

Παράδειγμα 1: Κλάση Person

Κλάση Person

```
public class Person
{
    // Ιδιότητες
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public int Age { get; set; }

    // Κατασκευαστής
    public Person(string firstName, string lastName, int age)
    {
        FirstName = firstName;
        LastName = lastName;
        Age = age;
    }

    // Μέθοδος για εμφάνιση πλήρους ονόματος
    public string GetFullName()
    {
        return $"{FirstName} {LastName}";
    }

    // Μέθοδος για έλεγχο ενηλικίωσης
    public bool IsAdult()
    {
        return Age >= 18;
    }
}
```

Χρήση της Κλάσης Person

```
public class Program
{
    public static void Main()
    {
        // Δημιουργία αντικειμένου Person
        Person person1 = new Person("John", "Doe", 25);

        // Χρήση ιδιοτήτων και μεθόδων
        Console.WriteLine($"Full Name: {person1.GetFullName()}"); // John Doe
        Console.WriteLine($"Is Adult: {person1.IsAdult()}"); // True
    }
}
```

Άσκηση 1: Κλάση Book

Περιγραφή

Δημιουργήστε μια κλάση Book που θα έχει τις εξής ιδιότητες και μεθόδους:

- Ιδιότητες: Title, Author, ISBN, Price
- Κατασκευαστής που θα αρχικοποιεί τις ιδιότητες
- Μέθοδο GetBookInfo που θα επιστρέφει πληροφορίες για το βιβλίο σε μορφή string

Λύση

```
public class Book
{
    // Ιδιότητες
    public string Title { get; set; }
    public string Author { get; set; }
    public string ISBN { get; set; }
    public double Price { get; set; }

    // Κατασκευαστής
    public Book(string title, string author, string isbn, double price)
    {
        Title = title;
        Author = author;
        ISBN = isbn;
        Price = price;
    }

    // Μέθοδος για επιστροφή πληροφοριών για το βιβλίο
    public string GetBookInfo()
    {
        return $"Title: {Title}, Author: {Author}, ISBN: {ISBN}, Price: ${Price}";
    }
}
```

Χρήση της Κλάσης Book

```
public class Program
{
    public static void Main()
    {
        // Δημιουργία αντικειμένου Book
        Book book1 = new Book("The Great Gatsby", "F. Scott Fitzgerald", "9780743273565", 10.99);

        // Χρήση μεθόδου
        Console.WriteLine(book1.GetBookInfo());
    }
}
```

Άσκηση 2: Κλάση BankAccount

Περιγραφή

Δημιουργήστε μια κλάση BankAccount που θα έχει τις εξής ιδιότητες και μεθόδους:

- Ιδιότητες: AccountNumber, AccountHolder, Balance
- Κατασκευαστής που θα αρχικοποιεί τις ιδιότητες
- Μέθοδο Deposit για κατάθεση χρημάτων
- Μέθοδο Withdraw για ανάληψη χρημάτων
- Μέθοδο GetAccountInfo που θα επιστρέφει πληροφορίες για τον λογαριασμό σε μορφή string

Λύση

```
public class BankAccount
{
    // Ιδιότητες
    public string AccountNumber { get; set; }
    public string AccountHolder { get; set; }
    public double Balance { get; private set; }

    // Κατασκευαστής
    public BankAccount(string accountNumber, string accountHolder, double initialBalance)
    {
        AccountNumber = accountNumber;
        AccountHolder = accountHolder;
        Balance = initialBalance;
    }

    // Μέθοδος για κατάθεση χρημάτων
    public void Deposit(double amount)
    {
        if (amount > 0)
        {
            Balance += amount;
            Console.WriteLine($"Deposited: ${amount}. New Balance: ${Balance}");
        }
        else
        {
            Console.WriteLine("Deposit amount must be positive.");
        }
    }

    // Μέθοδος για ανάληψη χρημάτων
    public void Withdraw(double amount)
    {
        if (amount > 0 && amount <= Balance)
        {
            Balance -= amount;
            Console.WriteLine($"Withdrew: ${amount}. New Balance: ${Balance}");
        }
    }
}
```

```

    }
    else
    {
        Console.WriteLine("Invalid withdraw amount.");
    }
}

// Μέθοδος για επιστροφή πληροφοριών για τον λογαριασμό
public string GetAccountInfo()
{
    return $"Account Number: {AccountNumber}, Account Holder: {AccountHolder}, Balance:
    ${Balance}";
}
}

```

Χρήση της Κλάσης BankAccount

```

public class Program
{
    public static void Main()
    {
        // Δημιουργία αντικειμένου BankAccount
        BankAccount account1 = new BankAccount("123456789", "Alice Smith", 1000.00);

        // Χρήση μεθόδων
        Console.WriteLine(account1.GetAccountInfo());

        account1.Deposit(200.00); // Κατάθεση
        account1.Withdraw(150.00); // Ανάληψη

        Console.WriteLine(account1.GetAccountInfo());
    }
}

```

Άσκηση 3: Κλάση Rectangle

Περιγραφή

Δημιουργήστε μια κλάση Rectangle που θα έχει τις εξής ιδιότητες και μεθόδους:

- Ιδιότητες: Length, Width
- Κατασκευαστής που θα αρχικοποιεί τις ιδιότητες
- Μέθοδο GetArea για υπολογισμό της επιφάνειας του ορθογωνίου
- Μέθοδο GetPerimeter για υπολογισμό της περιμέτρου του ορθογωνίου

Λύση

```

public class Rectangle
{
    // Ιδιότητες
    public double Length { get; set; }
    public double Width { get; set; }

    // Κατασκευαστής
    public Rectangle(double length, double width)
    {
        Length = length;
        Width = width;
    }

    // Μέθοδος για υπολογισμό της επιφάνειας
    public double GetArea()
    {
        return Length * Width;
    }

    // Μέθοδος για υπολογισμό της περιμέτρου
    public double GetPerimeter()
    {
        return 2 * (Length + Width);
    }
}

```

Χρήση της Κλάσης Rectangle

```

public class Program
{
    public static void Main()
    {
        // Δημιουργία αντικειμένου Rectangle
        Rectangle rect1 = new Rectangle(5.0, 3.0);

        // Χρήση μεθόδων
        Console.WriteLine($"Area: {rect1.GetArea()}"); // Area: 15.0
        Console.WriteLine($"Perimeter: {rect1.GetPerimeter()}"); // Perimeter: 16.0
    }
}

```