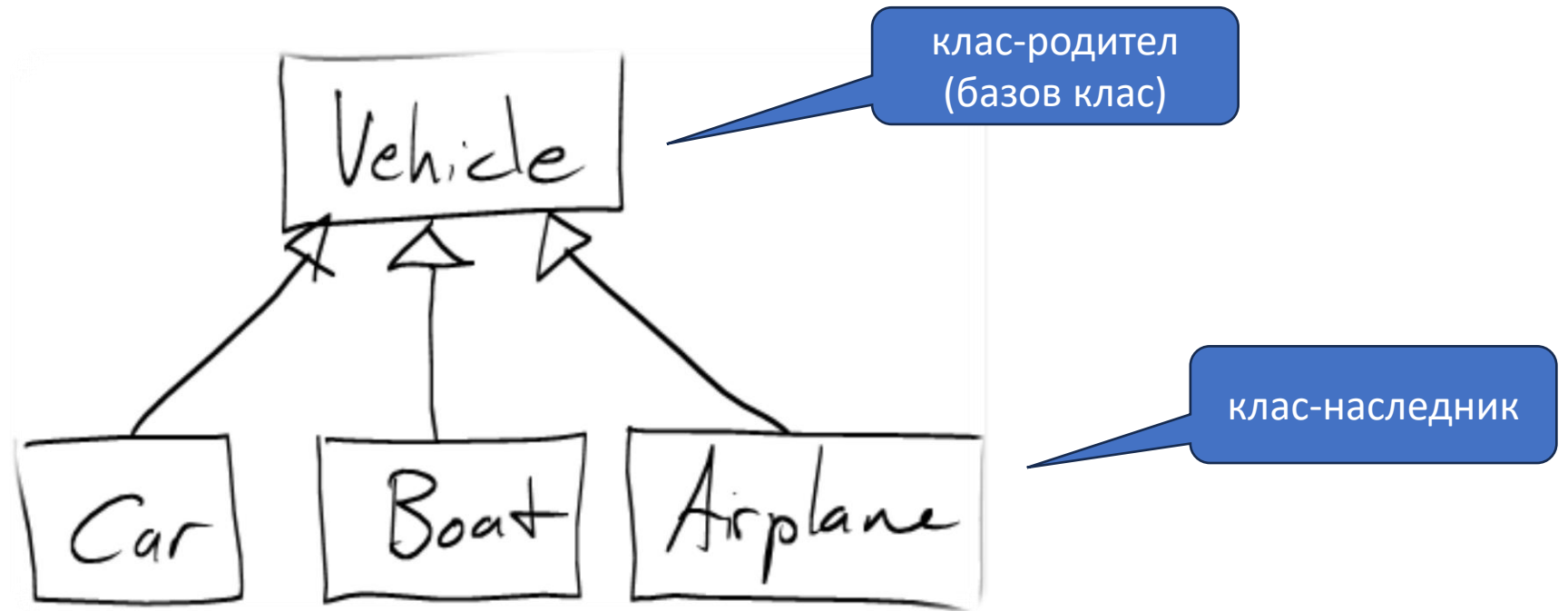


ЙЕРАРХИЯ ОТ КЛАСОВЕ. НАСЛЕДЯВАНЕ

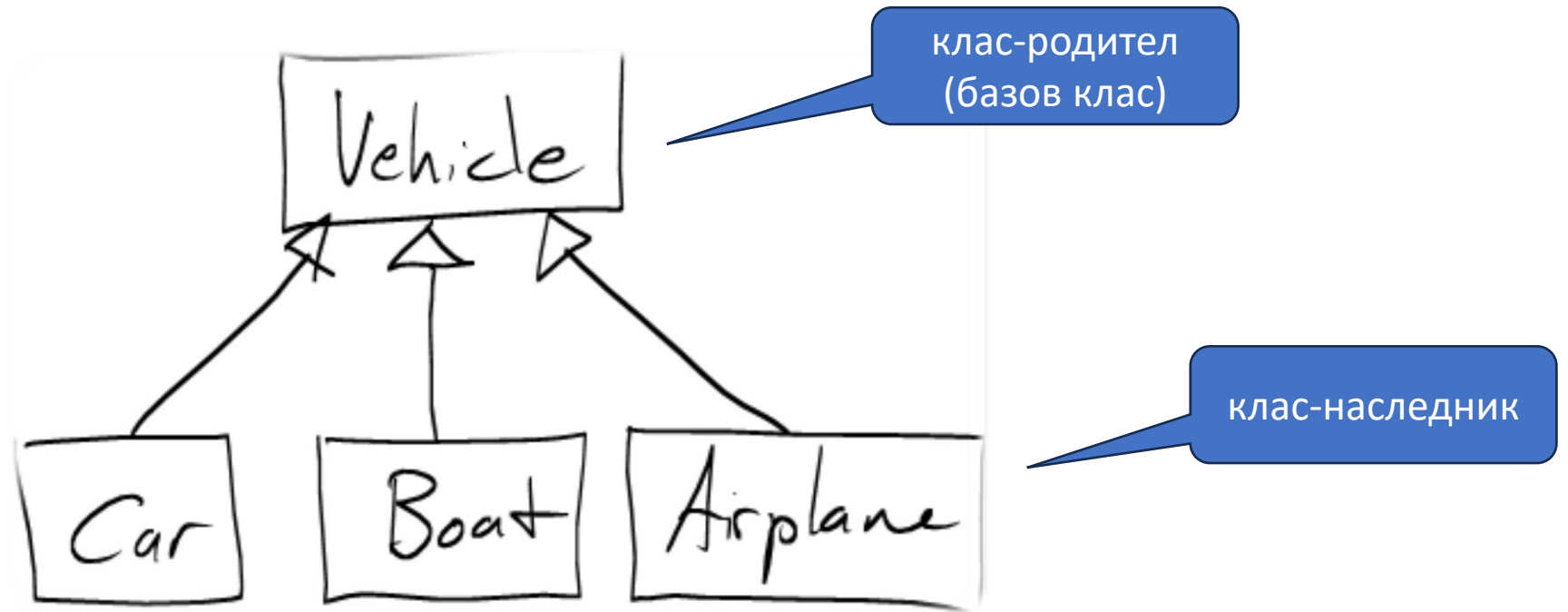
1. Наследяване
2. Йерархия от класове
3. Достъп до членовете на базовия клас
4. Модификаторът **protected** при наследяване
5. Пренаписване на методи
6. Задачи

Наследяване – разширяване на класове



- **Базов клас (Родител)** – съдържа всички общи атрибути и методи и ги предава на класовете наследници (**производни**)
- Класове **наследници** – съдържат специфични елементи за класа наследник

Наследяване



- Класа-наследник създаваме с конструкцията:

class <клас наследник> : <клас-родител>

```
class Car : Vehicle { тяло на класа-наследник }
```

