

## НОВІ МОЖЛИВОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОЇ МОВИ ПРОГРАМУВАННЯ C#

Об'єктно-орієнтована мова програмування C# від компанії Microsoft на сьогодні є однією з найпоширеніших у світі і займає четвертий рядок в рейтингу популярних мов програмування сайту TIOBE [1].

Широке розповсюдження та популярність цієї мови програмування зумовлюється великим спектром задач, які можна вирішувати за її допомогою – починаючи від розробки програмного забезпечення для мобільних пристроїв і закінчуючи створенням корпоративних Web-порталів та програм для хмарних обчислень.

Останнім часом компанія Microsoft зробила цілий ряд суттєвих доповнень до вже існуючих можливостей цієї мови програмування, які повинні ще більше підвищити продуктивність праці розробників на платформі .NET, зробити процес програмування більш ефективним, а програмний код більш компактним та надійним.

Найбільш значущими доповненнями мови C# є шаблони, кортежі, локальні функції та інші [2].

### Зіставлення з шаблоном.

В останній версії C# 7.0 вводиться поняття шаблонів (pattern), які, абстрактно кажучи, є синтаксичними елементами мови, що можуть перевіряти змінну на відповідність певному шаблону та отримувати у разі відповідності її значення.

Прикладами шаблонів в C# 7.0 є:

- Константні шаблони виду `c` (де `c` є постійним виразом в C#), які перевіряють, дорівнює ця змінна константі чи ні.

- Шаблони виду `T x` (де `T` є типом, а `x` - змінною), які перевіряють, чи має змінна тип `T`, і якщо так, то передає її значення в нову змінну `x` типу `T`.

- `var` шаблони виду `var x` (де `x` є змінною), які завжди обчислюється та використовується для створення нової змінної `x` з тим же типом і з тим же значенням.

Для підтримки шаблонів було внесені зміни у певні конструкції мови:

- `is` тепер може мати у якості правого аргументу не лише тип, а і шаблон;

- `case` в операторі `switch` може тепер використовувати як константи, так і шаблони.

### Кортежі

Іноді буває потрібно повернути більше одного значення з методу. Наявні на сьогоднішній день варіанти не є оптимальними:

- `Out` параметри мають дещо складний синтаксис і не будуть працювати з асинхронними методами.

- `System.Tuple <...>` вимагає створення окремого додаткового об'єкта.

- Побудова нового типу для кожного методу вимагає багато коду.

Для того, щоб позбутися цих проблем, в C# 7.0 було додано кортежі і літералів кортежів.

Кортежі є типами значень, а їх елементи є просто відкритими полями, які можуть змінюватися. Кортежі можна порівнювати, а це означає, що два кортежі рівні (і мають один і той же хеш-код), якщо всі їхні елементи попарно рівні (і мають один і той же хеш-код).

Це робить кортежі корисними не лише для повернення декількох значень з методу. Наприклад, якщо потрібен словник з декількома ключами, то можна використовувати кортеж як ключ. І в ситуації, коли потрібен список з кількома значеннями в кожному положенні, також можна використовувати список кортежів.

### Локальні функції.

Іноді допоміжна функція має сенс тільки всередині одного методу, який її використовує. Тепер можна оголосити такі функції всередині інших функцій в якості локальної функції. Параметри і локальні змінні з області видимості зовнішньої функції доступні всередині локальної функції, так само, як і для лямбда-виразів.

Таким чином, основні новації в мові C# 7.0 повинні полегшити роботу з даними, спростити код програми та покращити продуктивність праці розробників.

## Список літератури

1. *TIOBE Index for February 2017* [Електронний ресурс]. – Режим доступу к ресурсу: <http://www.tiobe.com/tiobe-index?title=Персистентные структуры данных>
2. *What's New in C# 7.0* [Електронний ресурс]. – Режим доступу к ресурсу: <https://blogs.msdn.microsoft.com/dotnet/2016/08/24/whats-new-in-csharp-7-0>