

Что такое анонимные типы в C#? В чем их преимущества и недостатки?

Анонимные типы в C# позволяют создавать объекты с определенным набором свойств без определения класса. После определения свойства нельзя изменять, но в остальном их можно использовать как обычные свойства.

Пример определения анонимного типа:

```
var figure1 = new { Type = "Rectangle", S = 10 };
```

При этом для объекта figure1 будет создан анонимный тип и если добавить в программу строчку:

```
var figure2 = new { Type = "Square", S = 13 };
```

то для анонимные типы для figure1 и figure2 будут совпадать. Однако данные объекты нельзя преобразовать к какому-нибудь другому классу (в общем случае – типу), даже при условии наличия подобного набора свойств.

Также можно создавать массивы из объектов анонимных типов:

```
Var figures = new[]  
{  
    new { Type="Rect"},  
    new { Type="Square"}  
}
```

Преимущество состоит в том, что для локального объединения данных в единый объект нет необходимости определять дополнительный класс (в случае с которым при необходимости изменить или добавить свойство будет нужно менять сам класс, который может использоваться только конкретно в этой части кода, а в случае с анонимным типом это сделать куда проще прямо в этой части кода на моменте определения переменной).

Недостатком, как уже было сказано ранее, является невозможность изменять свойства после их инициализации.