

РК №1 по курсу БКИТ

Демченков Д.А., РТ5-31

Вопрос:

Как реализованы операторы сопоставления с образцом в языке C#?

Ответ:

Операторы сопоставления с образцом (pattern matching) – это классическая особенность языков программирования, использующих функциональный подход.

Сопоставление с образцом похоже на условные операторы и поэтому построено на их основе.

Пусть дан массив объектов, который содержит строки и целые числа:

```
object[] array1 = { 1, "строка 1", 2, "строка 2", 3 };
```

Для того чтобы расширить условный оператор до оператора сопоставления с образцом в язык была добавлена конструкция is.

Пример сопоставления с образцом на основе условного оператора if:

```
foreach(object obj in array1)
{
    if(obj is int i1)
    {
        Console.WriteLine("Число : " + i1.ToString());
    }
    else if (obj is string s1)
    {
        Console.WriteLine("Строка : " + s1);
    }
}
```

В данном примере в цикле foreach по очереди перебираются элементы массива.

Если элемент массива соответствует целому числу, то он автоматически приводится к целому типу и помещается в переменную i1. Далее с ним можно выполнять необходимые действия.

Если элемент массива соответствует строке, то он автоматически приводится к строке и помещается в переменную s1.

Вторым вариантом сопоставления с образцом в C# является использование оператора switch.

Пример сопоставления с образцом на основе оператора switch:

```
foreach (object obj in array1)
{
    switch(obj)
    {
        case string s1:
            Console.WriteLine("Строка -> " + s1);
            break;
        case int i1 when i1 > 2:
            Console.WriteLine("Число больше 2 -> " +
                i1.ToString());
            break;
        case int i1:
            Console.WriteLine("Число -> " + i1.ToString());
            break;
    }
}
```

В этом случае в каждом операторе case указывается проверяемый тип и переменная, в которую сохраняется результат приведения типа.

С использованием ключевого слова when можно задавать так называемые «охранные выражения», которые накладывают дополнительные ограничения на проверку. В данном примере

возможность приведения к целому типу проверяется дважды, случай $i1 > 2$ рассматривается отдельно и возникновение такого случая проверяется с помощью охранного выражения.