

JAVA基础：使用Properties类带来的好处 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022_JAVA_E5_9F_BA_E7_A1_80_c104_145337.htm 许多开发者抱怨C++不能

像Java那样绑定Properties类。Java的Properties类内在包含一个文件，该文件用来读写Properties类中的属性，可以写成这样形式：`=`（例如：`ConnectToInternet = Use IE`）。使

用Properties类的好处就是你可以很轻松的理解和修改它们。

在本文的第一部分中，你将看到我们也可以在C++中使用

Properties类。本文的第二部分将向你演示通过使用操作

符`>>`和现在介绍C++ Properties文件的结构。该文件的每一行可以是下面三种情况中的某一种：空行（认为它是注释中的一部分）以`#`开始的注释行`=`行，这是给一个属性

赋值的语句 现在让我们再看看Properties类的特点：注释是持久性的（当保存Properties类时，它们不会丢失掉）。注

意每一个注释都属于某个属性。在`=`行上的注释行属于该`=`属性。当保存Properties类后，属性仍然保留自己的位置。它对各种字符类型都有效：`char`、`wchar_t`等等 Properties

类的使用相当简单：`save()`：保存属性

`has_property(strPropertyName)`：如果类中有该属性则返回`真`

`string get_property(strPropertyName)`：返回指定的属性（如果指定属性不存在，则抛出例外）

`set_property(strPropertyName, strPropertyValue)`：设置给定属性

`string get_property_comment(strPropertyName)`：返回属于指定属性的注释（如果指定属性的注释不存在，则抛出例外）

`set_property_comment(strPropertyName, strPropertyComment)`：

设置指定属性的注释（如果指定属性的注释不存在，则抛出例外）下面是file_reader_writer类以及相应例子的代码。运行它之后，请查看properties.txt文件。看看访问和修改它是多么容易的一件事。

```
#include <exception>#include <string>#include <sstream>#include <map>#include <vector>#include <fstream>#include <algorithm>#include <functional>//允许字符串转化template<inline std::basic_string convert_string( const std::basic_string & strSource) const{ return (strSource.find( ch) != std::basic_string::npos). }>//消除字符串中的空格template<std::basic_string trim_spaces( const std::basic_string & strSource){std::basic_string strSpaces. strSpaces = ( CharType) . strSpaces = ( CharType)\t. typedef std::basic_string string_type.string_type::const_iterator
```

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