Task :02

4.1

```
using System;
class BankAccount
{
 public double Balance { get; private set; }
 public BankAccount(double initialBalance)
 {
 Balance = initialBalance;
 }
 public void Deposit(double amount)
 {
 if (amount > 0)
 {
 Balance += amount;
 }
 }
}
class Program
{
 static void Main()
 { BankAccount account = new BankAccount(100.0);
 account.Deposit(20.0);
 Console.WriteLine($"Current Balance: {account.Balance:C}");
 }
}
```

4.2

```
 using System;
class Account
{
 private string accountName;
 private double balance;
 public Account(string name, double initialBalance)
 {
 accountName = name;
 balance = initialBalance;
 }
 public void Withdraw(double amount)
 {
```

```
 if (amount > 0 && amount <= balance)
 {
 balance -= amount;
 }
 else
 {
 Console.WriteLine("Withdrawal failed. Insufficient balance.");
 }
 }
 public void Deposit(double amount)
 {
 if (amount > 0)
 {
 balance += amount;

}
 }
 public override string ToString()
 {
 return $"Account Name: {accountName}, Balance: {balance:C}";
 }
}
class Program
{
 static void Main()
 {

 Account heikkisAccount = new Account("Heikki's account", 1000.0);
 Account personalAccount = new Account("Personal account",
0.0);
 heikkisAccount.Withdraw(100.0);
 personalAccount.Deposit(100.0);

 Console.WriteLine(heikkisAccount.ToString());
 Console.WriteLine(personalAccount.ToString());
 }
}
```

4.3

```
using System;
namespace DogNamespace
{
 public class Dog
 {
 private string name;
 private string breed;
 private int age;
 public Dog(string name, string breed, int age)
 {
 this.name = name;
 this.breed = breed;
 this.age = age;
 }
 public void DisplayDogInfo()
 {
 Console.WriteLine("Dog Name: " + name);
 Console.WriteLine("Breed: " + breed);
 Console.WriteLine("Age: " + age + " years");

}
 }
 class Program
 {
 static void Main()
 {
 Dog myDog = new Dog("Fido", "Labrador", 3);
 myDog.DisplayDogInfo();
 }
 }
}
```

4.4

```
using System;
namespace RoomNamespace // You can choose your own namespace

{
 public class Room
 {
 private string code;
 private int seats;
 public Room(string classCode, int numberOfSeats)
 {
code = classCode;
 seats = numberOfSeats;
 }
 public void DisplayRoomInfo()
 {
 Console.WriteLine("Room Code: " + code);
 Console.WriteLine("Number of Seats: " + seats);
 }
 }
 class Program
 {
```

```
 static void Main()
 {
 Room myRoom = new Room("A101", 30);
 myRoom.DisplayRoomInfo();
 }
 }
}
```

4.5:

```
using System;
class Whistle
{
 private string sound;


public Whistle(string whistleSound)
 {
 sound = whistleSound;
 }
 public void MakeSound()
 {
 Console.WriteLine(sound);
 }
}
class Program
{
 static void Main()
 {

 Whistle myWhistle = new Whistle("Whoo-Whoo!");
 myWhistle.MakeSound();
 }
}
```

```
4.6:using System;
class Product
{
 private string name;
 private double price;
 private int quantity;
 public Product(string name, double price, int quantity)
 {
 this.name = name;

this.price = price;
```

```
 this.quantity = quantity;
 }
 public void PrintProduct()
 {
 Console.WriteLine("Product Information:");
 Console.WriteLine($"Name: {name}");
 Console.WriteLine($"Price: ${price:F2}");
 Console.WriteLine($"Quantity: {quantity}");
 }
}
class Program
{
 static void Main()
 { Product myProduct = new Product("Example Product", 19.99, 10);
 myProduct.PrintProduct();
 }
}
```

```
4.8:using System;
class Debt
{
 private double balance;
 private double interestRate;
 public Debt(double initialBalance, double initialInterestRate)
 {
 balance = initialBalance;

interestRate = initialInterestRate;
 }
 public void PrintBalance()
 {
 Console.WriteLine($"Current Balance: {balance:C}");
 }
 public void WaitOneYear()
 {
 double debtIncrease = balance * interestRate;
 balance += debtIncrease;
 }
}
class Program
{
 static void Main()
 {
 Debt myDebt = new Debt(1000.0, 0.05);

 myDebt.PrintBalance();
 myDebt.WaitOneYear();
myDebt.PrintBalance();
 }
}
```

4.9:

```
 class Dalmatian
{
 public string Name { get; set; }
 public int Spots { get; set; }
 public Dalmatian(string name, int spots)
 {
 Name = name;
 Spots = spots;
 }
}
class Program
{
 static void Main()
 {

 Dalmatian dalmatian = new Dalmatian("Buddy", 101);

 dalmatian.Name = "Lucky";
 dalmatian.Spots = 99;

 Console.WriteLine($"Name: {dalmatian.Name}");
 Console.WriteLine($"Spots: {dalmatian.Spots}");
 }
}
```

4.10

```
using System;

class Gauge
 {
 public int Value { get; set; }
 public Gauge()
 {
 Value = 0;
 }
 public void Increase()
 {
 if (Value < 5)
 {
```

```
 Value++;
 }
 }
 public void Decrease()
 {
 if (Value > 0)
 {
 Value--;
 }
 }
 public bool Full()
 {
 return Value == 5;
 }
 }
 class Program
 {
 static void Main()
 {
 Gauge gauge = new Gauge();


gauge.Increase();
 gauge.Increase();
 gauge.Increase();
 gauge.Decrease();
 gauge.Increase();


Console.WriteLine($"Current Value: {gauge.Value}");
 Console.WriteLine($"Is it Full? {gauge.Full()}");
 }
 }
```