MODULE- WEB DEVELOPMENT AND DATABASES

TERM-1

WEEK- 8

Name: Saina Singh

Student ID: 24071247

Orbund ID: 10975

Programme: BSc.AI (L4)

Section: B

Date: 9th, Jan, 2026

Q. Read the following case study, which describes the data requirements for a DVD rental company. The DVD rental company has several branches throughout the USA. The data held on each branch is the branch address made up of street, city, state, and zip code, and the telephone number. Each branch is given a branch number, which is unique throughout the company. Each branch is allocated staff, which includes a Manager. The Manager is responsible for the day-to-day running of a given branch. The data held on a member of staff is his or her name, position, and salary. Each member of staff is given a staff number, which is unique throughout the company. Each branch has a stock of DVDs. The data held on a DVD is the catalog number, DVD number, title, category, daily rental, cost, status, and the names of the main actors, and the director. The catalog number uniquely identifies each DVD. However, in most cases, there are several copies of each DVD at a branch, and the individual copies are identified using the DVD number. A DVD is given a category such as Action, Adult, Children, Drama, Horror, or Sci-Fi. The status indicates whether a specific copy of a DVD is available for rent. Before hiring a DVD from the company, a customer must first register as a member of a local branch. The data held on a member is the first and last name, address, and the date that the member registered at a branch. Each member is given a member number, which is unique throughout all branches of the company. Once registered, a member is free to rent DVDs, up to maximum of ten at any one time. The data held on each DVD rented is the rental number, the full name and number of the member, the DVD number, title, and daily rental, and the dates the DVD is rented out and date returned. The rental number is unique throughout the company.

(a) Identify the main entity types of the DVD rental company.

(b) Identify the main relationship types between the entity types described in (a) and represent each relationship as an ER diagram.

(c) Determine the multiplicity constraints for each relationship described in (b). Represent the multiplicity for each relationship in the ER diagrams created in (b).

(d) Identify attributes and associate them with entity or relationship types. Represent each attribute in the ER diagrams created in (c).

(e) Determine candidate and primary key attributes for each (strong) entity type.

Using your answers (a) to (e) attempt to represent the data requirements of the DVD rental company as a single ER diagram. State any assumptions necessary to support your design.

(f) Compare the above case study with Sakila DB model shown in your lecture slides?

Q.A. Identify the main entities in the DVD rental system

The main entities in the DVD rental system are:

- **Branch** – represents each rental branch of the company
- **Staff** – represents employees working at branches
- **Member** – represents registered customers
- **DVD_Catalog** – represents DVD titles available for rental
- **DVD_Copy** – represents individual physical copies of DVDs
- **Rental** – represents rental transactions between members and DVDs

These entities are required to store operational, customer, and rental information.

Q.B. Identify the relationships between the entities

The key relationships are:

- A **Branch employs Staff** (one-to-many)
- A **Branch registers Members** (one-to-many)
- A **DVD_Catalog has DVD_Copies** (one-to-many)
- A **Branch holds DVD_Copies** (one-to-many)
- A **Member rents DVD_Copies** via **Rental** (one-to-many)
- A **DVD_Copy can appear in many Rentals over time** (one-to-many)

The **Rental** entity resolves the many-to-many relationship between **Member** and **DVD_Copy**.

Q.C. Determine the multiplicity constraints for each relationship described in (b).

Represent the multiplicity for each relationship in the ER diagrams created in (b).

Multiplicity constraints (min..max) for each relationship

1) Branch — employs — Staff

- **Branch → Staff: 1..*** (a branch employs one or more staff)
- **Staff → Branch: 1..1** (each staff member works at exactly one branch)

ERD: Branch (1) ⸺ (N) Staff

2) Branch — registers — Member

- **Branch → Member: 0..*** (a branch can have zero or many members registered)
- **Member → Branch: 1..1** (each member is registered at exactly one branch)

ERD: Branch (1) ⸺ (N) Member

3) DVD_Catalog — has — DVD_Copy

- **DVD_Catalog → DVD_Copy: 1..*** (a title can have one or many copies)
- **DVD_Copy → DVD_Catalog: 1..1** (each copy is of exactly one title)

