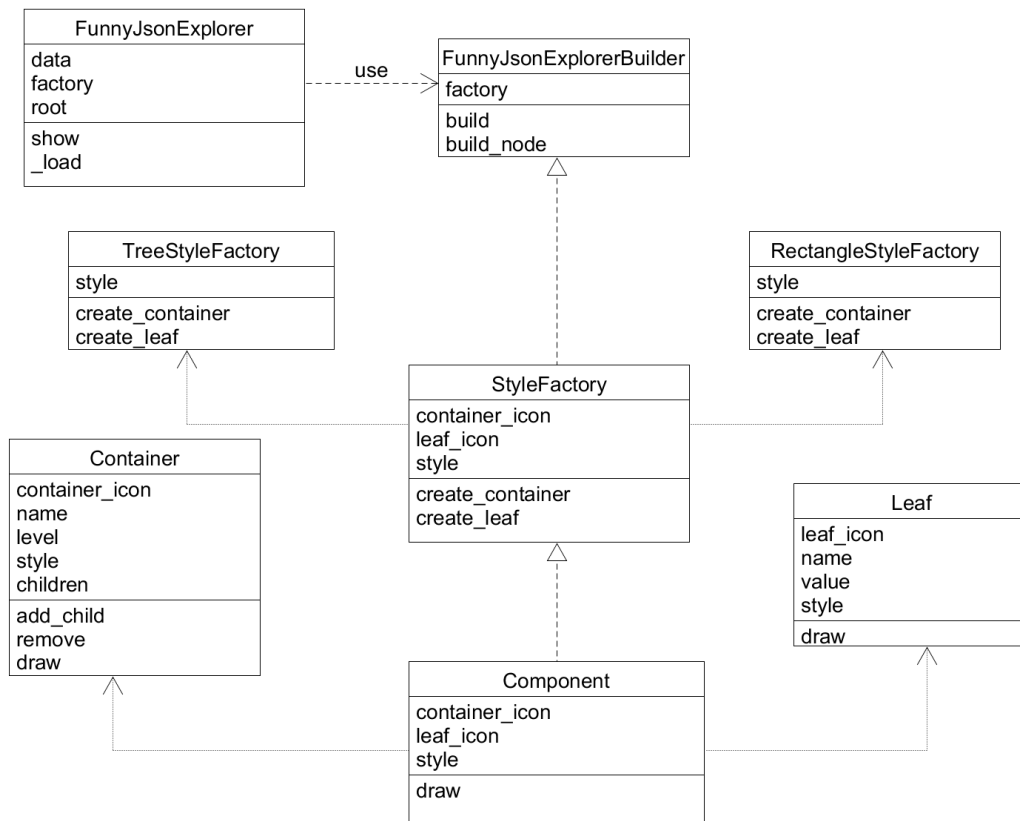


Funny JSON Explorer 设计文档

Funny JSON Explorer(FJE), 是一个 SON 文件可视化的命令行界面小工具, FJE 可以快速切换风格(style), 包括: 树形(tree)、矩形(rectangle), 也可以指定对应的图标族。下面是设计对应的类图。

Funny JSON Explorer 领域模型



说明:

FunnyJsonExplorer 主要是抽象构建节点树, 工厂是用来创建节点。

FunnyJsonExplorerBuilder 主要是具体创建节点。

StyleFactory 是抽象类, 有创建中间节点和叶节点的函数, **TreeStyleFactory** 和 **RectangleStyleFactory** 继承 **StyleFactory** 类, 分别重写创建中间节点和叶节点函数。

Component 是抽象类, 说明对应中间节点和叶节点的图标, 并且有对应的风格。**Leaf** 和 **Container** 继承 **Component** 类, 然后又对应的属性, 属性主要用于最后 `draw` 绘制节点。

只需要添加对应的抽象工厂，即可添加对应的风格，从下面代码可以看出，新建一个类可以指定对应的风格，并在 `component` 类中添加对应节点的绘制即可完成新的风格的绘制

```
class TreeStyleFactory(StyleFactory):
    def create_container(self, name, level=0):
        self.style = 'tree'
        return Container(name, level, self.style, self.container_icon)

    def create_leaf(self, name, value=None):
        self.style = 'tree'
        return Leaf(name, value, self.style, self.leaf_icon)
```

而图标是在主函数中指定的，并没有重新新建一个类来进行说明（也是可以的）。

使用设计模式及作用：

工厂方法模式用于定义一个创建对象的接口，通过这种方式，工厂方法将实例操作延迟到不同风格的子类中（树类和矩形类）

抽象工厂模式用于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。中间节点和叶子节点分别创建自己的节点。

建造者模式用于将一个复杂对象的构建过程与它的表示分离，使得同样的构建过程可以创建不同的表示。FunnyJsonExplorerBuilder 分离构建树的过程。

组合模式用于将对象组合成树形结构以表示“部分-整体”的层次结构。组合模式使得客户端对单个对象和组合对象的使用具有一致性，由总体可以看出。

使用命令

```
python main.py -f data.json -s tree -i star
```

对应的命令结果截图如下：

```
PS C:\Users\Yearn\Desktop\funny-json-explorer-py> python main.py -f data.json -s tree
├ oranges
│   └ mandarin
│       ├── clementine
│       └ tangerine: cheap & juicy!
└ apples
    ├── gala
    └ pink lady
```

```
PS C:\Users\Yearn\Desktop\funny-json-explorer-py> python main.py -f data.json -s rectangle
├ oranges ───────────────────┐
│ ├── mandarin ───────────┐   │
│ │ ├── clementine ────┐   │
│ │ └ tangerine: cheap & juicy! ────┐   │
└ apples ───────────────────┘
  ├── gala ───────────┐
  └ pink lady ─────────┘
```

```
PS C:\Users\Yearn\Desktop\funny-json-explorer-py> python main.py -f data.json -s tree -i star
├ ♣ oranges
│   └ ♣ mandarin
│       ├── ♦ clementine
│       └ ♦ tangerine: cheap & juicy!
└ ♣ apples
    ├── ♦ gala
    └ ♦ pink lady
```

```
PS C:\Users\Yearn\Desktop\funny-json-explorer-py> python main.py -f data.json -s rectangle -i star
├ ♣ oranges ───────────────────┐
│ ├── ♣ mandarin ───────────┐   │
│ │ ├── ♦ clementine ────┐   │
│ │ └ ♦ tangerine: cheap & juicy! ────┐   │
└ ♣ apples ───────────────────┘
  ├── ♦ gala ───────────┐
  └ ♦ pink lady ─────────┘
```