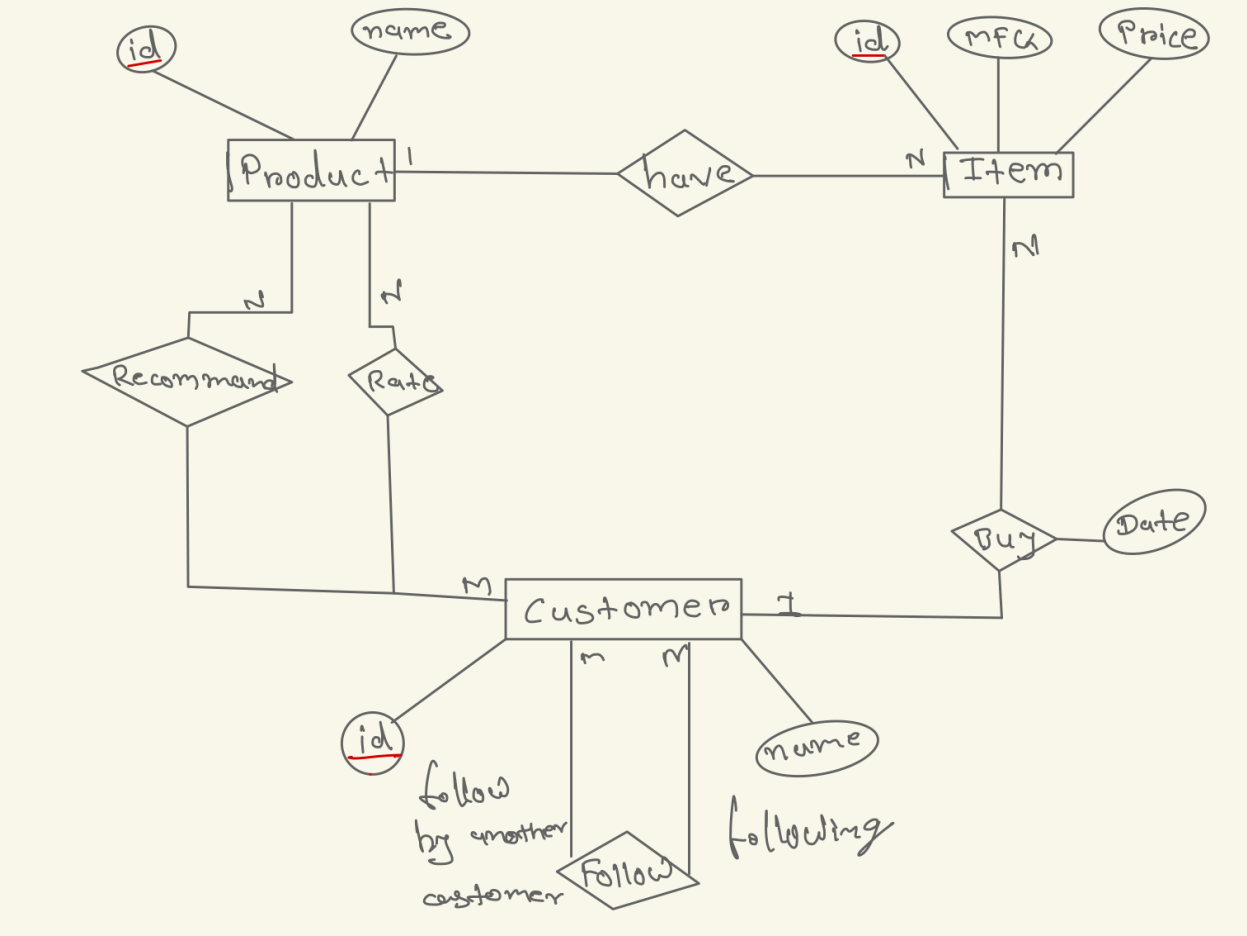
Question Part (50 points)

Question 1 (15 points)

We plan to construct a database for an online store.

