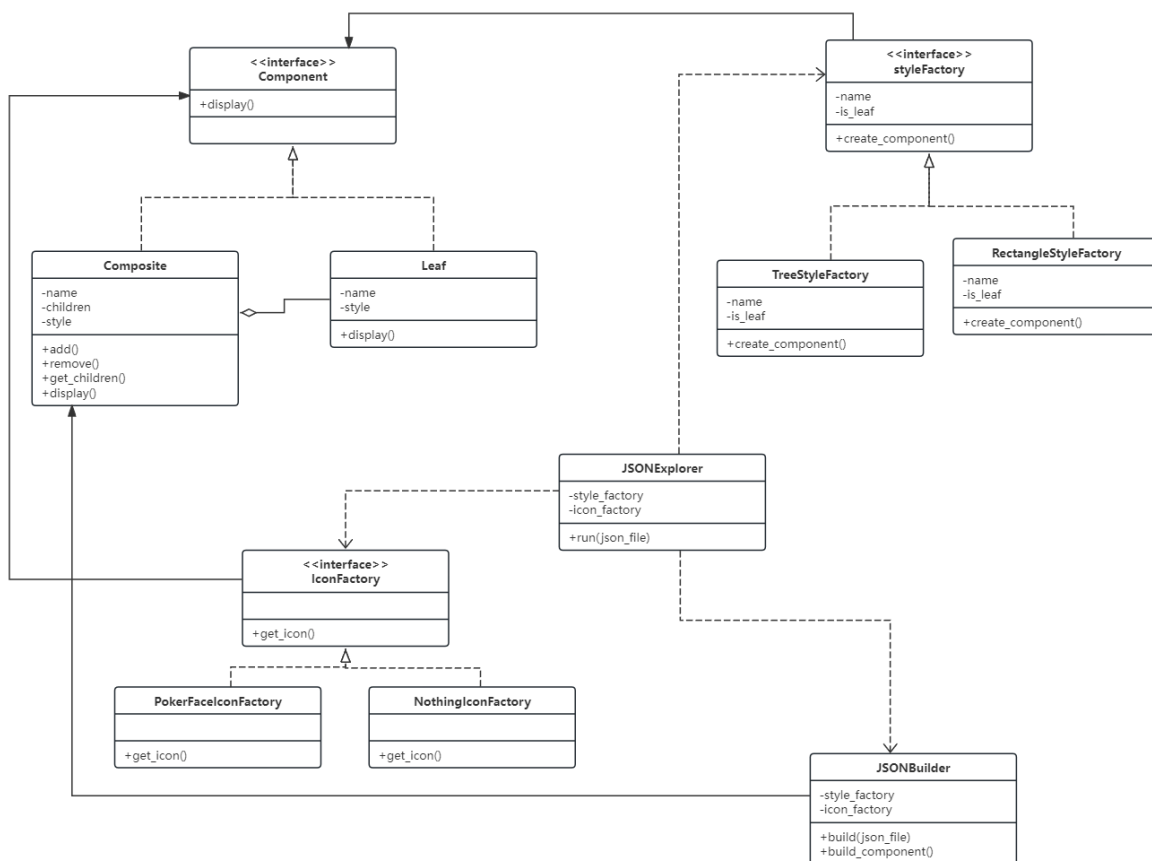


Funny JSON Explorer 设计文档

概述

Funny JSON Explorer 是一个命令行工具，用于可视化 JSON 文件。该工具支持树形 (tree) 和矩形 (rectangle) 两种风格，以及不同的图标族 (如扑克图标族)。通过使用工厂方法、抽象工厂、建造者和组合模式，本项目实现了高扩展性和灵活性。

类图



类之间的关系

- Component 接口:**
 - 被 **Composite** 和 **Leaf** 实现。
- Composite 类:**
 - 继承 **Component**。
 - 包含 **Component** 类型的子组件。
- Leaf 类:**
 - 继承 **Component**。
- StyleFactory 抽象类:**
 - 被 **TreeStyleFactory** 和 **RectangleStyleFactory** 实现。
- TreeStyleFactory 类:**
 - 继承 **StyleFactory**。
- RectangleStyleFactory 类:**

- 继承 `StyleFactory`。

7. `IconFactory` 抽象类:

- 被 `PokerFaceIconFactory` 实现。

8. `PokerFaceIconFactory` 类:

- 继承 `IconFactory`。

9. `JSONBuilder` 类:

