# План на урока: Класове и обекти в Обектно Ориентираното Програмиране (ООП)

Клас: 11-ти клас

Продължителност: 40–45 минути

Цели на урока:

1. 1. Учениците да разберат концепциите за клас и обект в контекста на ООП.
2. 2. Да се запознаят с основните термини и понятия – полета, свойства, конструктори и методи.
3. 3. Да се научат как да създават прости класове и обекти, и да разбират тяхното приложение.

## 1. Въведение и цел на урока (5 минути)

Представяне на темата: "Днес ще разгледаме концепциите за класове и обекти, които са основни компоненти на Обектно Ориентираното Програмиране (ООП). Ще научим какво представляват класовете и как да ги използваме в нашите програми."

Обсъждане на целите на урока: Обяснение на основните термини и концепции на ООП и тяхното приложение.

## 2. Основни понятия и дефиниции (10 минути)

Клас и обект:

Класът е шаблон или модел за създаване на обекти, като съдържа определение за характеристиките и поведението на обекта.

Обектът е конкретна инстанция на класа, създадена по този шаблон, която може да има различни стойности за характеристиките, но споделя същото поведение.

Полета и свойства:

- Полета: Променливи, които съхраняват състоянието на обекта (данни).

- Свойства: Специални методи, които осигуряват контрол върху достъпа и манипулирането на полетата. Свойствата обикновено имат get и set методи.

Методи:

Методите са функции, които дефинират поведението на обекта и му позволяват да извършва действия.

Конструктор:

Конструкторът е специален метод, който се изпълнява при създаване на обект и обикновено инициализира полетата на класа.

## 3. Пример за дефиниране на прост клас (10 минути)

Задача: Дефиниране на клас `Person`, който съдържа име, възраст и метод за отпечатване на информация за човека.

Примерен код в C#:  
  
public class Person  
{  
 // Полета  
 private string name;  
 private int age;  
  
 // Конструктор  
 public Person(string name, int age)  
 {  
 this.name = name;  
 this.age = age;  
 }  
  
 // Свойства  
 public string Name  
 {  
 get { return name; }  
 set { name = value; }  
 }  
  
 public int Age  
 {  
 get { return age; }  
 set   
 {   
 if (value >= 0)  
 age = value;   
 }  
 }  
  
 // Метод  
 public void PrintInfo()  
 {  
 Console.WriteLine($"Име: {name}, Възраст: {age}");  
 }  
}

## 4. Създаване на обекти и използване на методи (10 минути)

Примерен код за използване на класа `Person`:

using System;  
  
class Program  
{  
 static void Main(string[] args)  
 {  
 // Създаване на обект на класа Person  
 Person person1 = new Person("Анна", 20);  
   
 // Използване на метода PrintInfo  
 person1.PrintInfo();  
  
 // Промяна на свойствата чрез сетъри  
 person1.Name = "Борис";  
 person1.Age = 25;  
 person1.PrintInfo();  
 }  
}

## 5. Упражнение (5 минути)

Задача за учениците:

Създайте клас `Car`, който съдържа поле за марка, модел и година на производство. Добавете конструктор, свойства за тези полета и метод `PrintCarInfo()`, който отпечатва информация за автомобила.

## 6. Резюме и въпроси (5 минути)

Преглед на основните понятия от урока:

- Какво са класовете и обектите в ООП.

- Какво са полета, свойства, конструктори и методи.

- Как да създаваме обекти и да използваме методите на класа.

Отговор на въпроси от учениците:

Отговорете на въпроси от учениците и обяснете неясни моменти. Подчертайте значението на обектите и класовете в програмирането и как те помагат за организиране на кода.