WikipediA

# Expression-oriented programming language

An **expression-oriented programming language** is a programming language where every (or nearly every) construction is an expression and thus yields a value. The typical exceptions are macro definitions, preprocessor commands, and declarations, which expression-oriented languages often treat as statements rather than expressions. Some expression-oriented languages introduce a void return type to be yielded by expressions that merely cause side-effects.

ALGOL 68 and Lisp are examples of expression-oriented languages. Pascal is not an expression-oriented language. All functional programming languages are expression-oriented.

## Contents

## Criticism

Critics, including language designers,[1] blame expression-orientation for an entire class of programming mistakes wherein a programmer introduces an assignment expression where they meant to test for equality. For example, the designers of Ada and Java were so worried about this type of mistake, they restricted control expressions to those that evaluate strictly to the boolean data type.[2][3] The designers of Python had similar worries but took the alternative strategy of implementing assignment as a statement rather than an expression, thus prohibiting assignment from nesting inside of any other statement or expression.[4]

## Examples

- ALGOL 68
- Lisp

- Perl
- Ruby
- Haskell
- Rust
- Scala

## See also

- Command-query separation
- functional programming

# Notes

# References

1. Java Code Conventions "10.4 Variable Assignments" (http://java.sun.com/docs/codeconv/html/CodeConventions.doc9.html#547)
2. The Java Language Specification, Java SE 8 Edition "14.9 The if Statement" (https://docs.oracle.com/javase/specs/jls/se8/html/jls-14.html#jls-14.9)
3. Introducing Ada (http://www.adaic.org/whyada/intro4.html)
4. The Python Language Reference "6.2. Assignment statements" (https://docs.python.org/reference/simple_stmts.html#assignment)

This page was last edited on 1 September 2020, at 15:22 (UTC).