

蜕变关系

该文档主要记录识别的蜕变关系。为了下文方便描述做以下定义：

SJB: 原始测试用例的JavaBean/POJO;

FJB: 衍生测试用例的JavaBean/POJO;

SF: 原始测试用例的JSON文件;

FF: 衍生测试用例的JSON文件;

SO: 原始测试用例反序列化后的对象;

FO: 衍生测试用例反序列化后的对象;

蜕变关系1

FF通过在SF中每一个键对应值的末尾添加一行注释得到：SJB与FJB完全一致；如果存在一个SO与FO中名字一样的成员变量具有不同的值，该蜕变关系被违反。

SF文件内容：

```
{
  "1": {
    "airClass": 2,
    "area": 0,
    "economicfee": 5153.698093040778,
    "luggage": 22.281053775603675,
    "student": false
  },
  "2": {
    "airClass": 3,
    "area": 0,
    "economicfee": 1936.2340923480338,
    "luggage": 48.14329753468175,
    "student": false
  }
}
```

FF文件内容：

```
{
  "1": {
    "airClass": 2,/*这是注释*/
    "area": 0,/*这是注释*/
    "economicfee": 5153.698093040778,/*这是注释*/
    "luggage": 22.281053775603675,/*这是注释*/
    "student": false/*这是注释*/
  },/*这是注释*/
  "2": {
    "airClass": 3,/*这是注释*/
    "area": 0,/*这是注释*/
    "economicfee": 1936.2340923480338,/*这是注释*/
  }
}
```

```
"luggage": 48.14329753468175,/*这是注释*/
"student": false/*这是注释*/
}/*这是注释*/
}
```

蜕变关系2

FF是将SF转化为字节后得到的字节数组；SJB与FJB完全一致；如果存在一个SO与FO同名的成员变量具有不同的值，该蜕变关系被违反。

蜕变关系3

FF是将SF转化为字符串后的结果；SJB与FJB完全一致；如果存在一个SO与FO同名的成员变量具有不同的值，该蜕变关系被违反。

SF文件内容：

```
{
  "1": {
    "airClass": 2,
    "area": 0,
    "economicfee": 5153.698093040778,
    "luggage": 22.281053775603675,
    "student": false
  }
}
```

FF文件内容

```
\ "1\": {\"airClass\": \"2\", \"area\": \"0\",
\"economicfee\": \"5153.698093040778\", \"luggage\":
\"22.281053775603675\", \"student\": \"false\"}
```

蜕变关系4

如果SF中float类型的变量值含有少于7个有效数字。FF在该变量值的末尾随机添加7位有效数字；SJB与FJB完全一致；如果SO与FO中该变量值的前7位数字完全相同且大小与SF中变量值相等，该蜕变关系没有被违反。

SF文件内容：

```
{
  "1": {
    "airClass": 2,
    "area": 0,
    "economicfee": 5153.6,/*假设为float类型*/
    "luggage": 22.2,/*假设为float类型*/
    "student": false
  }
}
```

FF文件内容

```

{
    "1": {
        "airClass": 2,
        "area": 0,
        "economicfee": 5153.61234567,/*假设为float类型*/
        "luggage": 22.21234567,/*假设为float类型*/
        "student": false
    }
}

```

蜕变关系5

如果SF中float类型的变量值含有多于7个有效数字。FF从该变量值的末尾开始截取若干数字，使得FF中的该变量具有7位有效数字；SJB与FJB完全一致；如果SO与FO中该变量值的前7位数字完全相同且大小与FF中变量值相等，该蜕变关系没有被违反。

```

SF文件内容:
{
    "1": {
        "airClass": 2,
        "area": 0,
        "economicfee": 5153.61234567,/*假设为float类型*/
        "luggage": 22.21234567,/*假设为float类型*/
        "student": false
    }
}

FF文件内容

{
    "1": {
        "airClass": 2,
        "area": 0,
        "economicfee": 5153.612,/*假设为float类型*/
        "luggage": 22.21234,/*假设为float类型*/
        "student": false
    }
}

```

蜕变关系6

SJB中存在一个成员变量为枚举类型，SF中该变量的值是任意一个枚举对象中的元素。FF将枚举类型的成员变量的值替换为不是枚举对象元素的值；SJB与FJB完全一致；如果FO中枚举类型的成员变量的值不为NULL，或其它成员变量的值与SO不相同，该蜕变关系被违反。

```

SJB文件:
enum Code {
    SUCCESS, FAILURE
}

```

```

}

public class Num {
    private int id;
    private Code code;
    //getter and setter
}

```

SF文件内容:

```

{
    "code":SUCCESS
    "id":1
}

```

FF文件内容:

```

{
    "code":ERROR
    "id":1
}

```

蜕变关系7

FF将SF中表示日期的值用统一的其它格式的形式表示，其它内容完全相同；SJB与FJB完全一致；如果SO除了日期成员变量，存在一个成员变量的值与FO不同或者FO中表示日期的成员变量值的形式与FF中不同又或者O2表示时间的成员变量的值与O1不同，该蜕变关系被违反。

蜕变关系8

FF将SF中的所有驼峰形式的键用蛇形表示，不改变键的值；SJB与FJB完全一致；如果在得到FO的过程中报错，或者FO与SO相同成员变量的值不同，该蜕变关系被违反。

```

/*SF文件内容*/
{
    "1": {
        "airClass": 2,
        "area": 0,
        "economicfee": 5153.61234567,/*假设为float类型*/
        "luggage": 22.21234567,/*假设为float类型*/
        "student": false
    }
}

/*FF文件内容*/
{
    "1": {
        "air_class": 2,
        "area": 0,
        "economic_fee": 5153.61234567,/*假设为float类型*/

```

```
    "luggage": 22.21234567, /*假设为float类型*/
    "student": false
  }
}
```

蜕变关系9

FF将SF中的所有驼峰形式的键用大驼峰表示，不改变键的值；SJB与FJB完全一致；如果在得到FO的过程中报错，或者FO与SO相同成员变量的值不同，该蜕变关系被违反。

```
/*SF文件内容*/
{
  "1": {
    "airClass": 2,
    "area": 0,
    "economicfee": 5153.61234567, /*假设为float类型*/
    "luggage": 22.21234567, /*假设为float类型*/
    "student": false
  }
}

/*FF文件内容*/
{
  "1": {
    "AirClass": 2,
    "area": 0,
    "EconomicFee": 5153.61234567, /*假设为float类型*/
    "luggage": 22.21234567, /*假设为float类型*/
    "student": false
  }
}
```