

# EXP52-J. Use braces for the body of an if, for, or while statement

Use opening and closing braces for `if`, `for`, and `while` statements even when the body contains only a single statement. Braces improve the uniformity and readability of code.

More important, it is easy to forget to add braces when inserting additional statements into a body containing only a single statement, because the conventional indentation gives strong (but misleading) guidance to the structure.

## Noncompliant Code Example

This noncompliant code example authenticates a user with an `if` statement that lacks braces:

```
int login;

if (invalid_login())
    login = 0;
else
    login = 1;
```

This program behaves as expected. However, a maintainer might subsequently add a debug statement or other logic but forget to add opening and closing braces:

```
int login;

if (invalid_login())
    login = 0;
else
    // Debug line added below
    System.out.println("Login is valid\n");
    // The next line is always executed
    login = 1;
```

The code's indentation disguises the functionality of the program, potentially leading to a security breach.

## Compliant Solution

This compliant solution uses opening and closing braces even though the body of the `if` and `else` bodies of the `if` statement are single statements:

```
int login;

if (invalid_login()) {
    login = 0;
} else {
    login = 1;
}
```

## Noncompliant Code Example

This noncompliant code example nests an `if` statement within another `if` statement, without braces around the `if` and `else` bodies:

```
int privileges;

if (invalid_login())
    if (allow_guests())
        privileges = GUEST;
else
    privileges = ADMINISTRATOR;
```

The indentation might lead the programmer to believe users are granted administrator privileges only when their login is valid. However, the `else` statement actually binds to the inner `if` statement:

```
int privileges;

if (invalid_login())
  if (allow_guests())
    privileges = GUEST;
  else
    privileges = ADMINISTRATOR;
```

Consequently, this defect allows unauthorized users to obtain administrator privileges.

## Compliant Solution

This compliant solution uses braces to remove the ambiguity, consequently ensuring that privileges are correctly assigned:

```
int privileges;

if (invalid_login()) {
  if (allow_guests()) {
    privileges = GUEST;
  }
} else {
  privileges = ADMINISTRATOR;
}
```

## Applicability

Failure to enclose the bodies of `if`, `for`, or `while` statements in braces makes code error prone and increases maintenance costs.

## Automated Detection

Tool	Version	Checker	Description
<a href="#">SonarQube</a>	6.7	<a href="#">S2681</a> <a href="#">S00121</a>	

## Bibliography

<a href="#">[GNU 2013]</a>	<a href="#">§5.3, "Clean Use of C Constructs"</a>
----------------------------	---

