

Me refiero a la implementación interna del método equals en la api de java

Buscar en alguna de las webs dadas en la documentación

▲ Yes by default equals method implements `==` in `Object` class . But you can Override the `equals` method in your own class to change the way `equality` is done between two objects of the same class. For example the `equals` method in `String` class is overridden as follows:

8

▼

```
public boolean equals(Object anObject) {
    if (this == anObject) {
        return true;
    }
    if (anObject instanceof String) {
        String anotherString = (String)anObject;
        int n = count;
        if (n == anotherString.count) {
            char v1[] = value;
            char v2[] = anotherString.value;
            int i = offset;
            int j = anotherString.offset;
            while (n-- != 0) {
                if (v1[i++] != v2[j++])
                    return false;
            }
            return true;
        }
    }
    return false;
}
```

STACKOVERFLOW

+1 TAMBIEN SE PUEDE BUSCAR USANDO NETBEANS (CREANDO UN EJEMPLO EN NETBEANS QUE INCLUYA LA CLASE STRING Y CLICANDO SOBRE LA PALABRA STRING PULSANDO CONTROL EN WINDOWS O COMMAND EN MAC OSX)

```
public boolean equals(Object anObject) {  
    if (this == anObject) {  
        return true;  
    }  
    if (anObject instanceof String) {  
        String aString = (String)anObject;  
        if (coder() == aString.coder()) {  
            return isLatin1() ? StringLatin1.equals(value, aString.value)  
                : StringUTF16.equals(value, aString.value);  
        }  
    }  
    return false;  
}
```