**[JSON 序列化和反序列化——JavaScriptSerializer实现](http://www.cnblogs.com/jams742003/archive/2009/12/24/1631485.html)**

通过JavaScriptSerializer来实现。它的名字空间为：System.Web.Script.Serialization

如果要使用它，还须添加

System.Web.Extensions库文件引用

参考实体类：Customer

public class Customer  
{  
   public int Unid { get; set; }  
   public string CustomerName { get; set; }  
}

类JavaScriptSerializer描述：为启用 AFAX 的应用程序提供序列化和反序列化功能。

（一） 序列化

方法：public string Serialize(Object obj)，用于将对象转换为 JSON 字符串

public string ScriptSerialize(Customer customer)  
{  
    JavaScriptSerializer js = new JavaScriptSerializer();  
  
    return js.Serialize(customer);  
}

测试

Customer cc = new Customer { Unid = 1, CustomerName = "John" };  
string strJson = ScriptSerialize(cc);  
  
Console.WriteLine(strJson);

（二）反序列化

public Customer ScriptDeserialize(string strJson)  
{  
    JavaScriptSerializer js = new JavaScriptSerializer();  
    return js.Deserialize<Customer>(strJson);  
  
}

通过Deserialize<T>方法来实现。

测试：

Customer c1 = ScriptDeserialize(strJson);  
Console.WriteLine(c1.Unid + " " + c1.CustomerName);

（三）方法泛型

[复制代码](javascript:void(0);)

public string ScriptSerialize<T>(T t)  
{  
    JavaScriptSerializer js = new JavaScriptSerializer();  
    return js.Serialize(t);  
}  
  
public T ScriptDeserialize<T>(string strJson)  
{  
    JavaScriptSerializer js = new JavaScriptSerializer();  
    return js.Deserialize<T>(strJson);  
}

[复制代码](javascript:void(0);)

测试：

[复制代码](javascript:void(0);)

Customer cc = new Customer { Unid = 1, CustomerName = "John" };  
string strJson = ScriptSerialize<Customer>(cc);  
Console.WriteLine(strJson);   
  
Customer c1 = ScriptDeserialize<Customer>(strJson);  
Console.WriteLine(c1.Unid + " " + c1.CustomerName);

[复制代码](javascript:void(0);)