

### Relational Schema:

1. User(userId INT [PK], userName VARCHAR(200), account VARCHAR(50), password VARCHAR(50), email VARCHAR(50), phoneNumber INT)
2. Post(postId INT [PK], userId INT, storeName VARCHAR(50), productName VARCHAR(50), price REAL, link VARCHAR(200), categoryId INT, expirationDate Date, groupLimit INT, paymentMethod VARCHAR(20))
3. Category(categoryId INT [PK], categoryName VARCHAR(200))
4. Group(groupId INT [PK], userId INT [FK to User.userId], postId INT [FK to Post.postId])
5. Leader(leaderId INT [PK], postId INT [FK to Post.postId], userId INT [FK to Post.postId])
6. Payment(paymentId INT [PK], paymentMethod VARCHAR(20), postId INT [FK to Post.postId], leaderId INT, groupId INT [FK to Group.groupId])

### Assumption:

1. **Post:** This entity has group shopping information of a new post, which includes the attributes postId, userId, storeName, productName, price, link, categoryId, expirationDate, groupLimit.
2. **User:** This entity records information of a user, which includes userId, userName, account, password, email, phoneNumber
3. **Group:** This entity has the information of the group within the post, which includes the attributes groupId, userId, and postId. Group is a weak entity and depends on Post and User.
4. **Leader:** This entity records the information of the group buying leader, which includes the attributes leaderId, postId, userId. Leader is a weak entity and depends on Post.
5. **Payment:** This entity records the payment details of a post, which includes the attributes paymentId, paymentMethod, postId, leaderId, userId, groupId. Payment is a weak entity and depends on Group.
6. **Category:** This entity has the category of a group shopping product, which includes categoryId and categoryName.

### Description of each relationship and its cardinality:

1. **Publish:** The relationship between User and Post. It follows a cardinality of below: Each User must be related to (“Publish”) by zero to many Posts.  
Each Post must be related to (“Publish”) exactly one User.
2. **Include:** The relationship between Group and Post. It follows a cardinality of below: Each Group must be related to (“Include”) exactly one Post.  
Each Post must be related to (“Include”) exactly one Group.
3. **Create:** The relationship between Payment and Group. It follows a cardinality of below: Each Payment must be related to (“Create”) exactly one Group.  
Each Group must be related to (“Create”) exactly one Payment.
4. **Receive:** The relationship between Payment and Leader. It follows a cardinality of below:  
Each Payment must be related to (“Receive”) exactly one Leader.  
Each Leader must be related to (“Receive”) by zero to many Payments.
5. **Contain:** The relationship between Post and Category. It follows a cardinality of below: Each Post must be related to (“Contain”) exactly one Category.  
Each Category must be related to (“Contain”) exactly one Post.
6. **Have:** The relationship between Post and Leader. It follows a cardinality of below: Each Post must be related to (“Have”) exactly one Leader.  
Each Leader must be related to (“Have”) exactly one Post.
7. **Join:** The relationship between User and Group. It follows a cardinality of below: Each User must be related to (“Join”) by zero to many Groups.  
Each Group must be related to (“Join”) one to many Users.