

JAVA.UTIL.OBSERVABLE.NOTIFYOBSERVERS METHOD

http://www.tutorialspoint.com/java/util/observable_notifyobservers_object.htm

Copyright © tutorialspoint.com

Description

The **java.util.Observable.notifyObservers***Object***arg** method notify all of its observers if this object has changed and then call the `clearChanged` method to indicate that this object has no longer changed.

Declaration

Following is the declaration for **java.util.Observable.notifyObservers** method

```
public void notifyObservers(Object arg)
```

Parameters

- **arg** -- Any object.

Return Value

- NA

Exception

- NA

Example

The following example shows the usage of `java.util.Observable.notifyObservers`*Object* method.

```
package com.tutorialspoint;

import java.util.Observable;
import java.util.Observer;

class ObservedObject extends Observable {
    private String watchedValue;

    public ObservedObject(String value) {
        watchedValue = value;
    }

    public void setValue(String value) {
        // if value has changed notify observers
        if(!watchedValue.equals(value)) {
            System.out.println("Value changed to new value: "+value);
            watchedValue = value;

            // mark as value changed
            setChanged();
            // trigger notification with arguments
            notifyObservers(value);
        }
    }
}

public class ObservableDemo implements Observer {
    public String name;
    public ObservableDemo(String name) {
        this.name = name;
    }

    public static void main(String[] args) {
        // create watched and watcher objects
```

```
ObservedObject watched = new ObservedObject("Original Value");
// watcher object listens to object change
ObservableDemo watcher = new ObservableDemo("Watcher");

// add observer to the watched object
watched.addObserver(watcher);
// trigger value change
watched.setValue("New Value");
}

public void update(Observable obj, Object arg) {
    System.out.println(name+" called with Arguments: "+arg);
}
}
```

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

```
Value changed to new value: New Value
Watcher called with Arguments: New Value
```

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