**Code Correctness: null Argument to equals() Development Mitigation SOP**

Code correctness vulnerabilities occur when an Object API is not used properly or as intended. Code correctness vulnerabilities can occur when equals(null) is called on an object because this will always be false. This will result in a NullPointerException.

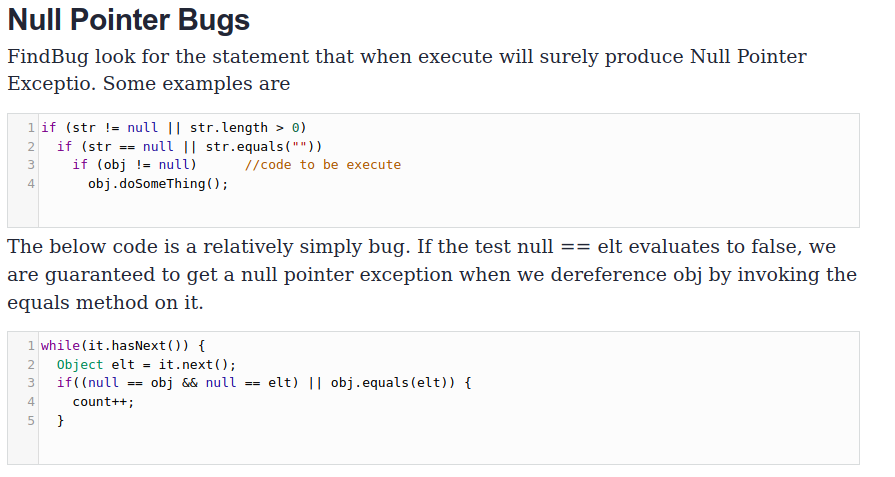
**Defense Against Code Correctness: null Argument to equals()**

In this case, the developer should check to see what the intension of the code was in the first place. The developer may have written obj.equals(null), but intended to write obj == null.

**Example**

**General Example**

obj.equals(null)



**Explanation**

This statement will always evaluate to false and result in a NullPointerException. The program uses the equals() method to compare an object with null. This comparison will always return false, since the object is not null. (If the object is null, the program will throw a NullPointerException).

**Recommendation**

obj == null

**References**

1. <https://vulncat.fortify.com/en/detail?id=desc.structural.java.code_correctness_null_argument_to_equals>
2. <https://www.cvedetails.com/cwe-details/398/Indicator-of-Poor-Code-Quality.html>
3. <https://cwe.mitre.org/data/definitions/754.html>