

Primeira série de exercícios

Objectivos: Prática com a API de reflexão

Data limite de entrega: 3 de Abril de 2016

NOTA 1: resolva a série de exercícios completando a base de código disponível [neste repositório Github](#)

NOTA 2: Adicione os testes unitários necessários para validar o correcto funcionamento das funcionalidades pedidas.

JSON (*JavaScript Object Notation*) é um formato de dados textual para representação de objectos. O JSON é independente da linguagem de programação e a sua notação usa convenções familiares ao idioma C, C#, Java ou Javascript. Baseia-se num subconjunto da linguagem de programação JavaScript, Standard ECMA-262.

Consulte as regras gramaticais do formato JSON em <http://www.json.org>.

Nesta série de exercícios pretende-se implementar a classe `Jsonzai.Reflect.Jsonfier` que disponibiliza um método `ToJson` que retorna uma representação JSON para o objecto recebido por parâmetro, conforme o exemplo seguinte:

```
using Jsonzai.Reflect;

Student expected = new Student(27721, "Ze Manel");
string json = Jsonfier.ToJson(expected);
Console.WriteLine(json); // -> {"nr": 27721, "name": "Ze Manel"}

int [] expected = { 4, 5, 6, 7};
json = Jsonfier.ToJson(expected);
Console.WriteLine(json); // -> [4,5,6,7]
```

O método `ToJson` é recursivo na serialização de membros de objectos e de elementos de *arrays*, tendo como condição terminal o argumento ser `null`, `string` ou um tipo primitivo. Considere a possibilidade de serializar como *array* os objectos que tenham um indexador.

Faça uma implementação que seja flexível na adaptação do **critério de selecção** dos membros. Ou seja, deve existir a possibilidade de modificar qual a estratégia de selecção dos membros a usar na representação JSON. Por exemplo, na representação JSON de objectos podem ser considerados os valores dos campos do objecto, ou das suas propriedades, ou dos métodos sem parâmetros e retorno *non void*, ou outro critério qualquer. O objectivo é que a modificação do **critério de selecção** seja feita com o mínimo de alterações ao código.

Deve implementar testes unitários que validem todas as situações e os diferentes critérios de selecção implementados. No mínimo tem que implementar uma estratégia de selecção de campos e outra só de propriedades.

Na base de código disponibilizada em <https://github.com/isel-leic-ave/jsonzai> são dados a título exemplificativo 3 testes unitários. Nestes testes é usada a biblioteca `Newtonsoft.Json` apenas para reconstruir o objecto a partir de uma representação JSON e validar se é igual ao objecto original.