

JAVA.UTIL.PROPERTYPERMISSION.NEWPERMISSION COLLECTION METHOD

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

Description

The **java.util.PropertyPermission.newPermissionCollection** method creates and returns a new **PermissionCollection** object.

Declaration

Following is the declaration for **java.util.PropertyPermission.newPermissionCollection** method

```
public PermissionCollection newPermissionCollection()
```

Parameters

- NA

Return Value

This method returns the newly created PermissionCollection object.

Exception

- NA

Example

The following example shows the usage of java.util.PropertyPermission.newPermissionCollection method.

```
package com.tutorialspoint;

import java.security.PermissionCollection;
import java.util.PropertyPermission;

public class PropertyPermissionDemo {
    private static PermissionCollection permissions;

    public static void main(String[] args) {
        // Build property permissions collection
        PropertyPermission permission =
            new PropertyPermission("java.*", "read");
        permissions = permission.newPermissionCollection();
        permissions.add(permission);
        permissions.add(new PropertyPermission("java.home.*", "read,write"));

        // Check permissions
        checkFileReadPermission("java.home");
        checkFileWritePermission("java.home");
        checkFileReadPermission("java.home usr");
        checkFileWritePermission("java.home usr");
    }

    private static void checkFileReadPermission(String path) {
        // Check permission on read action
        if(permissions.implies(new PropertyPermission(path, "read"))) {
            System.out.println("Has permissions on "+path+" for read");
        }
    }
}
```

```
private static void checkFileWritePermission(String path) {  
    // Check permission on read action  
    if(permissions.implies(new PropertyPermission(path, "write"))) {  
        System.out.println("Has permissions on "+path+" for write");  
    }  
}  
}
```

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

```
Has permissions on java.home for read  
Has permissions on java.home.usr for read  
Has permissions on java.home.usr for write  
Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js
```