ASSIGNMENT:1

```
Name : Online Shopping Cart

Description:

Design a simple online shopping cart system using object-oriented programming concepts. Create the following classes:

Product:

Attributes: name, price, quantity
Methods:
Constructor to initialize the attributes
Getter and setter methods for each attribute
ShoppingCart:

Attributes: products (a list of Product objects)
Methods:
Constructor to initialize the products list
Method to add a product to the cart
Method to remove a product from the cart
Method to calculate the total price of all products in the cart
Method to display the contents of the cart
Main:

Create an instance of the ShoppingCart class
Add a few products to the cart
Display the contents of the cart
Calculate and display the total price of the products in the cart
```

In [16]:

```python
class Product:
    def __init__(self, name, price, quantity):
        self.name = name
        self.price = price
        self.quantity = quantity

    def get_name(self):
        return self.name

    def set_name(self, name):
        self.name = name

    def get_price(self):
        return self.price

    def set_price(self, price):
        self.price = price

    def get_quantity(self):
        return self.quantity

    def set_quantity(self, quantity):
        self.quantity = quantity


class ShoppingCart:
    def __init__(self):
        self.products = []

    def add_product(self, product):
        self.products.append(product)

    def remove_product(self, product):
        self.products.remove(product)
```

```
    def calculate_total_price(self):
        total_price = 0
        for product in self.products:
            total_price += product.get_price() * product.get_quantity()
        return total_price

    def display_cart(self):
        for product in self.products:
            print(f"Product: {product.get_name()}, Price: {product.get_price()}, Quantity: {product.get_quantity()}")

shopping_cart = ShoppingCart()

# Adding products to the cart
product1 = Product("Item 1", 30, 7)
product2 = Product("Item 2", 38, 5)

shopping_cart.add_product(product1)
shopping_cart.add_product(product2)

# Displaying the contents of the cart
shopping_cart.display_cart()

# Calculating and displaying the total price
total_price = shopping_cart.calculate_total_price()
print("Total Price:", total_price)
```

```
Product: Item 1, Price: 30, Quantity: 7
Product: Item 2, Price: 38, Quantity: 5
Total Price: 400
```

ASSIGNMENT:2

```
Name : Social Media Platform

Description:

Design and develop a social media platform using object-oriented programming concepts.

Create a class called "User" with the following attributes and methods:

Attributes: userId (integer), name (string), email (string), friends (list of User objects)
Methods: getUserId(), getName(), getEmail(), addFriend(user), removeFriend(user), viewFriends()
Create a class called "Post" with the following attributes and methods:

Attributes: postId (integer), user (User object), content (string), likes (integer)
Methods: getPostId(), getUser(), getContent(), getLikes(), like(), unlike()
Create a class called "SocialMediaPlatform" with the following attributes and methods:

Attributes: users (list of User objects), posts (list of Post objects)
Methods: addUser(user), removeUser(user), createPost(user, content), removePost(post), displayPosts()
```

In [17]:

```
class User:
    def __init__(self, userId, name, email):
        self.userId = userId
        self.name = name
        self.email = email
        self.friends = []

    def getUserId(self):
        return self.userId

    def getName(self):
        return self.name

    def getEmail(self):
        return self.email

    def addFriend(self, user):
        self.friends.append(user)

    def removeFriend(self, user):
        self.friends.remove(user)

    def viewFriends(self):
        for friend in self.friends:
            print(f"Friend: {friend.getName()}, Email: {friend.getEmail()}")


class Post:
    def __init__(self, postId, user, content):
        self.postId = postId
        self.user = user
        self.content = content
        self.likes = 0
```

```
    def getPostId(self):
        return self.postId

    def getUser(self):
        return self.user

    def getContent(self):
        return self.content

    def getLikes(self):
        return self.likes

    def like(self):
        self.likes += 1

    def unlike(self):
        if self.likes > 0:
            self.likes -= 1

class SocialMediaPlatform:
    def __init__(self):
        self.users = []
        self.posts = []

    def addUser(self, user):
        self.users.append(user)

    def removeUser(self, user):
        self.users.remove(user)

    def createPost(self, user, content):
        postId = len(self.posts) + 1
        post = Post(postId, user, content)
        self.posts.append(post)
```

```
    def removePost(self, post):
        self.posts.remove(post)

    def displayPosts(self):
        for post in self.posts:
            print(f"Post ID: {post.getPostId()}")
            print(f"User: {post.getUser().getName()}, Email: {post.getUser().getEmail()}")
            print(f"Content: {post.getContent()}")
            print(f"Likes: {post.getLikes()}\n")
social_media = SocialMediaPlatform()

# Create users
user1 = User(1, "Nandhu", "Nandhu@example.com")
user2 = User(2, "Gowthu", "Gowthu@example.com")
user3 = User(3, "poojitha(ammu)", "pooji@example.com")

# Add users to the social media platform
social_media.addUser(user1)
social_media.addUser(user2)
social_media.addUser(user3)

# Create posts
social_media.createPost(user1, "Hello")
social_media.createPost(user2, "How are you!")
social_media.createPost(user3, "happy days from today on words!")

# Add friends
user1.addFriend(user2)
user1.addFriend(user3)
user2.addFriend(user3)

# Display posts
social_media.displayPosts()
```

```python
# Like posts
post1 = social_media.posts[0]
post1.like()
post1.like()

post2 = social_media.posts[1]
post2.like()

# Display posts with updated likes
social_media.displayPosts()

# Remove user and post
social_media.removeUser(user2)
social_media.removePost(post1)

# Display posts and friends after removal
social_media.displayPosts()
user1.viewFriends()
```

```
Post ID: 1
User: Nandhu, Email: Nandhu@example.com
Content: Hello
Likes: 0

Post ID: 2
User: Gowthu, Email: Gowthu@example.com
Content: How are you!
Likes: 0

Post ID: 3
User: poojitha(ammu), Email: pooji@example.com
Content: happy days from today on words!
Likes: 0
```

```
Post ID: 3
User: poojitha(ammu), Email: pooji@example.com
Content: happy days from today on words!
Likes: 0

Post ID: 2
User: Gowthu, Email: Gowthu@example.com
Content: How are you!
Likes: 1

Post ID: 3
User: poojitha(ammu), Email: pooji@example.com
Content: happy days from today on words!
Likes: 0

Friend: Gowthu, Email: Gowthu@example.com
Friend: poojitha(ammu), Email: pooji@example.com
```