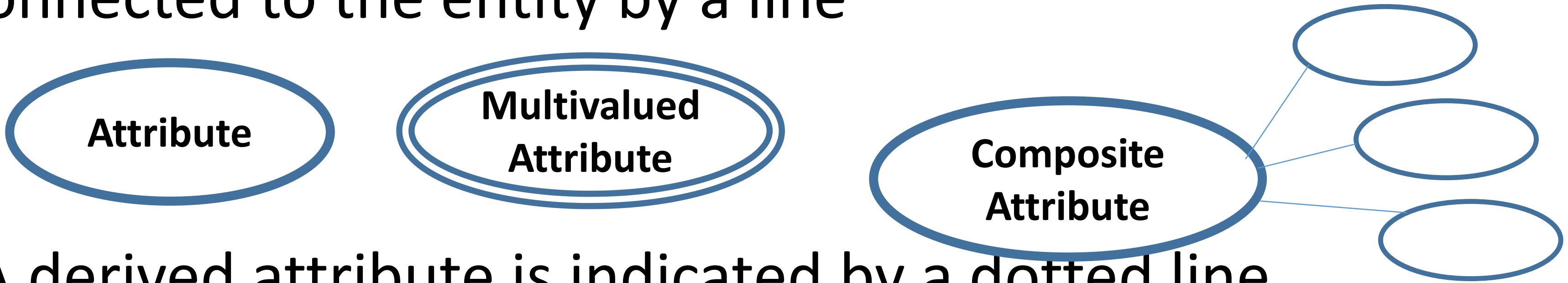
- **Instance:** is the collection of data in the database at a particular point of time (snap-shot).
  - Also called State or Snap Shot or Extension of the database.
  - Refers to the actual data in the database at a specific point in time
- State of database is changed any time we add, delete or update an item.
- **Valid state:** the state that satisfies the structure and constraints specified in the schema and is enforced by DBMS.

Entity Relationship Diagram(1)

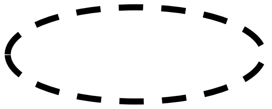
- Entity is represented by a rectangle containing the name of the entity



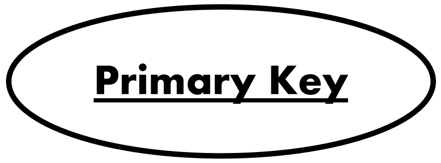
- Attributes are represented by ovals and are connected to the entity by a line



- A derived attribute is indicated by a dotted line

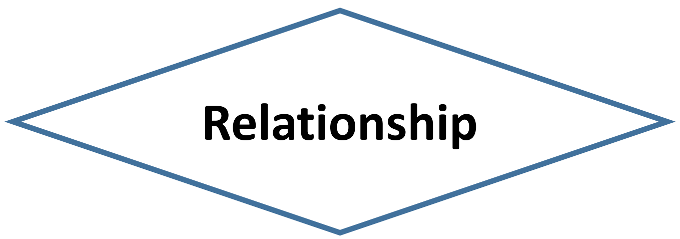


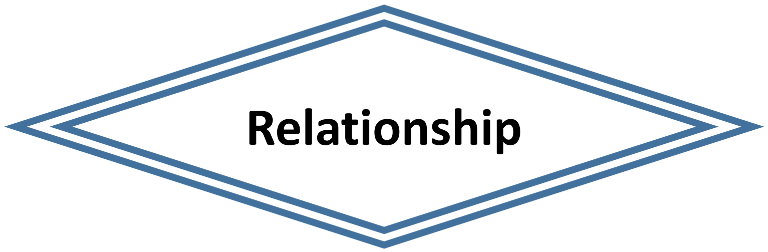
- Primary Keys are underlined



Entity Relationship Diagram(2)

