

Java Expressions (EXP)

Mario Alviano

University of Calabria, Italy

A.Y. 2016/2017

Do not ignore values returned by methods

- Non-void methods return something
- ✗ Ignoring the returned value is a bad habit
- ✓ Verify the correct execution of the method

```
https://www.securecoding.cert.org/confluence/  
display/java/EXP00-J.+Do+not+ignore+values+  
returned+by+methods
```

Do not use a null in a case where an object is required

- Variables may refer null
- ✗ Invoking methods of a null object is a mistake
- ✓ Verify that the variable is not null

```
https://www.securecoding.cert.org/confluence/  
display/java/EXP01-J.+Do+not+use+a+null+in+a+case+  
where+an+object+is+required
```

Do not use the `Object.equals()` method to compare two arrays

- ✗ Do you really want to compare object references?
- ✓ Use `Arrays.equals()`
- ✓ Use `==` (?)

```
https://www.securecoding.cert.org/confluence/  
display/java/EXP02-J.+Do+not+use+the+Object.  
equals%28%29+method+to+compare+two+arrays
```

Do not use the equality operators when comparing values of boxed primitives

- ✗ Operators == and != compare object references
- ✓ The other operators unbox the wrappers
- ✓ Avoid confusion

```
https://www.securecoding.cert.org/confluence/display/java/EXP03-J.+Do+not+use+the+equality+operators+when+comparing+values+of+boxed+primitives
```

Do not pass arguments to certain Java Collections Framework methods that are a different type than the collection parameter type

- ✗ Some collection methods accept Object references
- ✓ Pay attention to `get()`, `contains()` and `remove()`

```
https://www.securecoding.cert.org/confluence/display/java/EXP04-J.+Do+not+pass+arguments+to+certain+Java+Collections+Framework+methods+that+are+a+different+type+than+the+collection+parameter+type
```

Do not follow a write by a subsequent write or read of the same object within an expression

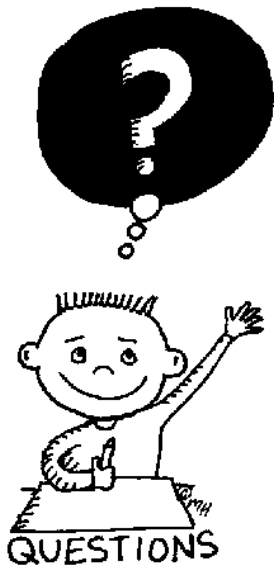
- ✗ Compressing code in one expressions is bad
- ✓ Declare temporary variables if you need them
- ✓ Efficiency will not be affected

```
https://www.securecoding.cert.org/confluence/  
display/java/EXP05-J.+Do+not+follow+a+write+by+a+  
subsequent+write+or+read+of+the+same+object+  
within+an+expression
```

Expressions used in assertions must not produce side effects

- ✗ Do not modify state variables in assertions
- Assertions disappear in production byte code
- ✓ Write assertions equivalent to no-op (no operation)

```
https://www.securecoding.cert.org/confluence/  
display/java/EXP06-J.+Expressions+used+in+  
assertions+must+not+produce+side+effects
```

END OF THE
LECTURE