



Vyšší odborná škola obalové techniky
a střední škola, Štětí

Digitální učební materiály

Programování - Programování C#

Ivan Pomykacz



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Licence



Digitální učební materiály, jejímž autorem je Ivan Pomykacz, podléhají licenci [Creative Commons: Uvedte autora - Nevyužívejte dílo komerčně - Zachovejte licenci 3.0 Unported](https://creativecommons.org/licenses/by-nc-sa/3.0/).

Vytvořeno na základě tohoto díla: <http://dumy.odbornaskola.cz/pomykacz>

Práva nad rámec této licence jsou popsána zde: <http://dumy.odbornaskola.cz/pomykacz>.

Disclaimer

Tento PDF dokument byl strojově vygenerován z HTML stránek

<http://dumy.odbornaskola.cz/pomykacz/>.

Je tedy možné, že sazba textu může obsahovat chyby. Jde převážně o vizuální a typografické chyby, které mohou narušit plynulou čitelnost textu. V některých případech může být ohrožena i funkčnost některých komponent (jako vnitřní odkazy).

Vzhledem k tomu, že vypracované materiály nebyly nikdy určeny pro výstupní formát PDF, autor se zříkává jakékoli odpovědnosti za nalezené chyby. Nesnažte se proto v této souvislosti autora kontaktovat.

Programování

Programování C#

Obsah

- Výstup

Výstup

Název školy	Vyšší odborná škola obalové techniky a Střední škola, Štětí, příspěvková organizace		
Adresa školky	Kostelní 134, 411 08 Štětí		
IČ	46773509		
Název operačního programu	OP Vzdělávání pro konkurenceschopnost		
Registrační číslo	CZ.1.07/1.5.00/34.1006		
Označení vzdělávacího materiálu	VY_32_INOVACE_23_PRG_447		
Název tématické oblasti (sady)	Programování		
Název materiálu	Výstup		
Anotace	Text představuje metody pro výstup textově orientovaných informací na monitor (do terminálu). Zároveň ukazuje možnosti výpisu hodnot uložených v proměnných.		
Autor	Ivan Pomykacz	Jazyk	čeština
Očekávaný výstup	Vypíše text nebo hodnotu proměnné na obrazovku (do terminálu). Spojuje do výstupu hodnoty z různých zdrojů pomocí operátoru.		
Klíčová slova	výstup		
Druh výukového zdroje	Výklad	Věková skupina žáků	15+
Typ interakce	aktivita	Ročník	1.
Speciální vzdělávací potřeby	žádné		
Zhotoveno, (datum/období)	19.09.2013	Celková velikost	

Obsah

- [Shrnutí](#)
- [Console](#)
 - [Argumenty](#)
 - [Spojování řetězců](#)
 - [Jiný způsob](#)

Shrnutí

Pod výstupem programu si můžeme představit cokoli, co opustilo `{..}` a je to vidět někde jinde než ve zdrojovém kódu. Např. zobrazení textu "Ahoj světe!" na monitoru. Jako výstupní zařízení byl v tomto případě zvolen monitor, resp. grafická karta. A ano, ve finále jde o okno terminálu, kde se text zobrazí.

Dalším výstupním zařízením by mohla být tiskárna. Ale zrovna tak by to mohl být soubor na pevném disku.

Console

Zaměřme se na první popisovaný výstup, tj. do konzole (terminálu, příkazové řádky, shellu atp.). V C# je dostupný objekt `Console`, který obsahuje metody `Write()` a `WriteLine()`. Tyto se liší pouze tím, že druhá jmenovaná provede po vypsání textu řádkový zlom (vloží stisk kl. Enter).

Vyzkoušejte chování

```
Console.Write("Ahoj");  
Console.Write("Světě");  
Console.Write("!");
```

a následně

```
Console.WriteLine("Ahoj");  
Console.WriteLine("Světě");  
Console.WriteLine("!");
```

Argumenty

Prozatím si řekněme, že jak metoda `Write()`, tak `WriteLine()` přebírají vstupní argument typu `string`.

Tzn., že můžeme metodě předat buď řetězec `"Ahoj svete!"` (včetně uvozovek), jak jste mohli vidět výše. Nebo proměnnou typu `string`.

```
string text;  
text = "Ahoj světe!";  
Console.WriteLine(text);
```

Spojování řetězců

Určitě se nám stane, že budeme chtít na výstup zkombinovat text a hodnoty uložené v různých proměnných. Jako např. v následujícím příkladu:

```
string ahoj;  
int pocet;
```

```
ahoj = "Ahoj světe!"  
pocet = 4;
```

```
Console.WriteLine(ahoj + " Už po " + pocet + ".");
```

Všimněte si, že v příkladu se míchá dokonce i typ `int`. Jak to, že to funguje? Proměnná `pocet` je objekt typu `int`, který obsahuje mimo jiné metodu `ToString()`, která se použije, když je potřeba.

Pokud se kombinuje více proměnných s textem, je argument metody `WriteLine()` poněkud nepřehledný. Zejména proto, jak se neustále spojují jednotlivé části pomocí operátoru `+` a potom ty uvozovky.

Jiný způsob

Pro výpis proměnných existuje proto ještě další způsob, který může být přehlednější. Vizte ukázkou:

```
string ahoj;  
int pocet;
```

```
ahoj = "Ahoj světe!"  
pocet = 4;
```

```
Console.WriteLine("{0} Už po {1}.", ahoj, pocet);
```

Do složených závorek se uvede pořadí proměnné, které jsou umístěny za úvodním řetězcem, jako další argumenty funkce. Pozor, čísluje se od nuly.