



MINISTERSTVO ŠKOLSTVÍ,  
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání  
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

# Výpočetní technika programování

## Třídy a metody

Tento materiál vznikl v rámci projektu CZ.1.07/1.2.15/01.0022  
„Záchytná síť– široké spektrum podpůrných aktivit“

Střední škola spojů a informatiky, Tábor, Bydlišského 2474, 390 11 Tábor  
[www.sous.cz](http://www.sous.cz)

© 2014 Ing. Almášiová Dana

## TVORBA METOD VE VLASTNÍ TŘÍDĚ

Objekty naučíme provádět požadované činnosti – definujeme metody v dané třídě („podprogramy“). Syntaxe definice metody:

```
public typ_návratové_hodnoty název_metody( seznam_parametrů)
{
    příkazy_metody
}
```

Takže si nejprve musíme ujasnit, jak se bude metoda jmenovat, jaké bude potřebovat vstupní parametry a jaké typy výstupních hodnot bude vracet. Když jde o metodu bez návratové hodnoty, jako typ se píše klíčové slovo *void*. Tělo metody obsahuje příkazy, které se mají postupně provést. Pokud nepotřebujeme ani žádný vstupní parametr, necháme závorky prázdné. Metoda může pracovat s vlastnostmi objektu pro nějž byla volána (spuštěna – viz. dále v programu), pak se nepoužívá tečková notace, protože metoda používá vlastnosti daného objektu.

### METODY S NÁVRATOVOU HODNOTOU

Tato metoda je charakterizována nutností předat získaný výsledek do místa volání. V jejím těle tak musí být řádek s klíčovým slovem *return*, který definuje návratovou hodnotu metody.

### METODY S PARAMETRY

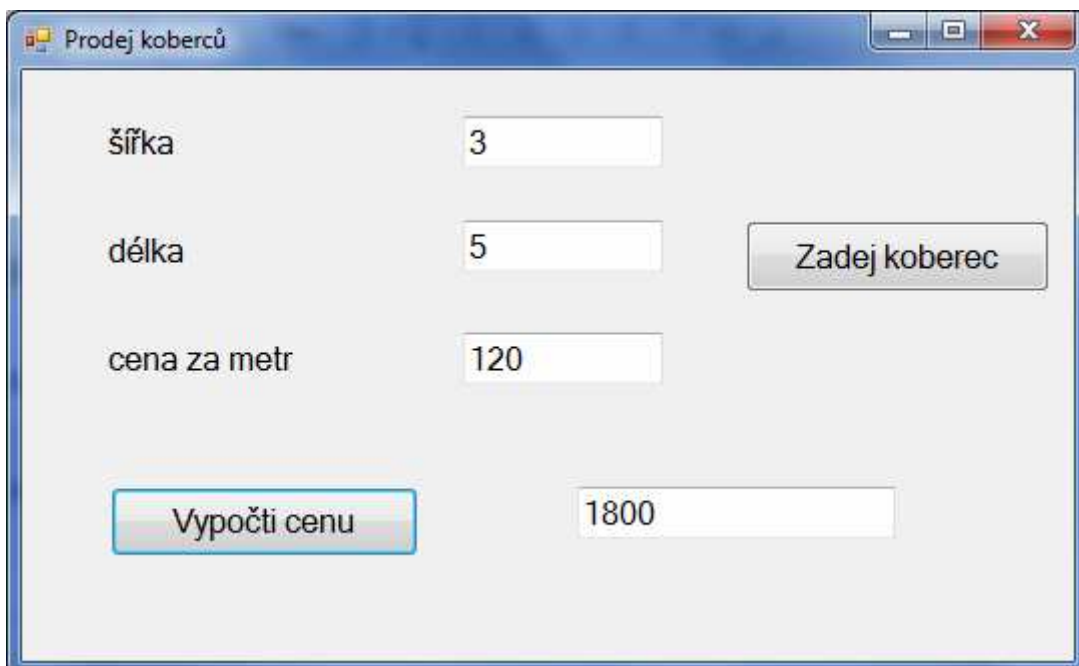
V definici metody vypadá parametr jako proměnná, ve které může být jakákoliv hodnota, zadaná až před zavoláním metody. Těmto parametrům se říká parametry formální, parametry použité v místě volání metody jsou parametry skutečné.

## ŘEŠENÝ PŘÍKLAD

### ZADÁNÍ:

Vytvořte třídu Koberec. Zadejte rozměry koberce a cenu 1m<sup>2</sup> – vlastnosti objektu třídy Koberec. Vypočítejte cenu celého koberce pomocí metody této třídy.