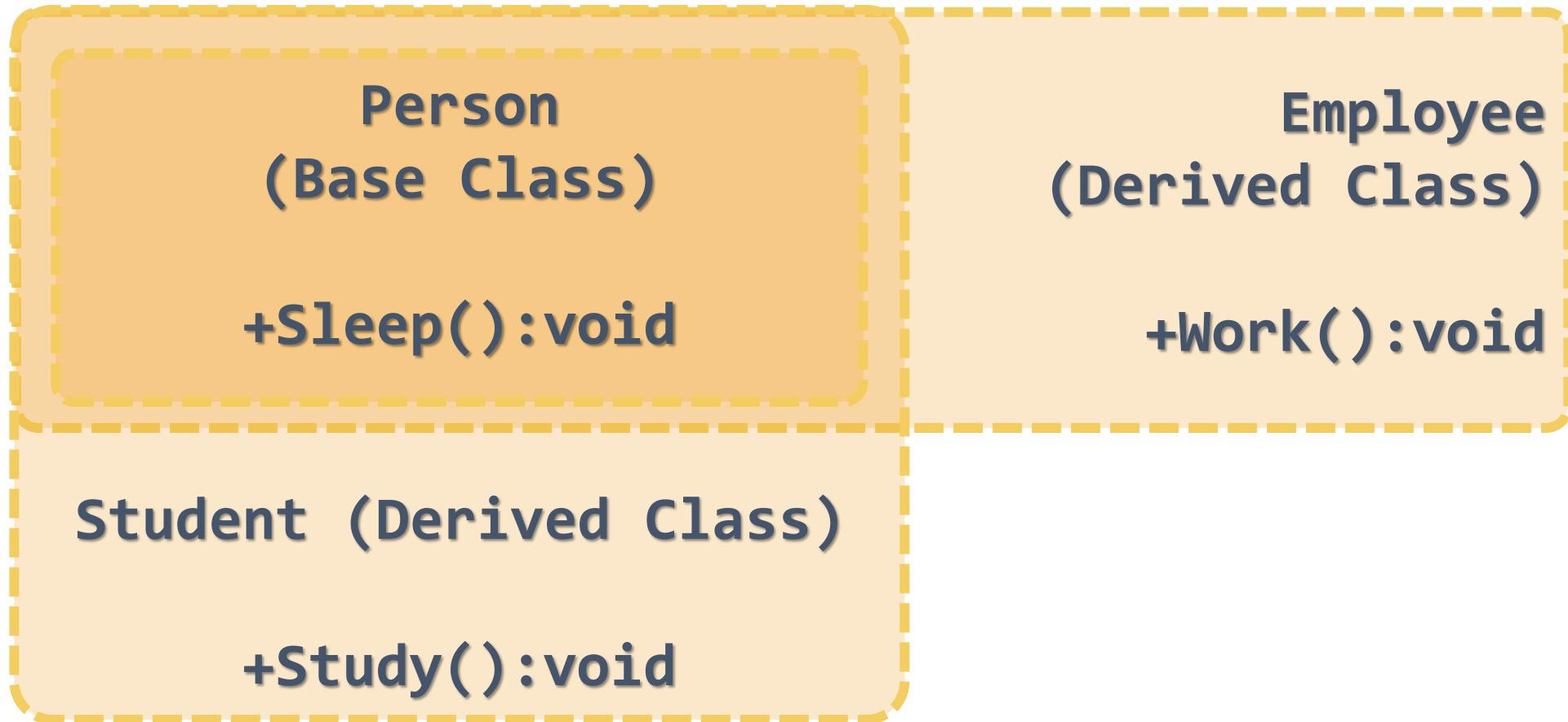
Наследяване

- Конструкторите **не се наследяват**
- Конструкторите **може да се ползват** от класовете наследници

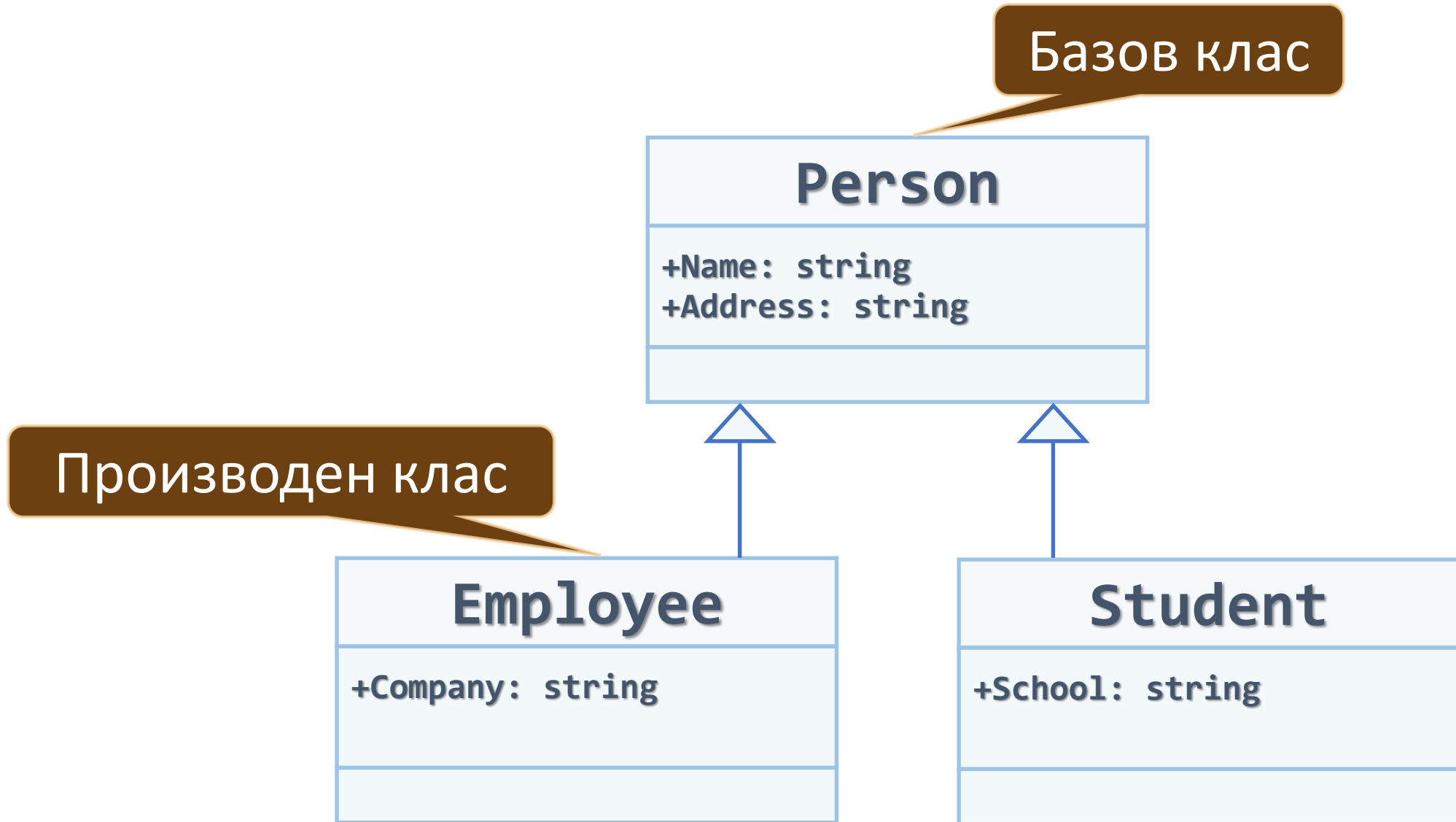
```
class Student : Person {  
    private School school;  
    public Student(String name, School school)  
        : base(name)  
    {  
        this.school = school;  
    }  
}
```

