

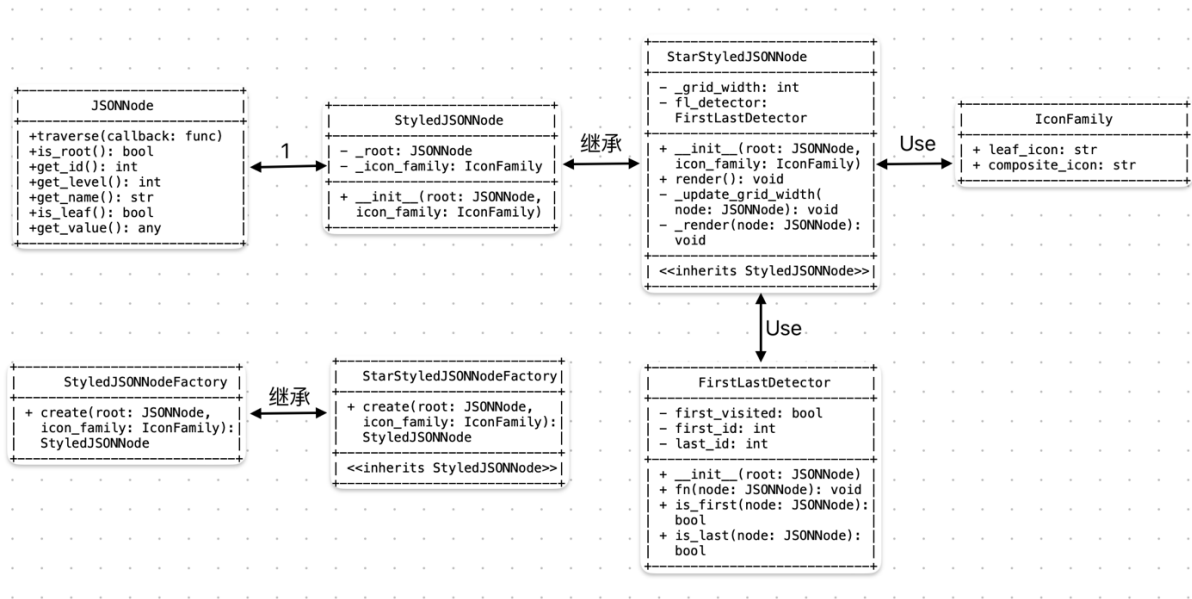
design pattern

吕烨振 21311154

代码使用了工厂模式,模版方法模式,装饰模式和访问者模式,具体如下

1. 工厂模式 (Factory Pattern) :
 - 涉及类 : `StyledJSONNodeFactory` 和 `StarStyledJSONNodeFactory`
 - 作用 : 工厂模式用于创建对象, 而无需指定确切的类。这使得代码更加灵活和可扩展。`StyledJSONNodeFactory` 和 `StarStyledJSONNodeFactory` 提供了 `create` 方法, 用于创建具体的 `StyledJSONNode` 实例, 封装了对象创建的细节, 客户端代码不需要知道具体的创建逻辑。
2. 模板方法模式 (Template Method Pattern) :
 - 涉及类 : `StyledJSONNode` 和 `StarStyledJSONNode`
 - 作用 : 模板方法模式在父类中定义一个算法的框架, 并允许子类在不改变算法结构的情况下重写某些步骤。`StyledJSONNode` 定义了基本的结构, 而 `StarStyledJSONNode` 重写了 `_render` 方法, 提供了具体的渲染逻辑。
3. 装饰模式 (Decorator Pattern) :
 - 涉及类 : `StarStyledJSONNode`
 - 作用 : 装饰模式用于在不改变现有类的情况下动态地为对象添加行为。
`StarStyledJSONNode` 继承自 `StyledJSONNode`, 并且添加了额外的行为和属性, 如 `_grid_width` 和 `_fl_detector`, 并覆盖了 `_render` 方法以实现特定的渲染逻辑。
4. 访问者模式 (Visitor Pattern) :
 - 涉及类 : `JSONNode` 和 `FirstLastDetector`
 - 作用 : 访问者模式用于将操作分离到不同的类中, 以便在不改变对象结构的情况下添加新的操作。`JSONNode` 类通过 `traverse` 方法接受一个函数作为参数, 允许外部逻辑 (如 `FirstLastDetector`) 操作节点树。`FirstLastDetector` 使用这个方法遍历节点树, 并检测第一个和最后一个节点。

其类图如下



测试 fje 截图如下

```
(base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s tree -i emoji-icon-family
├── oranges
│   └── mandarin
│       ├── clementine
│       └── tangerine: cheap & juicy!
└── apples
    ├── gala
    └── pink lady

(base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s rect -i emoji-icon-family
├── oranges
│   └── mandarin
│       ├── clementine
│       └── tangerine: cheap & juicy!
└── apples
    ├── gala
    └── pink lady

(base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s star -i emoji-icon-family
*── oranges *
│   *── mandarin *
│       *── clementine *
│       *── tangerine *: cheap & juicy!
└── apples *
    *── gala *
    *── pink lady *
```

```

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s tree -i poker-face-icon-family
├─ oranges
│   └─ mandarin
│       ├── clementine
│       └─ tangerine: cheap & juicy!
└─ apples
    ├── gala
    └─ pink lady

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s rect -i poker-face-icon-family
├─ oranges
│   └─ mandarin
│       ├── clementine
│       └─ tangerine: cheap & juicy!
└─ apples
    ├── gala
    └─ pink lady

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s star -i poker-face-icon-family
*─* oranges *─*
│   *─* mandarin *─*
│   │   *─* clementine *─*
│   │   *─* tangerine *: cheap & juicy! ─*
└─* apples *─*
    │   *─* gala *─*
    └─* pink lady *─*

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s tree -i default
├─ oranges
│   └─ mandarin
│       ├── clementine
│       └─ tangerine: cheap & juicy!
└─ apples
    ├── gala
    └─ pink lady

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s rect -i default
├─ oranges
│   └─ mandarin
│       ├── clementine
│       └─ tangerine: cheap & juicy!
└─ apples
    ├── gala
    └─ pink lady

• (base) mac@lyzdeMacBook-Air funny-json-explorer % fje -f test/test.json -s star -i default
*─* oranges *─*
│   *─* mandarin *─*
│   │   *─* clementine *─*
│   │   *─* tangerine *: cheap & juicy! ─*
└─* apples *─*
    │   *─* gala *─*
    └─* pink lady *─*

```