E – R MODEL

1. Entity

- **Definition**: An object or thing in the real world that is distinguishable from other objects.
- **Types**:
  - **Strong Entity**: Has a primary key.
  - **Weak Entity**: Cannot be uniquely identified without a related strong entity.
- **Example**:
  - Student is a **strong entity** (identified by Student_ID).
  - Dependent is a **weak entity** (depends on Employee).

3. Entity Set

- **Definition**: A collection of similar types of entities.
- **Example**:
  - All students in a college form the Student entity set.

2. Attribute

- **Definition**: A property or characteristic of an entity or relationship.
- **Types**:
  - **Simple (Atomic)**: Cannot be divided (e.g., Age).
  - **Composite**: Can be divided into smaller sub-parts (e.g., Name → First Name, Last Name).
  - **Derived**: Can be derived from other attributes (e.g., Age from Date of Birth).
  - **Multivalued**: Can have multiple values (e.g., Phone Numbers).
- **Example**:
  - Student entity may have attributes: Name (Composite), Age (Simple), Phone Numbers (Multivalued).

Relationship

- **Definition**: Association among two or more entities.
- **Types**:
  - **One-to-One (1:1)**: One entity is related to only one other entity.
  - **One-to-Many (1:N)**: One entity is related to many others.
  - **Many-to-Many (M:N)**: Many entities are related to many others.
- **Example**:
  - **1:1** – Each student has one ID card.
  - **1:N** – One teacher teaches many students.
  - **M:N** – Students enrolled in multiple courses; courses have many students.

5. Relationship Set

- **Definition**: A collection of similar relationships.
- **Example**:
  - The "Teaches" relationship between Teachers and Courses.

6. Keys

- **Primary Key**: A unique identifier for an entity.
- **Candidate Key**: All possible unique identifiers.
- **Foreign Key**: Refers to the primary key in another table.
- **Example**:
  - Student_ID is a primary key in the Student entity.

1. Student-Course Enrollment System

Entities:

•Student(Student_ID, Name, Age, Email)

•Course(Course_ID, Title, Credits)

•Instructor(Instructor_ID, Name, Department)

Relationships:

•Enrolls(Student ↔ Course) → with attribute: Enrollment_Date

•Teaches(Instructor ↔ Course)

2. Library Management System

Entities:

•Book(Book_ID, Title, ISBN, Genre)

•Member(Member_ID, Name, Address)

•Librarian(Librarian_ID, Name)

Relationships:

•Borrows(Member ↔ Book) → with attributes: Borrow_Date, Return_Date

•Manages(Librarian ↔ Book)

3. Hospital Management System
Entities:
•Patient(Patient_ID, Name, DOB, Gender)
•Doctor(Doctor_ID, Name, Specialization)
•Appointment(Appointment_ID, Date, Time)
Relationships:
•Visits(Patient ↔ Doctor) via Appointment
•Prescribes(Doctor ↔ Patient) → with attribute: Prescription_Details

4. Banking System
Entities:
•Customer(Customer_ID, Name, Address)
•Account(Account_No, Type, Balance)
•Branch(Branch_ID, Location)
Relationships:
•Owns(Customer ↔ Account)
•Operates(Account ↔ Branch)

5. Online Shopping System
Entities:
•Customer(Customer_ID, Name, Email)
•Product(Product_ID, Name, Price)
•Order(Order_ID, Date, Total_Amount)
Relationships:
•Places(Customer ↔ Order)
•Contains(Order ↔ Product) → with attribute: Quantity

6. Social Media Network
Entities:
•User(User_ID, Name, Email)
•Post(Post_ID, Content, Date_Posted)
•Comment(Comment_ID, Text, Date)
Relationships:
•Writes(User ↔ Post)
•CommentsOn(User ↔ Post) via Comment
•Follows(User ↔ User) → recursive relationship

✅ 7. Airline Reservation System

Entities:

•Passenger(Passenger_ID, Name, Passport_No)
•Flight(Flight_ID, Origin, Destination)
•Ticket(Ticket_No, Seat_No, Price)

Relationships:

•Books(Passenger ↔ Flight) via Ticket

✅ 8. Hotel Booking System

Entities:

•Guest(Guest_ID, Name, Phone)
•Room(Room_ID, Type, Rate)
•Booking(Booking_ID, Checkin, Checkout)

Relationships:

•Books(Guest ↔ Room) via Booking

✅ 9. Employee Payroll System

Entities:

•Employee(Emp_ID, Name, Designation, Salary)
•Department(Dept_ID, Name)
•Payslip(Payslip_ID, Date, Net_Salary)

Relationships:

•BelongsTo(Employee ↔ Department)
•Receives(Employee ↔ Payslip)

✅ 10. School Management System

Entities:

•Student(Student_ID, Name, Grade)
•Teacher(Teacher_ID, Name, Subject)
•Classroom(Classroom_ID, Room_No, Capacity)

Relationships:

•Assigned(Teacher ↔ Classroom)
•Studies(Student ↔ Classroom)