



← → W http://en.wikipedia.org/wiki/Java_bytecode W Java bytecode - Wikipedia, ... X


File Edit View Favorites Tools Help



WIKIPEDIA
The Free Encyclopedia

- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia
- Wikimedia Shop
- Interaction
 - Help
 - About Wikipedia
 - Community portal
 - Recent changes
 - Contact page
- Tools
- Print/export
- Languages 
 - Bosanski
 - Català
 - Español
 - Français
 - 한국어
 - Bahasa Indonesia
 - Magyar
 - 日本語
 - Polski
 - Português
 - Русский
 - Sрpskohrvatski / српскохрватски

Article Talk Read Edit View history



Amending our Terms of Use:
Please comment on a proposed amendment regarding undisclosed paid editing.

Java bytecode

From Wikipedia, the free encyclopedia

Java bytecode is the form of instructions that the Java virtual machine executes. Each bytecode opcode is one byte in length, although some require parameters, resulting in only 127 of the possible 256 opcodes are used. 51 are reserved for future use. Beyond that, Sun Microsystems, the original developer of the Java platform, had set aside three values for internal use.

Contents [hide]

- 1 Relation to Java
- 2 Instructions
- 3 Model of computation
- 4 Example
- 5 Generation
- 6 Execution
- 7 Support for dynamic languages
- 8 See also
- 9 References
- 10 External links

Relation to Java [edit]

A Java programmer does not need to be aware of or understand Java bytecode at all. However, as suggested in the IBM developerWorks journal, "Understanding bytecode or assembly code generated by a Java compiler helps the Java programmer in the same way that knowledge of assembly helps the C or C++ programmer."^[2]

Instructions [edit]

See also: Java bytecode instruction listings

As each byte has 256 potential values, there are 256 possible opcodes. Of these, 0x00 through 0xca, 0xfe, and 0xff are assigned values. 0xca is reserved as a breakpoint opcode, its type is not used by the language. Similarly, 0xfe and 0xff are not used by the language and are reserved for internal use by the Java virtual machine.

Instructions fall into a number of broad groups:

- Load and store (e.g. aload, b, istore)