Наследяването е разширение

- Инстанцията на класа наследник **съдържа** инстанция на класа родител

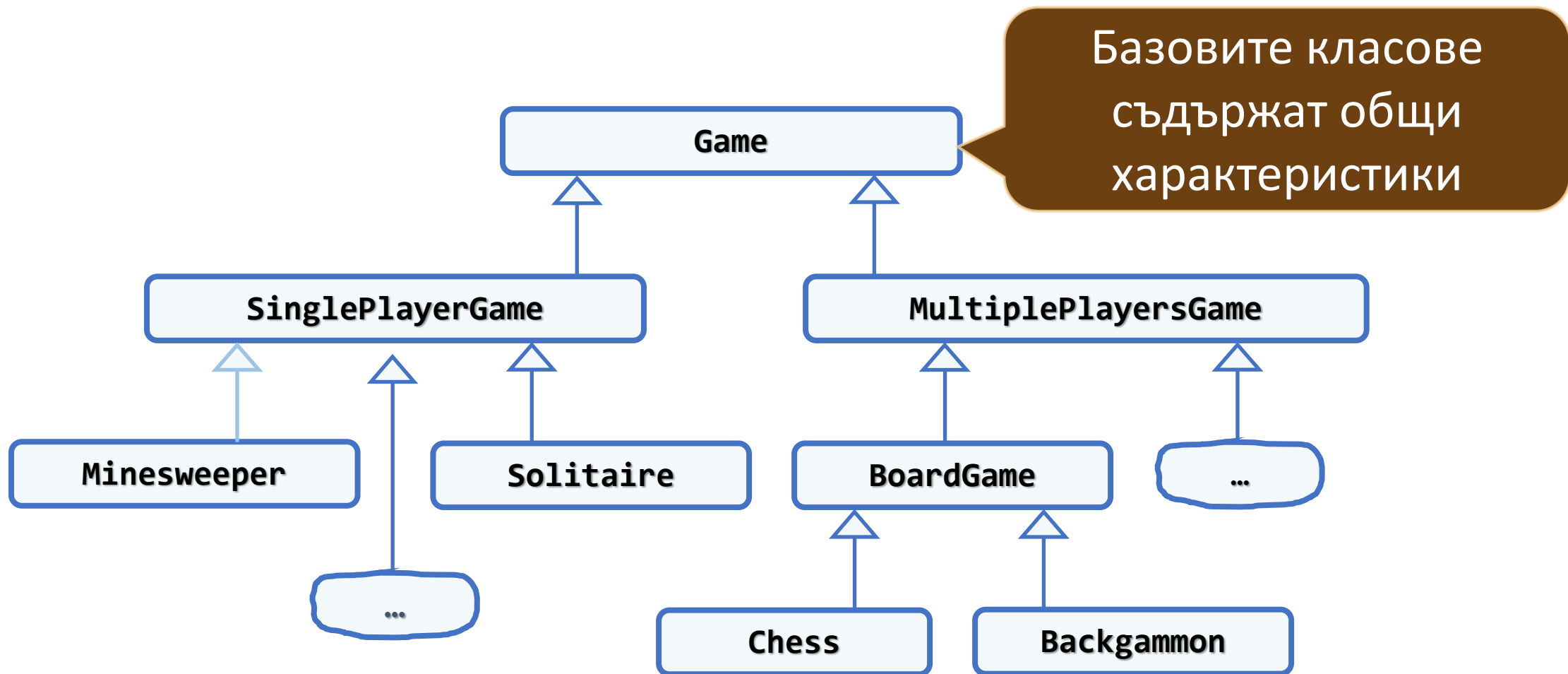


Наследяване – пример



Йерархия от класове

- Наследяването води до йерархия от класове



Достъп до членовете на базовия клас

- Чрез ключовата дума **base**

```
class Person { ... }  
class Employee : Person  
{  
    void Fire(string reasons) {  
        Console.WriteLine  
            ($"{base.name} got fired because {reasons}");  
    }  
}
```


Модификаторът `protected` при наследяване

- Подобно на модификатора `private` скрива членовете на класа за другите класове с изключение на класовете наследници.

