```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace lab_2
{
    using System;

    class Book
    {
        public string Title { get; set; }
        public string Author { get; set; }
    }

    class Program
    {
        static void Main(string[] args)
        {
            // Instantiate an object of the Book class
            Book myBook = new Book();

            // Set values for the properties
            myBook.Title = "Madol Duwa";
            myBook.Author = "Martin Wikramasingha";

            // Display the information on the console
            Console.WriteLine("Book Title: " + myBook.Title);
```

```
        Console.WriteLine("Author: " + myBook.Author);

        Console.ReadLine(); // Keep the console window open
    }
  }

}
```

02)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace q2
{
        using System;

class BankAccount
    {
        public string AccountNumber { get; set; }
        public double Balance { get; private set; }

        public BankAccount(string accountNumber, double initialBalance)
        {
            AccountNumber = accountNumber;
            Balance = initialBalance;
        }
```

```
    public void Deposit(double amount)
    {
        if (amount <= 0)
        {
            Console.WriteLine("Deposit amount must be greater than zero.");
        }
        else
        {
            Balance += amount;
            Console.WriteLine("Deposit of $" + amount + " successful.");
        }
    }
}

class Program
{
    static void Main(string[] args)
    {
        // Instantiate a BankAccount object
        BankAccount myAccount = new BankAccount("123456789", 100.0);

        // Display initial balance
        Console.WriteLine("Initial Balance: $" + myAccount.Balance);

        // Deposit money into the account
        double depositAmount = 50.0;
        myAccount.Deposit(depositAmount);

        // Display updated balance
```

```
            Console.WriteLine("Updated Balance: $" + myAccount.Balance);

            Console.ReadLine(); // Keep the console window open
        }
    }


}
```

03)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace q3
{
    using System;

    class TemperatureTracker
    {
        private double[] temperatures;

        public TemperatureTracker(int days)
        {
            temperatures = new double[days];
        }

        public void SetTemperature(int day, double temperature)
```

```
    {
        if (day >= 0 && day < temperatures.Length)
        {
            temperatures[day] = temperature;
        }
        else
        {
            Console.WriteLine("Invalid day. Day should be between 0 and " + (temperatures.Length - 1));
        }
    }

    public void DisplayWeeklyReport()
    {
        Console.WriteLine("Weekly Temperature Report:");
        for (int i = 0; i < temperatures.Length; i++)
        {
            Console.WriteLine("Day " + i + ": " + temperatures[i] + "°C");
        }
    }
}

class Program
{
    static void Main(string[] args)
    {
        int daysInWeek = 7;
        TemperatureTracker tracker = new TemperatureTracker(daysInWeek);

        // Input temperatures for each day
```

```
        for (int i = 0; i < daysInWeek; i++)
        {
            Console.Write("Enter temperature for day " + i + ": ");
            double temperature = Convert.ToDouble(Console.ReadLine());
            tracker.SetTemperature(i, temperature);
        }

        // Display the weekly temperature report
        tracker.DisplayWeeklyReport();

        Console.ReadLine(); // Keep the console window open
    }
  }

}
```

04)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace q4
{
  using System;

  class Product
  {
    public string ProductName { get; set; }
```

```
    public double Price { get; set; }

    // Parameterized constructor
    public Product(string productName, double price)
    {
        ProductName = productName;
        Price = price;
    }
}

class Program
{
    static void Main(string[] args)
    {
        // Instantiate objects using the constructor
        Product product1 = new Product("Laptop", 999.99);
        Product product2 = new Product("Smartphone", 499.99);

        // Display product details
        Console.WriteLine("Product 1:");
        Console.WriteLine("Name: " + product1.ProductName);
        Console.WriteLine("Price: $" + product1.Price);

        Console.WriteLine();

        Console.WriteLine("Product 2:");
        Console.WriteLine("Name: " + product2.ProductName);
        Console.WriteLine("Price: $" + product2.Price);
```

```
        Console.ReadLine(); // Keep the console window open
    }
  }

}
```

05)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace q5
{
  using System;
  using System.Collections.Generic;

  class LibraryBook
  {
    public string Title { get; set; }
    public string Author { get; set; }
    public bool Available { get; set; }

    public LibraryBook(string title, string author, bool available)
    {
      Title = title;
      Author = author;
      Available = available;
    }
```

```
    public void BorrowBook()
    {
        if (Available)
        {
            Available = false;
            Console.WriteLine("Book '" + Title + "' by " + Author + " has been borrowed.");
        }
        else
        {
            Console.WriteLine("Sorry, the book '" + Title + "' by " + Author + " is not available.");
        }
    }
}

class Program
{
    static void Main(string[] args)
    {
        // Instantiate multiple book objects
        List<LibraryBook> books = new List<LibraryBook>
    {
        new LibraryBook("The Great Gatsby", "F. Scott Fitzgerald", true),
        new LibraryBook("To Kill a Mockingbird", "Harper Lee", false),
        new LibraryBook("1984", "George Orwell", true)
    };

        // Perform book borrowing
        Console.WriteLine("Library Status Before Borrowing:");
```

```
        DisplayLibraryStatus(books);

        Console.WriteLine("\nBorrowing a book...");
        books[0].BorrowBook();

        Console.WriteLine("\nLibrary Status After Borrowing:");
        DisplayLibraryStatus(books);

        Console.ReadLine(); // Keep the console window open
    }

    static void DisplayLibraryStatus(List<LibraryBook> books)
    {
        foreach (var book in books)
        {
            Console.WriteLine("Title: " + book.Title + ", Author: " + book.Author + ", Available: " +
book.Available);
        }
    }
  }

}
```

06)