ERD: DVD_Catalog (1) ⸺ (N) DVD_Copy

4) Branch — stores/holds — DVD_Copy

- **Branch → DVD_Copy: 0..*** (a branch may hold zero or many copies)
- **DVD_Copy → Branch: 1..1** (each copy belongs to one branch at a time)

ERD: Branch (1) ⸺ (N) DVD_Copy

5) Member — makes — Rental

- **Member → Rental: 0..*** (a member may make zero or many rentals over time)
- **Rental → Member: 1..1** (each rental is made by exactly one member)

ERD: Member (1) —— (N) Rental

6) DVD_Copy — appears in — Rental

- **DVD_Copy → Rental: 0..*** (a copy can be rented many times over time, or never)
- **Rental → DVD_Copy: 1..1** (each rental is for exactly one DVD copy)

ERD: DVD_Copy (1) —— (N) Rental

All of these are **one-to-many** relationships:

- Branch 1—∞ Staff
- Branch 1—∞ Member
- DVD_Catalog 1—∞ DVD_Copy
- Branch 1—∞ DVD_Copy
- Member 1—∞ Rental
- DVD_Copy 1—∞ Rental

Q.D. Identify attributes and associate them with entity or relationship types. Represent each attribute in the ER diagrams created in (c).

Identifying and associating attributes

Attributes were identified for each entity and attached accordingly. Primary keys uniquely identify each entity, while foreign keys represent relationships.

- Branch: branch_no (PK), street, city, state, zip_code, telephone, manager_staff_no (FK)
- Staff: staff_no (PK), first_name, last_name, position, salary, branch_no (FK)
- Member: member_no (PK), first_name, last_name, address, date_registered, branch_no (FK)
- DVD_Catalog: catalog_no (PK), title, category, daily_rental, cost, director, main_actors
- DVD_Copy: dvd_no (PK), status, catalog_no (FK), branch_no (FK)

- Rental: rental_no (PK), date_out, date_returned, member_no (FK), dvd_no (FK)

Relationship-specific attributes such as rental dates are stored in the Rental entity. All attributes are represented in the ER diagram using standard ER notation.

Q.E. Determine candidate and primary key attributes for each (strong) entity type.

Using your answers (a) to (e) attempt to represent the data requirements of the DVD rental company as a single ER diagram. State any assumptions necessary to support your design.

Candidate keys + primary keys (strong entities)

- **Branch**
  - **Candidate key(s): branch_no** *(telephone could be candidate if guaranteed unique)*
  - **Primary key: branch_no**
- **Staff**
  - **Candidate key(s): staff_no**
  - **Primary key: staff_no**
- **Member**
  - **Candidate key(s): member_no** *(email/phone could be candidate if stored + unique)*
  - **Primary key: member_no**
- **DVD_Catalog (Title)**
  - **Candidate key(s): catalog_no**
  - **Primary key: catalog_no**
- **DVD_Copy (Physical disc)**
  - **Candidate key(s): dvd_no**
  - **Primary key: dvd_no**
- **Rental (Transaction)** *(often treated as an entity because it has attributes like dates)*

- o **Candidate key(s): rental_no** *(or composite: dvd_no + date_out, but rental_no is simpler)*

- o **Primary key: rental_no**

Single ER diagram (what to draw)

Entities: Branch, Staff, Member, DVD_Catalog, DVD_Copy, Rental

Relationships + multiplicity (crow's foot):

- Branch 1 —— N Staff *(Staff works at exactly 1 branch)*

- Branch 1 —— N Member *(Member registered at exactly 1 branch)*

- DVD_Catalog 1 —— N DVD_Copy *(each copy is one title)*

- Branch 1 —— N DVD_Copy *(each copy held at one branch)*

- Member 1 —— N Rental *(each rental belongs to one member)*

- DVD_Copy 1 —— N Rental *(each rental is for one copy)*

  - o Rental has attributes: date_out, date_returned

Manager: Branch has one manager (a Staff member)

- Add attribute/relationship: Branch.manager_staff_no → Staff.staff_no

Assumptions

1. Each staff member works at one branch only.

2. Each member registers at one branch only.

3. Each DVD copy belongs to one branch at a time.

4. A DVD copy can be rented many times over time, but only one active rental at a time (handled by business rules).

5. The "max 10 DVDs at a time" rule is a business constraint enforced by application/queries, not by ER structure.