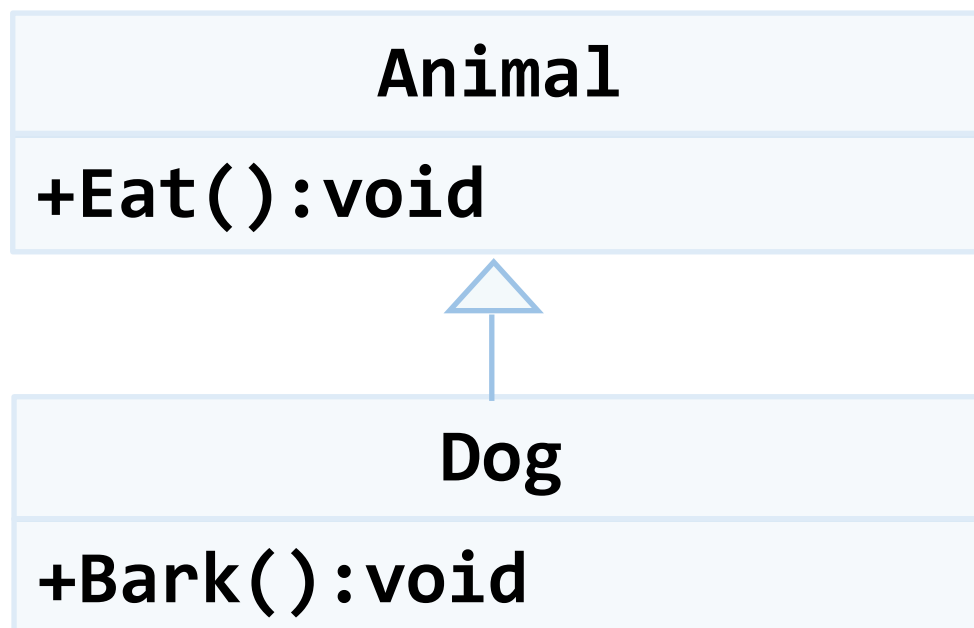
Пренаписване на методи

- Ако в базовия клас метод е деклариран като **virtual**, то той може да бъде **пренаписан** в класа наследник с ключовата дума **override**, като това ще промени поведението на този метод.

```
public class Animal
{
    protected virtual void MakeSound()
    {
        Console.WriteLine("Animal makes a generic sound.");
    }
}

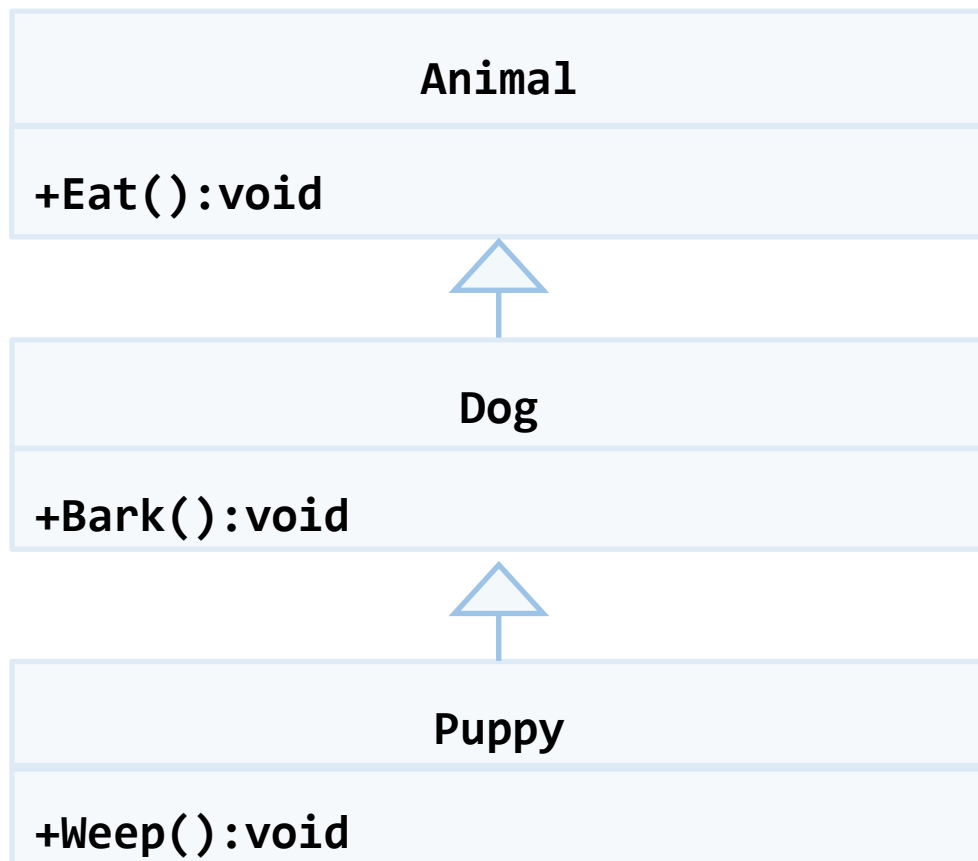
public class Cat : Animal
{
