# Записи

*Запись* (record) – ссылочный тип. Тип неизменяемый (immutable), поэтому более безопасный.

Пример неизменяемого типа: String

Создание типа структуры:

*модификатор record [struct] ИмяЗаписи*

*{// данные записи*

*// функции записи (могут отсутствовать)}*

Особенности записи:

* по умолчанию переопределяют ToString (возвращают название типа и значения каждого свойства)
* по умолчанию переопределяют Equals (сравниваются значения каждого свойства двух объектов)
* классы записи могут наследоваться
* по умолчанию создаются записи-классы, но если указать record struct, то создастся запись-структура
* чтобы запретить изменение значения свойства, вместо set пишется init.
* для изменения значения свойства, в котором есть init, используется конструктор
* можно создать позиционные записи — часть свойств будет указана в первичном конструкторе

Пример создания записи для хранения информации о человеке:

|  |  |
| --- | --- |
| Создание типа записи | Использование переменных типа запись |
| public record Person  {public string Name { get; set; }  public Person(string name) => Name = name;}  public record struct Person  {public string Name { get; set; }  public Person(string name) => Name = name;}  // позиционная запись  public record Person(string Name, int Age);  ---  public record Person  {public string Name { get; init; }  public Person(string name) => Name = name;} | var person = new Person("Tom");  person.Name = "Bob";  var person1 = new Person("Tom");  var person2 = new Person("Tom");  Console.WriteLine(person1.Equals(person2)); // true  Console.WriteLine(person1 == person2); // true  Console.WriteLine(person1 != person2); // false  ---  var person = new Person("Tom");  person.Name = "Bob"; // ошибка |