

Extra Reading

Table of Contents

- Principles
- Performance
- Memory management
- Syntax
- Method Keywords

Wikipedia

Principles

Primary goals in creation of Java: 1. simple, object-oriented, familiar 2. robust, secure 3. architecture-neutral, portable 4. high-performance execution 5. interpreted, threaded, dynamic

Performance

- Java typically is slower and uses more memory than C++
 - overhead due to interpreter from Java bytecode to machine code

Memory management

- Java has an automatic garbage collector
 - programmer determines when objects are created
 - Java runtime recovers memory once objects are no longer in use
 - when no references to an object remain, unreachable memory becomes eligible to be freed by the garbage collector
 - memory leaks still occur if code holds references to objects no longer needed
 - * in this case throws null pointer exception
- garbage collection occurs at idle, or is triggered if there is insufficient memory on the heap to allocate a new object, which can cause program to stall
- explicit memory management and pointer arithmetic is not supported
- variables of primitive data types are stored directly in fields (for objects) or on the stack (for methods) rather than on the heap

Syntax

- Java does not support
 - operator overloading
 - multiple inheritance for classes

Method Keywords

Keywords applied to methods - **public**: method can be called from code in other classes, or class may be used by classes outside the class hierarchy - **static**: associated only with the class, and not a specific instance of the class - can be invoked without a reference to an object - cannot access class members that are not also static - **void**: main method does not return a value