    protected override void MakeSound()
    {
        Console.WriteLine("Meow! The cat makes a sound.");
    }
}
```

Задача - наследяване



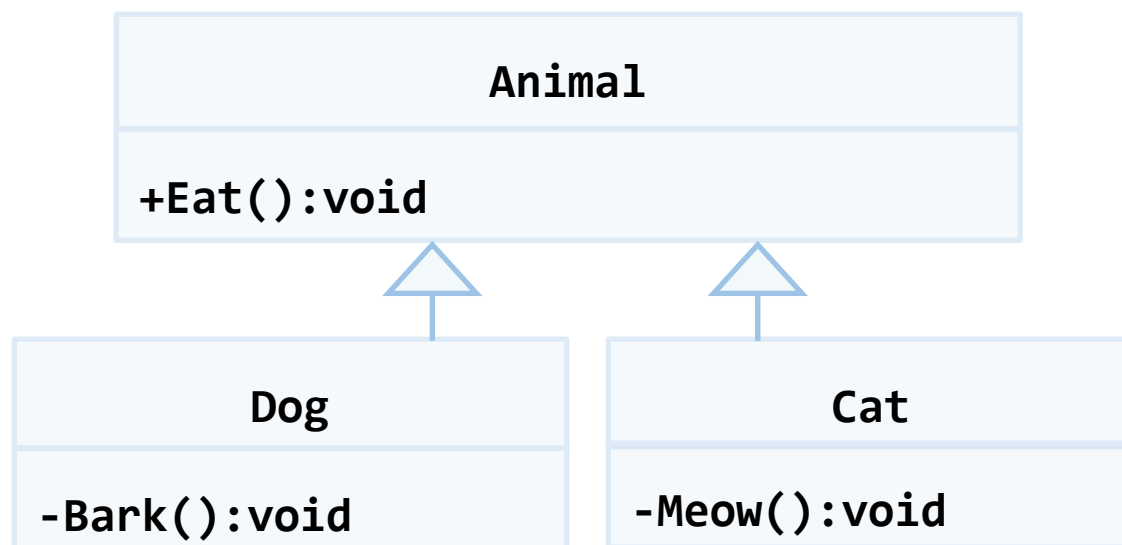
```
Dog dog = new Dog();  
dog.Eat();  
dog.Bark();
```

Наследяване на няколко нива



```
Puppy puppy = new Puppy();  
puppy.Eat();  
puppy.Bark();  
puppy.Weep();
```

Йерархично наследяване



```
Dog dog = new Dog();  
dog.Eat();  
dog.Bark();
```

```
Cat cat = new Cat();  
cat.Eat();  
cat.Meow();
```