Devinettes en C#

par Jon Skeet (Page d'accueil) (Blog) Jean-Michel Ormes (traduction) (Page d'accueil) (Blog)

Date de publication : 23/03/2012

Dernière mise à jour :

Régulièrement, je tombe sur une situation intéressante en C# qui donne des résultats surprenants. Cette page contient une liste d'exemples. Dans les exemples où il n'y à qu'un bout de code, nous supposerons que celui-ci est dans la méthode Main. Afin de ne pas tomber accidentellement sur les résultats avant que vous ne le souhaitiez, j'ai mis les **réponses sur une autre page**.

Traduction....3
1 - Surcharge....3
2 - Ordre ! Ordre !....3
3 - Bête arithmétique....4
4 - Print, print, print....4
5 - En apparence, tout semble aller avec le compilateur ici....4
6 - Inférence type à gogo....5
Remerciements....5

Copyright ® 2012 Jon Skeet. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts. Cette page est déposée à la SACD.

Traduction

Ceci est la traduction la plus fidèle possible de l'article de Jon Skeet, **Brainteasers C#**.

1 - Surcharge

Qu'est-ce qui est affiché, et pourquoi?

```
using System;

class Base
{
    public virtual void Foo(int x)
    {
        Console.WriteLine ("Base.Foo(int)");
    }
}

class Derived : Base
{
    public override void Foo(int x)
    {
        Console.WriteLine ("Derived.Foo(int)");
    }

    public void Foo(object o)
    {
        Console.WriteLine ("Derived.Foo(object)");
    }
}

class Test
{
    static void Main()
    {
        Derived d = new Derived();
        int i = 10;
        d.Foo(i);
    }
}
```

2 - Ordre ! Ordre !

Qu'est-ce qui va s'afficher, pourquoi, et en êtes-vous sûr ?

```
using System;

class Foo
{
    static Foo()
    {
        Console.WriteLine ("Foo");
    }
}

class Bar
{
    static int i = Init();

    static int Init()
    {
        Console.WriteLine("Bar");
```


Copyright ® 2012 Jon Skeet. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts. Cette page est déposée à la SACD.

```
        return 0;
    }
}

class Test
{
    static void Main()
    {
        Foo f = new Foo();
        Bar b = new Bar();
    }
}
```

3 - Bête arithmétique

Les ordinateurs sont censés être bons en calcul, n'est-ce pas ? Alors pourquoi la console renvoie "False" ?

```
double d1 = 1.000001;
double d2 = 0.000001;
Console.WriteLine((d1-d2)==1.0);
```

4 - Print, print, print....

Voici un code utilisant les fonctionnalités de méthode anonyme de C # 2. Que fait-il?

```
using System;
using System.Collections.Generic;

class Test
{
    delegate void Printer();

    static void Main()
    {
        List<Printer> printers = new List<Printer>();
        for (int i=0; i < 10; i++)
        {
            printers.Add(delegate { Console.WriteLine(i); });
        }

        foreach (Printer printer in printers)
        {
            printer();
        }
    }
}
```

5 - En apparence, tout semble aller avec le compilateur ici ...

Est-ce que ce code pourrait compiler ? Compile t'il ? Qu'est-ce que cela signifie?

```
using System;

class Test
{
    enum Foo { Bar, Baz };

    static void Main()
    {
        Foo f = 0.0;
        Console.WriteLine(f);
```


Copyright ® 2012 Jon Skeet. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts. Cette page est déposée à la SACD.

```
    }
}
```

En voici plus avec les mêmes lignes....

```
using System;

class Test
{
    enum Foo { Bar, Baz };

    const int One = 1;
    const int Une = 1;

    static void Main()
    {
        Foo f = One-Une;
        Console.WriteLine(f);
    }
}
```

6 - Inférence type à gogo

J'ai d'abord vu ceci sur le blog d'Ayende (en une forme un peu plus obscure, il est vrai). Encore une fois, réfléchissez sur ce que le code va afficher et pourquoi.

```
using System;

class Test
{
    static void Main()
    {
        Foo("Hello");
    }

    static void Foo(object x)
    {
        Console.WriteLine("object");
    }

    static void Foo<T>(params T[] x)
    {
        Console.WriteLine("params T[]");
    }
}
```

Remerciements

Je tiens à remercier Jon Skeet pour son aimable autorisation de traduire cet article, ainsi que .... pour sa relecture attentive et ses corrections.

Copyright ® 2012 Jon Skeet. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts. Cette page est déposée à la SACD.