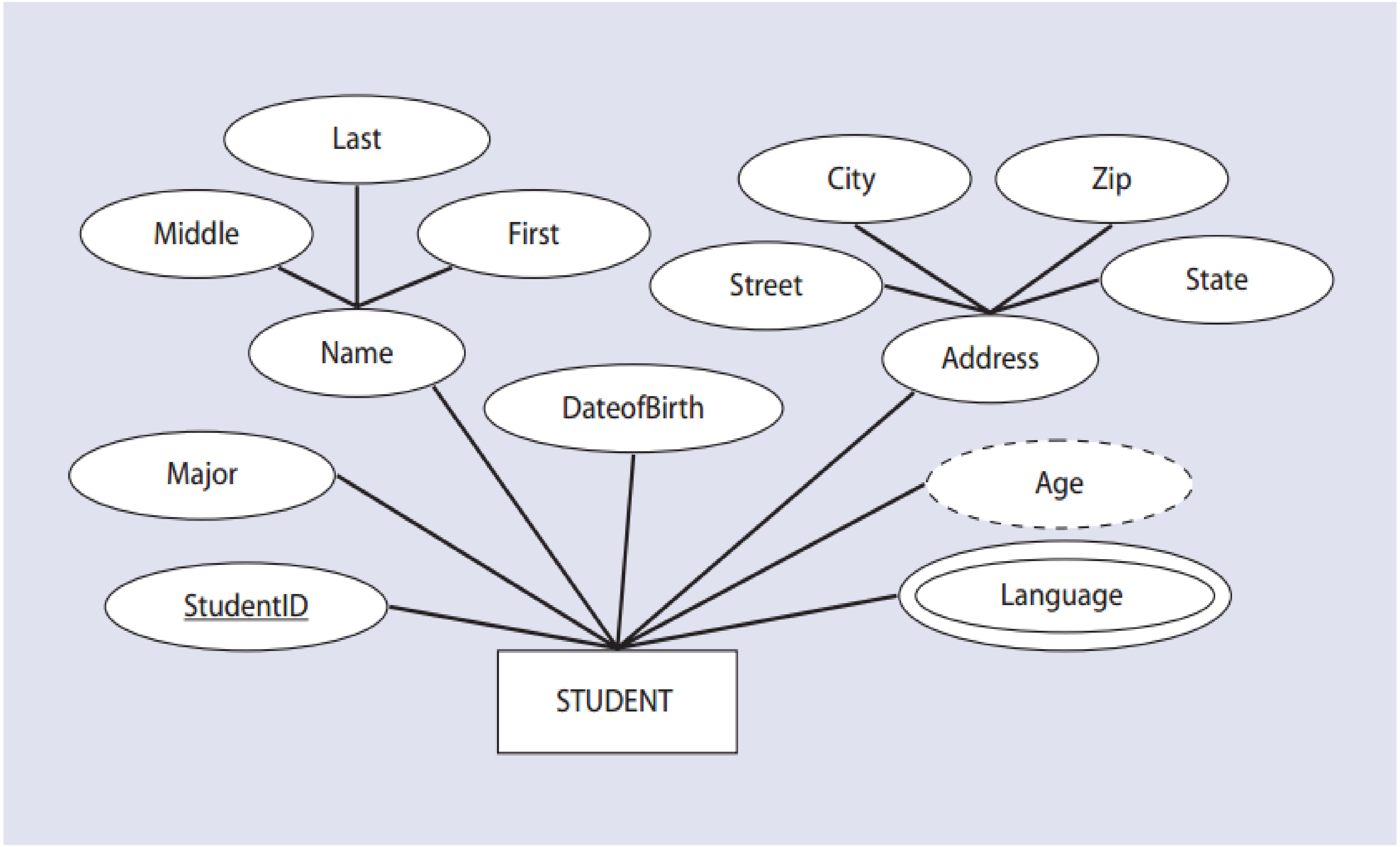
- Relationships are represented by Diamond shaped symbols
  - Weak Relationship is a relationship between Weak and Strong Entities.
  - Strong Relationship is a relationship between two strong Entities.





Entity Relationship Diagram(3)

Attributes of the STUDENT entity type.



ERD – Class Exercise (In Group)

A Personnel record management system will have the following two basic data object categories with their own features or properties: **Employee** will have an Id, Name, DoB, Age, Tel and **Department** will have an Id, Name, Location. Whenever an Employee is assigned in one Department, the duration of his stay in the respective department should be registered.

Draw ERD for the above case study

Thanks !!!