

Java设计模式之Interpreter模式Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_Java_E8_AE_BE_E8_AE_A1_c104_644813.htm Interpreter 定义: 定义语言的文法,并且建立一个解释器来解释该语言中的句子. Interpreter 似乎使用面不是很广,它描述了一个语言解释器是如何构成的,在实际应用中,我们可能很少去构造一个语言的文法.我们还是来简单的了解一下:首先要建立一个接口,用来描述共同的操作.

public interface AbstractExpression { void interpret(Context context). } 再看看包含解释器之外的一些全局信息 public interface Context { } AbstractExpression 的具体实现分两种:终结符表达式和非终结符表达式:来源:考试大

```
public class TerminalExpression implements AbstractExpression { public void interpret( Context context ) { } } 对于文法中没一条规则,非终结符表达式都必须的: public class NonterminalExpression implements AbstractExpression { private AbstractExpression successor. public void setSuccessor( AbstractExpression successor ) { this.successor = successor. } public AbstractExpression getSuccessor() { return successor. } public void interpret( Context context ) { } }
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com