In [ ]:

```
class Smartphone:
    # Define Smartphone class with name and options attributes
    def __init__(self, name, options):
        # Initialize attributes
        self.name = name  # Smartphone name
        self.options = options  # Available options for the smartphone

class Option:
    # Define Option class with capacity and price attributes
    def __init__(self, capacity, price):
        # Initialize attributes
        self.capacity = capacity  # Storage capacity of the option
        self.price = price  # Price of the option

# Define available smartphones with their respective options
iphone_12 = Smartphone("iPhone 12", (Option("128GB", 3199), Option("256GB", 3599)))
iphone_13_mini = Smartphone("iPhone 13 Mini", (Option("64GB", 2499), Option("128GB", 2
iphone_13_pro = Smartphone("iPhone 13 Pro", (Option("256GB", 4999), Option("512GB", 59
samsung_galaxy_s1 = Smartphone("Samsung Galaxy S1", (Option("128GB", 2899), Option("25
samsung_galaxy_s2 = Smartphone("Samsung Galaxy S2", (Option("64GB", 1799), Option("128

# Define functions for displaying smartphones and their options
def display_smartphones(smartphones):
    # Display available smartphones
    print("Available Smartphones:")
    for i, phone in enumerate(smartphones, 1):
        print(f"{i}. {phone.name}")  # Display smartphone name with index

def display_options(phone):
    # Display available options for a given smartphone
    print(f"Options for {phone.name}:")
    count = 1
    for option in phone.options:
        # Display each option with its capacity and price
        print(f"   {count}. {option.capacity} - {option.price} AED")
        count += 1

# Define user input functions
def get_choice(prompt, options):
    # Get user choice within the specified range
    while True:
        try:
            choice = my_input(prompt)  # Get user input
            if choice.lower() == 'quit':
                return 'quit'  # Exit loop if user wants to quit
            choice = int(choice)  # Convert input to integer
            if choice < 1 or choice > options:  # Check if choice is within range
                raise ValueError  # Raise exception for invalid input
            return choice  # Return valid choice
        except ValueError:
            print("Invalid input. Please enter a valid option number or 'quit'.")

def my_input(prompt):
    # Custom input function to display prompt and get user input
    print(prompt, end='')  # Display prompt without newline
    return input()  # Get user input and return

def calculate_total_price(choices, smartphones):
```

```
    # Calculate total price based on user choices
    total_price = 0
    for choice in choices:
        phone = smartphones[choice[0] - 1]  # Get selected smartphone
        option = phone.options[choice[1] - 1]  # Get selected option for the smartphor
        total_price += option.price * choice[2]  # Multiply option price by quantity o
    return total_price  # Return total price

def apply_discount(total_price, payment_method):
    # Apply discount based on payment method
    discount_percentage = 0
    if payment_method.lower() == 'fab':
        discount_percentage = 10
    elif payment_method.lower() == 'adib':
        discount_percentage = 15
    elif payment_method.lower() == 'other':
        discount_percentage = 5

    discount = total_price * (discount_percentage / 100)  # Calculate discount amount
    discounted_price = total_price - discount  # Subtract discount from total price
    return discounted_price  # Return discounted price

def main():
    # Define available smartphones
    smartphones = [iphone_12, iphone_13_mini, iphone_13_pro, samsung_galaxy_s1, samsur
    choices = []  # Initialize list to store user choices
    while True:
        display_smartphones(smartphones)  # Display available smartphones
        model_choice = get_choice("Enter the number of the smartphone model you want t
        if model_choice == 'quit':
            break  # Exit loop if user wants to quit
        phone = smartphones[model_choice - 1]  # Get selected smartphone
        display_options(phone)  # Display available options for selected smartphone
        option_choice = get_choice("Enter the number of the option you want: ", len(ph
        quantity = int(my_input("Enter the quantity: "))  # Get user input for quantit
        while quantity <= 0:
            print("Quantity cannot be less than or equal to 0.")  # Display error mess
            quantity = int(my_input("Enter the quantity: "))  # Get corrected quantity
        choices.append((model_choice, option_choice, quantity))  # Add user choice to

    total_price = calculate_total_price(choices, smartphones)  # Calculate total price

    print("\nPayment Methods:")
    print("1. FAB")
    print("2. ADIB")
    print("3. Other")
    print("4. Cash")
    payment_method = my_input("Enter the payment method: ")  # Get user choice for pay

    if payment_method.lower() in ['fab', 'adib', 'other']:
        total_price = apply_discount(total_price, payment_method)  # Apply discount ba

    print(f"\nTotal price for your purchase: {total_price} AED")  # Display total pric

    # Calculate membership points earned
    membership_points = total_price // 100
    print(f"\nMembership points earned: {membership_points}")  # Display membership po

    if membership_points >= 10:
        redeem_choice = my_input("Do you want to redeem your points for a discount? (y
```

```
        if redeem_choice.lower() == 'yes':
            discount = membership_points * 5  # Calculate discount amount based on poi
            total_price -= discount  # Subtract discount from total price
            print(f"Discount applied: {discount} AED")  # Display applied discount
            print(f"Updated total price after discount: {total_price} AED")  # Display

if __name__ == "__main__":
    main()
```

```
Available Smartphones:
1. iPhone 12
2. iPhone 13 Mini
3. iPhone 13 Pro
4. Samsung Galaxy S1
5. Samsung Galaxy S2
Enter the number of the smartphone model you want to buy (or type 'quit' to finish):
Options for Samsung Galaxy S1:
   1. 128GB - 2899 AED
   2. 256GB - 3299 AED
Enter the number of the option you want: Enter the quantity: Available Smartphones:
1. iPhone 12
2. iPhone 13 Mini
3. iPhone 13 Pro
4. Samsung Galaxy S1
5. Samsung Galaxy S2
Enter the number of the smartphone model you want to buy (or type 'quit' to finish):
Options for Samsung Galaxy S2:
   1. 64GB - 1799 AED
   2. 128GB - 1999 AED
Enter the number of the option you want: Enter the quantity: Available Smartphones:
1. iPhone 12
2. iPhone 13 Mini
3. iPhone 13 Pro
4. Samsung Galaxy S1
5. Samsung Galaxy S2
Enter the number of the smartphone model you want to buy (or type 'quit' to finish):
Options for iPhone 13 Pro:
   1. 256GB - 4999 AED
   2. 512GB - 5999 AED
Enter the number of the option you want: Enter the quantity: Available Smartphones:
1. iPhone 12
2. iPhone 13 Mini
3. iPhone 13 Pro
4. Samsung Galaxy S1
5. Samsung Galaxy S2
Enter the number of the smartphone model you want to buy (or type 'quit' to finish):
Payment Methods:
1. FAB
2. ADIB
3. Other
4. Cash
Enter the payment method:
Total price for your purchase: 66882 AED

Membership points earned: 668
Do you want to redeem your points for a discount? (yes/no):
```

In [ ]: