Vyšší odborná škola obalové techniky
a střední škola, Štětí

Digitální učební materiály

Programování - Programování C#

Ivan Pomykacz

esf 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: Uveďte autora - Nevyužívejte dílo komerčně - Zachovejte licenci 3.0 Unported.
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

- Referenční příručka

Referenční příručka

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_444		
Název tématické oblasti (sady)	Programování		
Název materiálu	Referenční příručka		
Anotace	Materiál shrnuje probrané metody (objekty) formou referenční příručky. Je zde kromě popisu uveden i způsob použití uvedených metod.		
Autor	Ivan Pomykacz	**Jazyk**	čeština
Očekávaný výstup	Pracuje s referenční příručkou. Vyhledává informace v oficiální příručce jazyka.		
Klíčová slova	reference jazyka, metody, příklady použití		
Druh výukového zdroje	Výklad	**Věková skupina žáků**	15+
Typ interakce	aktivita	**Ročník**	1.
Speciální vzdělávací potřeby	žádné		
Zhotoveno, (datum/období)	13.09.2013	**Celková velikost**	

Obsah

- Shrnutí
- Přehled
  - `System.IO.File.AppendAllText()`
  - `System.Math.Ceiling()`
  - `System.IO.File.Exists()`
  - `System.Math.Log()`
  - `System.DateTime.Now`
  - `double.Parse()`
  - `int.Parse()`
  - `System.IO.File.ReadAllLines()`
  - `System.IO.File.ReadAllText()`
  - `System.Console.ReadLine()`
  - `string.Substring()`
  - `string.ToLower()`
  - `System.DateTime.Today`
  - `string.Trim()`
  - `System.Console.Write()`
  - `System.IO.File.WriteAllText()`
  - `System.Console.WriteLine()`

- Cvičení

Shrnutí

Přehled použitých metod ve cvičeních z Programování. Metody jsou psány abecedně podle názvu a následně podle jmenného prostoru (namespace). U každé metody je krátká ukázka jejího použití.

Pokud jsou v ukázkách použity tři tečky (...), znamená to, že kód na sebe nenavazuje a zpravidla se obě části kódu píší na jiných místech, než za sebou. Nicméně je nezbytné, aby byly uvedeny obě části "někde" v kódu.

Toto je pouze chabá náhražka referenční příručky jazyka C#. Rozhodně je doporučeno používat odkazovanou referenční příručku.

Přehled

Metoda	Objekt	Namespace	Použití

`System.IO.File.AppendAllText()`

AppendAllText()	File	*System.IO*	
Přidá nový obsah na konec souboru.			
	return	`string`	

`System.Math.Ceiling()`

Ceiling()	Math	*System*	
Provede zaokrouhlení desetinného čísla směrem nahoru.			
	return	`double`	

`System.IO.File.Exists()`

Exists()	File	*System.IO*	
Testuje existenci souboru.			
	return	`bool`	

`System.Math.Log()`

Log()	Math	*System*	
Spočítá přirozený logaritmus nebo logaritmus daného základu.			
	return	`double`	

`System.DateTime.Now`

Now	DateTime	*System*	
Vrací aktuální čas (včetně data)			
	return	`DateTime`	

`double.Parse()`

Parse()	double	*builtin*	
Parsuje vstupní argument, kterým je string, a pokud to lze, převede hodnotu na číselný datový typ double.			
	return	`double`	

```
double cislo;
double = double.Parse("3,14");

string text = "2,87";
cislo = double.Parse(text);
```

`int.Parse()`

Parse()	int	*builtin*	
Parsuje vstupní argument, kterým je string, a pokud to lze, převede hodnotu na číselný datový typ int.			
	return	`int`	

```
int cislo;
cislo = int.Parse("42");

string text = "158";
cislo = int.Parse(text);
```

`System.IO.File.ReadAllLines()`

ReadAllLines()	File	*System.IO*	
Program načte všechny řádky z textového souboru a uloží do pole řetězců.			
	return	`string[]`	

`System.IO.File.ReadAllText()`

ReadAllText()	File	*System.IO*	
Načte obsah souboru do paměti.			
	return	`string`	

`System.Console.ReadLine()`

ReadLine()	Console	*System*	
Umožní zadat uživateli text z klávesnice a vrátit jej jako string. Program se pozastaví - čeká na stisk klávesy Enter.			
	return	`string`	

```
using System;
//...
string jmeno;
Console.Write("Zadej jméno: ");
jmeno = Console.ReadLine();
```

`string.Substring()`

Substring()	string	*builtin*	
Metoda vrací část řetězce.			
	return	`string`	

`string.ToLower()`

ToLower()	string	*builtin*
Převede všechna velká písmena v řetězci na malá.		
	return	`string`

`System.DateTime.Today`

Today	DateTime	*System*
Vrací aktuální datum (nikoli čas)		
	return	`DateTime`

`string.Trim()`

Trim()	string	*builtin*
Zbaví okraje řetězece bílých znaků (mezery, řádkové zlomy, tabulátory, ...)		
	return	`string`

`System.Console.Write()`

Write()	Console	*System*
Vypíše parametr metody na obrazovku (konzoli) bez řádkového zlomu.		
	return	`void`

```
using System;
//...
Console.Write("Ahoj světe!");
string jmeno = "John";
Console.Write("Hello " + jmeno + "!");
```

`System.IO.File.WriteAllText()`

WriteAllText()	File	*System.IO*
Uloží - přepíše soubor novým obsahem.		
	return	`string`

`System.Console.WriteLine()`

WriteLine()	Console	*System*
Vypíše parametr metody na obrazovku (konzoli) s řádkovým zlomem.		
	return	`void`

```
using System;
//...
Console.WriteLine("Ahoj světe!");
string jmeno = "John";
Console.WriteLine("Hello " + jmeno + "!");
```