借助运行时类型标识（RTTI），可以在程序执行期间判定对象的类型。在C#中支持RTTI的关键字为：is、as和typeof。

is运算符：

expr is type

如果expr的类型与type类型相同或兼容（派生，装箱或拆箱转换），结果为true，否则为false。

例：程序rtti\_test1

// Copyright 2016.刘珅珅

// author：刘珅珅

// 运行时类型标识-is

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace rtti\_test1

{

class A { }

class B : A { }

class RTTITest

{

static void Main(string[] args)

{

A a = new A();

B b = new B();

if (a is A)

{

Console.WriteLine("a is an A");

}

if (b is A)

{

Console.WriteLine("b is an A because it is derived from A");

}

if (a is B)

{

Console.WriteLine("a is B");

}

else

{

Console.WriteLine("a is not B");

}

if (a is object)

{

Console.WriteLine("a is an object");

}

}

}

}

输出结果为：

a is an A

b is an A because it is derived from A

a is not B

a is an object

as运算符：如果希望在程序运行时执行转换，并且即使转换失败也不抛出异常（使用强类型转换，如果失败就会抛异常），可以使用as运算符。

expr as type

如果成功，返回type类型引用，否则返回一个空引用。

例：程序rtti\_test2

// Copyright 2016.刘珅珅

// author：刘珅珅

// RTTI运行时类型标识符：as运算符

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace rtti\_test2

{

class A { }

class B : A { }

class RTTITest

{

static void Main(string[] args)

{

A a = new A();

B b = new B();

b = a as B;

if (b == null)

{

Console.WriteLine("The cast in b = (B) a is NOT allowed.");

}

else

{

Console.WriteLine("The cast in b = (B) a is allowed.");

}

}

}

}

输出结果：

The cast in b = (B) a is NOT allowed.

typeof运算符：typeof可以获取某个类型的具体信息。

例：程序rtti\_test3

// Copyright 2016.刘珅珅

// author：刘珅珅

// RTTI运行时类型标识：typeof运算符

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Text;

using System.Threading.Tasks;

namespace rtti\_test3

{

class RTTITest

{

static void Main(string[] args)

{

Type t = typeof(StreamReader);

Console.WriteLine(t.FullName);

if (t.IsClass)

{

Console.WriteLine("Is a class.");

}

if (t.IsAbstract)

{

Console.WriteLine("Is abstract.");

}

else

{

Console.WriteLine("Is concrete.");

}

}

}

}

输出结果：

System.IO.StreamReader

Is a class.

Is concrete.