

Коротков Никита ИУ5-34

Работа с кортежами в языке C# на основе нового синтаксиса.

Для работы с новым синтаксисом кортежей необходимо подключить пакет «System.ValueTuple» с помощью диспетчера пакетов NuGet.

Кортеж может быть объявлен следующим образом с именованными параметрами:

```
(string strRK, int intRK) tuple1 = ("Привет", 13);
```

```
Console.WriteLine(tuple1.strRK);
```

```
Console.WriteLine(tuple1.intRK);
```

Результаты вывода в консоль:

Привет

13

Таким образом, тип-кортеж создается на лету и инициализируется значениями. В отличие от старого синтаксиса кортежей, где у полей были имена Item1 ... ItemN, в новом синтаксисе все поля имеют собственные имена. Имена полей можно указывать в правой части, в этом случае компилятор автоматически выводит типы полей:

```
var tuple2 = (strRK: "Привет", intRK: 13);
```

```
Console.WriteLine(tuple2.strRK);
```

```
Console.WriteLine(tuple2.intRK);
```

Кортеж является изменяемым типом данных, полям можно присваивать новые значения: tuple2.strRK = "Пока";

Кортежи можно использовать в качестве входных параметров функций. Например:

```
public static void InputTuple((string strA, int intA) tupleRK) {}
```

Вызываем:

```
InputTuple(tuple2);
```

Обратите внимание, что имена параметров кортежей tuple2 и tupleRK не совпадают. Но ошибки не возникает, потому что совпадают сигнатуры этих кортежей (у обоих два параметра, первый типа string, а второй типа int).

Кортежи также можно использовать в качестве выходных параметров функций. Например:

```
public static (string strRK, int intRK) OutputTuple()
{ return ("Привет", 13); }
```

Ее можно вызвать следующим образом, в этом случае переменная tuple3 будет кортежем:

```
var tuple3 = OutputTuple();
```

Можно при вызове произвести разыменование кортежа, в этом случае переменные str1 и int1 будут заполнены значениями из соответствующих полей кортежа:

```
(string str1, int int1) = OutputTuple();
```

Обратите внимание, что имена параметров кортежей (string strRK, int intRK) и (string str1, int int1) не совпадают. Но ошибки не возникает, так как совпадают сигнатуры кортежей.

Новый синтаксис кортежей упрощает создание типов, их передачу в функции и возвращение из функций.