- Each product has id and name.
- A product consists of multiple items.
- Each item has id, manufacturing date (i.e., MFG), and price.
- A customer has id and name.
- A customer is able to buy multiple items, and the date of each purchase is recorded.
- A customer can recommend a product and submit a rating for it.
- A customer can follow another customer, and we record these followings.

Draw an ER diagram for this application. Specify key attributes of each entity type and structural constraints on each relationship type. Note any unspecified requirements, and make appropriate assumptions to make the specification complete.



Question 2 (15 points)

Write DDL statements to create the tables for the ER diagram of Question1.

Table for Product

```
CREATE TABLE Product (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR(100)
);
```

Table for Item

```
CREATE TABLE Item (
    ItemID INT PRIMARY KEY,
    MFG DATE,
    Price DECIMAL(10, 2),
    ProductID INT,
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID)
);
```

Table for Customer

```
CREATE TABLE Customer (
    CustomerID INT PRIMARY KEY,
    CustomerName VARCHAR(100)
);
```

Table for Purchase (Relationship between Customer and Item)

```
CREATE TABLE Purchase (
    CustomerID INT,
    ItemID INT,
    PurchaseDate DATE NOT NULL,
    PRIMARY KEY (CustomerID, ItemID, PurchaseDate),
```

```
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID),
    FOREIGN KEY (ItemID) REFERENCES Item(ItemID)
);
```

Table for Recommendation (Relationship between Customer and Product)

```
CREATE TABLE Recommendation (
    CustomerID INT,
    ProductID INT,
    Rating INT CHECK (Rating BETWEEN 1 AND 5),
    PRIMARY KEY (CustomerID, ProductID),
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID),
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID)
);
```

Table for Follow (Self-relationship for following customers)

```
CREATE TABLE Follow (
    FollowerID INT,
    FollowingID INT,
    PRIMARY KEY (FollowerID, FollowingID),
    FOREIGN KEY (FollowerID) REFERENCES Customer(CustomerID),
    FOREIGN KEY (FollowingID) REFERENCES Customer(CustomerID)
);
```

Question 3 (20 points). Given a relation R (A, B, C, D, E). Answer the following questions with the given set of FDs: (1) identify candidate keys and (2) state the strongest normal form that R satisfies (e.g., 1NF, 2NF, 3NF, or BCNF)

(a) FDs: AB → CDE, CD → AB

AB → CDE
CD → AB

AB → CDE ∴ AB & CD is
CD → AB → CDE the superkey

↳ candidate keys : AB & CD

↳ strongest Normal form : BCNF

(b) FDs: A → B, B → C, C → D, D → E, E → A

Fds : A → B , B → C , C → D , D → E , E → A

A → B → C → D → E

↳ candidate keys : A, B, C, D, and E

A → B → C → D → E ∴ A is the SK
B → C → D → E → A ∴ B is the SK
C → D → E → A → B ∴ C is the SK
D → E → A → B → C ∴ D is the SK
E → A → B → C → D ∴ E is the SK

↳ Strongest Normal form : BCNF

(c) FDs: ABC → D, DE → A

Fds : ABC → D, DE → A

ABCE → DE → A ∴ ABCE is SK
BCDE → ABC → D ∴ BCD is SK

Candidates key: ABCE, BCD

Strongest Normal form : 3NF

(d) FDs: A → BE, A → C, BC → D

Fds : AB → C, BC → D, CD → A

A → BCE → DE ∴ A is super key

ABC → AD → BED ∴ ABC is super key

candidate key: A, ABC

Strongest Normal form : 2NF

(e) FDs: AB → C, BC → D, CD → A

FDs : AB -> C , BC -> D , CD -> A

ABE -> BCE -> DE ∴ ABE is the super key

BCE -> CDE -> AE ∴ BCE is the super key

BCDE -> ABE ∴ BCDE is the Super Key

candidate key: ABE, BCE, and BCDE

Strongest Normal form: 3NF