1.序列化

string s = JsonConvert.SerializeObject(model);

2.反序列化

string s = "{"code":204,"msg":"没有符合条件的数据！","resData":null}";

RetErr ret = JsonConvert.DeserializeObject<RetErr>(s);

Console.WriteLine("{0},{1}", ret.ABC, ret.msg);

3.忽略某些属性

OptOut：默认值,类中所有公有成员会被序列化,如果不想被序列化,可以用特性JsonIgnore

OptIn：默认情况下,所有的成员不会被序列化,类中的成员只有标有特性JsonProperty的才会被序列化,当类的成员很多,但客户端仅仅需要一部分数据时,很有用

用法：

[JsonObject(MemberSerialization.OptIn)]

[JsonObject(MemberSerialization.OptOut)]

4.自定义序列化的字段名称

[JsonProperty(propertyName: "resourceUuid")]

public string sourceIds { get; set; }

1. 驼峰处理

JsonSerializerSettings settings = new JsonSerializerSettings();

settings.ContractResolver = new CamelCasePropertyNamesContractResolver();

settings.Formatting = Formatting.Indented;

JsonConvert.SerializeObject(result, settings);