

JAVA.UTIL.PROPERTY RESOURCEBUNDLE.HANDLEGETOBJECT METHOD

http://www.tutorialspoint.com/java/util/propertyresourcebundle_handlegetobject.htm Copyright © tutorialspoint.com

Description

The **java.util.PropertyResourceBundle.handleGetObject***String*key method returns the object in the bundle for the specified key. The method returns **null** if the bundle doesn't contain the required object.

Declaration

Following is the declaration for **java.util.PropertyResourceBundle.handleGetObject** method

```
public Object handleGetObject(String key)
```

Parameters

- **key** -- The key for the required object.

Return Value

This method returns the object for the specified key.

Exception

- **NA**

Example

The following example shows the usage of **java.util.PropertyResourceBundle.handleGetObject***String* method.

```
package com.tutorialspoint;

import java.io.IOException;
import java.io.InputStream;
import java.io.StringBufferInputStream;
import java.util.PropertyResourceBundle;

public class PropertyResourceBundleDemo {
    public static void main(String[] args) {
        // Prepare content for simulating property files
        String fileContent =
            "# Integer value 1\n" +
            "s1=1\n" +
            "# String value PropertyResourceBundleDemo\n" +
            "s2=PropertyResourceBundleDemo\n" +
            "# Date value Fri Jan 31 00:00:00 IST 3913\n" +
            "s3=Fri Jan 31 00:00:00 IST 3913";
        InputStream propStream = new StringBufferInputStream(fileContent);

        try {
            // Create property resource bundle
            PropertyResourceBundle bundle =
                new PropertyResourceBundle(propStream);

            // Get resources
            Object res1 = bundle.handleGetObject("s1");
            Object res2 = bundle.handleGetObject("s2");
            Object res3 = bundle.handleGetObject("s3");

            // Print resource contents
```

```
System.out.println("[Resource1] "+res1);
System.out.println("[Resource2] "+res2);
System.out.println("[Resource3] "+res3);
} catch (IOException e) {
    e.printStackTrace();
}
}
```

Let us compile and run the above program, this will produce the following result:

```
[Resource1] 1
[Resource2] PropertyResourceBundleDemo
[Resource3] Fri Jan 31 00:00:00 IST 3913
```

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js