### NÁVRH PROJEKTU:



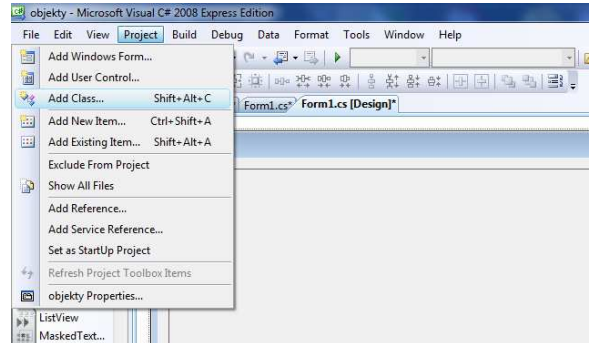
šířka	3	
délka	5	Zadej koberec
cena za metr	120	
Vypočti cenu	1800	

## KÓD ŘEŠENÍ:

Nejprve přidáme vlastní třídu výběrem přes nabídky Project – Add class. Třídě nastavíme vlastnosti s veřejným přístupem – public. Vytvoříme metodu pro výpočet ceny zadaného koberce jako její návratové hodnoty.

```
namespace TřídaKoberec
{
    class Koberec
    {
        public double šířka;
        public double délka;
        public double cena;

        public double cenaCelkem()
        {
            double celkem = šířka * délka * cena;
            return celkem;
        }
    }
}
```



Dále pokračujeme s tvorbou zdrojového kódu pro formulář.

```
namespace TřídaKoberec
{
    public partial class Form1 : Form
    {
        Koberec skutečný = new Koberec();
        public Form1()
        {
            InitializeComponent();
        }

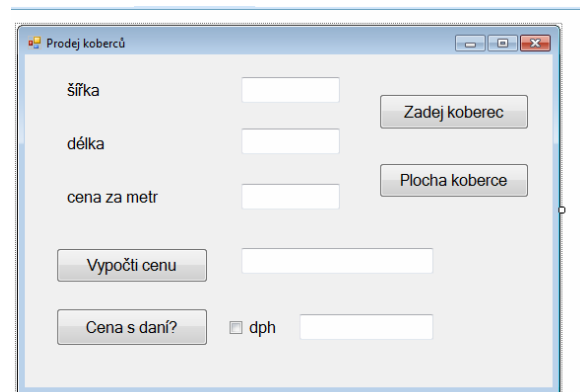
        private void button1_Click(object sender, EventArgs e)
        {
            // zadání vlastností jednoho objektu
            skutečný.šířka = Convert.ToDouble(poleSirka.Text);
            skutečný.délka = Convert.ToDouble(poleDelka.Text);
            skutečný.cena = Convert.ToDouble(poleCena.Text);
        }

        private void button2_Click(object sender, EventArgs e)
        {
            // volání metody a výpis jejího výsledku
            poleCelkem.Text = Convert.ToString(skutečný.cenaCelkem());
        }
    }
}
```

## PROCVÍČENÍ

Navrhněte další metody pro tuto třídu:

```
// jednoduchá metoda, bez vstupních a
// výstupních hodnot pro výpočet plochy
public void VypočtiPlochu()
{
    plocha = šířka * délka ;
}
```



Tlačítkem Plocha koberce spustíme tuto metodu a výslednou hodnotu zobrazíme v novém okně.

```
private void button4_Click(object sender, EventArgs e)
{
    //spustíme metodu
    skutečný.VypočtiPlochu();
    //zobrazíme v okně vypočtenou plochu
    MessageBox.Show("Plocha koberce je " + skutečný.plocha.ToString());
}
```

Další metoda vypočte cenu koberce i s DPH v případě, že bude zaškrtnuto tlačítko. Do metody budou předány dva vstupní parametry: logický – *dph* (zda počítat i s daní) a číselná hodnota daně – *dan*.

// metoda s vstupními a výstupní hodnotou, zápis vstupních parametrů je stejný jako deklaráce proměnné

```
public double cenaDPH(bool dph, double dan)
{
    // výpočet
    double celkem = šířka * délka * cena;
    // pokud má být DPH, zvýšíme o ni cenu
    if (dph)
        celkem += celkem * (dan / 100);
    //návrat vypočtené hodnoty
    return celkem;
}
```

Při volání metody s parametry je musíme zadat ve stejném pořadí, jako jsou uvedené v definici této metody ve třídě. Buď můžeme zadat přímo hodnoty nebo proměnné, které tyto hodnoty obsahují.

```
private void button3_Click(object sender, EventArgs e)
{
    // zjistíme zadanou daň
    double zadanaDan = Convert.ToDouble(poleDan.Text);
    // spustíme metodu a předáme v ní, zda chceme výpočet s daní a její hodnotu
    poleCelkem.Text =Convert.ToString(skutečný.cenaDPH(checkBox1.Checked, zadanaDan));
}
```

## TYPOVÉ PŘÍKLADY:

1. Do projektu s auty přidejte metodu NaStart, která nastaví auto na pravý kraj formuláře. Stisknutí klávesy N ji spustí pro všechna tři auta.
2. Připravte projekt se třídou Náklad, s vlastnostmi: Hmotnost nákladu, Cena dopravy 1t, Délka trasy a cena za 1km. Přidejte ještě zaškrťovací tlačítko pro volbu slevy stálého zákazníka (10%). Budete zadávat hodnoty pro objekt jednoho nákladu a počítat cenu jeho dopravy se slevou nebo bez slevy.