- 依赖 `StyleFactory` 和 `IconFactory`。
- 使用 `StyleFactory` 和 `IconFactory` 来构建 `Composite` 和 `Leaf` 对象。

10. `JSONExplorer` 类:

- 依赖 `StyleFactory` 和 `IconFactory`。
- 使用 `JSONBuilder` 来构建并展示 JSON 组件树。

设计模式

工厂方法模式

工厂方法模式通过定义创建对象的接口，让子类决定实例化哪个类。这里，`StyleFactory` 定义了创建组件的方法，不同的风格（树形和矩形）有不同的具体工厂实现。

抽象工厂模式

抽象工厂模式提供一个创建一系列相关或相互依赖对象的接口，而无需指定它们具体的类。

`IconFactory` 提供了获取图标的方法，不同的图标族（如扑克图标族）有不同的具体工厂实现。

建造者模式

建造者模式通过将复杂对象的构建过程封装起来，使得构建过程更加灵活和可控。`JSONBuilder` 类负责构建 JSON 组件树，使用 `StyleFactory` 和 `IconFactory` 来创建各个组件。

组合模式

组合模式允许你将对象组合成树形结构来表示部分-整体层次结构。`Component` 接口被 `Composite` 和 `Leaf` 实现，使得可以将叶子节点和复合节点统一处理。

迭代器模式

迭代器模式能够不暴露集合底层表现形式而访问集合所有元素。使用 `TreeIterator` 类创建迭代器，由于本项目使用DFS遍历树的节点，因此使用子类 `DFSIterator` 作为迭代器，其中使用栈来存储先前节点数据，从而实现调用 `getNext()` 函数即可得到下一个元素。且有其他迭代需求使用相同接口创建新的迭代器即可。

访问者模式

创建 `Viewer` 类来将输出与节点数据和访问算法隔离开来。使用 `Viewer` 类内 `display` 即可访问节点数据，而不需要对节点添加代码。

代码实现

见代码目录文件。

运行说明

1. 确保已安装 Python 3.0以上版本。
2. 将上述代码文件保存到同一目录下。
3. 运行命令：

```
python ./main -f [filename] -s [style] -i [icon]
```

4. 该命令将读取 `sample.json` 文件，使用树形风格和扑克图标族展示 JSON 结构。

这样，`Funny JSON Explorer` 工具就能根据指定的风格和图标族可视化 JSON 文件内容，展示出清晰的层次结构。

想要自定义icon，可在`config.json`中修改`leaf`与`node`的value，随后将`icon`值改为`config`即可，效果如下：

```
{
  "leaf": "2",
  "node": "1"
}
```

```
PS C:\Users\92305\Desktop\软件工程\Funny-JSON-Explorer> python JSONExplorer.py sample.json tree config
├─1oranges
│   └─1mandarin
│       └─2clementine
│           └─2tangerine: cheap & juicy!
├─1apples
│   └─2gala
│       └─2pink lady
```

操作截图

两种不同风格 `style rectangle` 和两种不同图标 `poker nothing`组合成的四种结果如下：

```
PS C:\Users\92305\Desktop\软件工程\Funny-JSON-Explorer> python JSONExplorer.py sample.json rectangle nothing
├─ oranges
│   └─ mandarin
│       └─ clementine
│           └─ tangerine: cheap & juicy!
├─ apples
│   └─ gala
│       └─ pink lady
```

```
PS C:\Users\92305\Desktop\软件工程\Funny-JSON-Explorer> python JSONExplorer.py sample.json rectangle poker
├─ ♠oranges
│   └─ ♠mandarin
│       └─ ♠clementine
│           └─ ♠tangerine: cheap & juicy!
├─ ♠apples
│   └─ ♠gala
│       └─ ♠pink lady
```

```
PS C:\Users\92305\Desktop\软件工程\Funny-JSON-Explorer> python JSONExplorer.py sample.json tree poker
├─ ♠oranges
│   └─ ♠mandarin
│       └─ ♠clementine
│           └─ ♠tangerine: cheap & juicy!
├─ ♠apples
│   └─ ♠gala
│       └─ ♠pink lady
```

```
PS C:\Users\92305\Desktop\软件工程\Funny-JSON-Explorer> python JSONExplorer.py sample.json tree nothing
├ oranges
│   ├── mandarin
│   │   ├── clementine
│   │   └── tangerine: cheap & juicy!
└ apples
    ├── gala
    └── pink lady
```