



Průlet verzemi C#

Michal Altair Valášek
development & security consultant

michal.valasek@altairis.cz
www.altair.blog



C# je

- univerzální
- imperativní
- procedurální
- objektový
- introspektivní
- reflexivní
- staticky typovaný jazyk
- s unifikovaným systémem typů

Demo: Od delegátů k lambda expressions

- 1.0** Delegates
- 2.0** Iterators and yield statement
- 2.0** Generic types
- 2.0** Anonymous methods
- 3.0** Lambda expressions
- 3.0** Extension methods
- 3.0** LINQ

Demo: Vlastnosti vlastností

- 1.0** Koncept properties
- 2.0** Rozdílná přístupnost get/set
- 3.0** Auto-implemented proproperties (`prop\t\t`)
- 6.0** getter-only property
(lze nastavit v konstruktoru nebo jako default value)
- 6.0** expression-bodied property
- 9.0** `init` setter pro immutable properties
(lze nastavit při vytváření)
- 7.3** Atributy pro backing field
(např. `[field:NonSerialized]`)
- 8.0** Možnost nastavit `[Obsolete]` pro accessory jednotlivě

Demo: Argumenty metod

- 1.0 Overloading (např. pro nepovinné parametry)
- 1.0 Param arrays
- 1.0 Výstupní argumenty – out
- 7.0 Konstrukce `out var` pro výstupní parametry
- 7.0 Discards
- 7.2 Vstupní argumenty – `in` (jsou read only)
- 4.0 Volitelné argumenty (příp. s výchozí hodnotou)
- 4.0 Pojmenované argumenty
- 5.0 Caller info attributes
[CallerMemberName], [CallerPath], [CallerLineNumber]
- 10.0 [CallerArgumentExpression]
- 6.0 Operátor `nameof`

Demo: Using namespaces

- 2.0** Type and namespace aliases
`using C = System.Console;`
- 6.0** Using static
`using static System.Math;`
- 10.0** File-scoped namespaces
- 10.0** Global using directives
- 10.0** Global using in csproj file
`<Using Include="..." />`
- 10.0** Implicit usings
- 10.0** Top-level statements

Demo: Strings

- 1.0** `String.Format`, včetně zarovnání a formátovacích řetězců
- 1.0** Verbatim strings (`@`, multiline)
- 6.0** String interpolation `$`, `$@`
- 8.0** Alternative interpolated verbatim strings `@$`
- 10.0** Constant interpolated strings

Demo: Netradiční operátory

- 1.0** Ternary conditional operator `?:`
- 1.0** Null-coalescing operator `??`
- 7.0** Throw expressions po `?:` a `??` nebo jako tělo lambda
- 8.0** Null-coalescing assignment operator `??=`
- 6.0** Null-conditional access operators `? .` a `? []`

Demo: Různé

- 3.0** Anonymous types
- 6.0** Exception filters (catch when)
- 7.0** Digit separator
- 7.0** Binary literals (0b)
- 7.2** Digit separator after 0b and 0x
- 8.0** Readonly members
- 8.0** Ranges and indexes [x..y] [^x] atd.
- 8.0** Enhanced usings with default scope



✉ michal.valasek@altairis.cz

📘 facebook.com/rider.cz

🐙 github.com/ridercz

🐦 twitter.com/ridercz

🌐 linkedin.com/in/ridercz

🌐 www.altairis.cz

www.rider.cz

www.altair.blog