Below is a Python program implementing a **Bike Rental System** using Object-Oriented Programming (OOP) concepts. The system allows the user to view the total stock of bikes, request bikes for rent, and exit the program.

### Python Code:

```python
class BikeRental:
    def __init__(self, stock):
        # Initialize the total stock of bikes
        self.stock = stock

    def display_stock(self):
        # Display the total number of bikes available for rent
        print(f"Total bikes available: {self.stock}")
        return self.stock

    def rent_bike(self, quantity):
        # Rent bikes if available in the required quantity
        if quantity <= 0:
            print("Invalid input. Quantity should be positive.")
        elif quantity > self.stock:
            print(f"Sorry, we only have {self.stock} bikes available for rent.")
        else:
            self.stock -= quantity
            print(f"You have rented {quantity} bike(s).")
            print(f"Bikes remaining in stock: {self.stock}")

    def exit_program(self):
        # Exit the program
        print("Thank you for using the Bike Rental System. Goodbye!")
```

```
def main():
    # Create a BikeRental object with an initial stock of 10 bikes
    bike_rental = BikeRental(10)

    while True:
        # Display menu options
        print("\n===== Bike Rental System =====")
        print("1. Display available bikes")
        print("2. Rent a bike")
        print("3. Exit")

        # Get user input for their choice
        choice = input("Enter your choice (1/2/3): ")

        if choice == '1':
            # Display the available bikes
            bike_rental.display_stock()

        elif choice == '2':
            # Rent a bike, ask the user for the number of bikes they want to rent
            try:
                quantity = int(input("How many bikes would you like to rent? "))
                bike_rental.rent_bike(quantity)
            except ValueError:
                print("Invalid input. Please enter a number.")

        elif choice == '3':
            # Exit the program
            bike_rental.exit_program()
            break
        else:
            # Handle invalid input
            print("Invalid choice. Please select a valid option.")

# Run the program
if __name__ == "__main__":
    main()
```

### Explanation:

1. **Class `BikeRental`**:
   - **`__init__(self, stock)`**: Initializes the bike stock when an instance of the class is created.
   - **`display_stock(self)`**: Displays the total number of bikes available for rent.
   - **`rent_bike(self, quantity)`**: Handles the process of renting bikes. It checks if the requested number of bikes is valid and updates the stock.
   - **`exit_program(self)`**: Ends the program with a message.

2. **Function `main()`**:
   - Displays a menu for the user to interact with the system.
   - Users can choose to:
     1. View available bikes.
     2. Rent bikes by specifying the quantity.
     3. Exit the program.

### Example of a Possible Interaction:

```
===== Bike Rental System =====
1. Display available bikes
```

```
2. Rent a bike

3. Exit

Enter your choice (1/2/3): 1

Total bikes available: 10


===== Bike Rental System =====

1. Display available bikes

2. Rent a bike

3. Exit

Enter your choice (1/2/3): 2

How many bikes would you like to rent? 3

You have rented 3 bike(s).

Bikes remaining in stock: 7


===== Bike Rental System =====

1. Display available bikes

2. Rent a bike

3. Exit

Enter your choice (1/2/3): 3

Thank you for using the Bike Rental System. Goodbye!
```

This program allows the user to manage a bike rental system easily and uses OOP concepts like classes and methods for better structure.