

# Glossary

**Abbreviation**

A shortened form of a word or phrase used chiefly in writing to represent the complete form.

**Abstract Class**

An abstract class is a type of class that cannot be instantiated, it merely exists to be extended and contains methods and variables common to all sub-classes.

**Abstraction**

A representation that only captures essential aspects of something thereby reducing complexity.

**Access Modifier**

An access modifier is a modifier that changes the visibility of a class or its members.

**Algorithm**

A step-by-step problem-solving procedure, especially an established, recursive computational procedure for solving a problem in a finite number of steps.

**API**

*Application Programming Interface*, Set of related classes and methods that provide certain functionality. The API represents the parts of a class exposed through access modifiers to code written by other programmers.

**Application Failure**

A situation resulting from a fatal error in a computer program that renders the program useless.

**Application**

A computer program that helps the user accomplish a specific task; for example, a word processing program. Application programs should be distinguished from system programs, which control the computer.

**Archive**

A file that contains a group of files and possibly directories which must be extracted in order to use them. An archive may optionally be compressed which would require it to be decompressed before the files can be extracted.

**Attribute**

A single piece of information that represents a property present in all instances of a class. An attribute is often modelled as a variable in a class.

**Automated**

Acting or operating in a manner essentially independent of external influence or control.

**Binaries**

An application binary is the compiled form of a program, the actual machine code that is executed by a system in order to run the program.

**Browser**

A program such as Mosaic, Netscape, Internet Explorer, FireFox and others that are used to view hypertext documents on the World Wide Web.

**Bug**

A colloquial term for a defect.

**Build**

The process of compiling and integrating all components of a piece of software, incorporating all changes made since the last build.

**Byte**

The amount of memory space used to store one ASCII character, which is usually 8 bits. A bit is the smallest unit of information a computer can hold; short for binary digit, the value can be either one or zero.

**Child Node**

A child node or descendant node is a node in a tree data structure that is linked to by a parent node.

**Class Member**

Class members are items that belong to that class, usually methods and variables and also nested classes.

**Class**

A class is the definition of a programmer defined type, the blueprint used to construct objects of that type. It is a logical unit that provides procedural and data abstraction.

**Clean**

A part of the build process in which temporary files used during the build process are removed.

**Click**

To press a button on a mouse or other pointer. Clicking is used to place the cursor, when working in text, or to select an object on the screen or a menu option.

**Command**

An instruction given to the computer, by means of a keyboard, punch card, mouse, voice command, or other method.

**Comment**

A portion of source code used to provide information about the source code. This

section is ignored by the compiler.

**Compiler**

A computer program that translates a high-level programming language into machine language. The program fed into the compiler is called the source program; the generated machine language program is called the object program.

**Component**

A piece of software with a clear function that can be isolated and replaced by another component with equivalent functionality.

**Compression**

The temporary coding of data in a way that saves storage space or transmission time. Most text files can be compressed to about half their normal size. Graphics can be compressed to 10 percent of their original size.

**Computer**

An electronic device that has the ability to store, retrieve, and process data, and can be programmed with instructions that it remembers. The physical parts that make up a computer (the central processing unit, input, output, and memory) are called hardware. Programs that tell a computer what to do are called software.

**Configuration**

The way that a program or computer is set up, various settings are used in the configuration.

**Console**

The command prompt window that receives output from a `System.out.print` call.

**Constructor**

An object instance is created by calling the constructor of a class. A constructor is method-like block of code that is called when the object is created. Typically constructors initialize data members and acquire resources the object may require.

**Debugging**

The process of locating errors in source code whether logical or syntactic and fixing them often through the use of a debugger.

**Declaration**

A statement that declares a class, interface, method, package or variable in a source file. It can also explicitly initialize a variable by assigning a value to it.

**Default**

An instruction that a computer assumes, unless the user gives it other instructions. For example, if the default typeface on a word processing program is Times Roman, the user may instruct the machine to use a different default typeface.

**Defect**

A flaw in any aspect of the system. See Error.

**Deflated**

An extracted archive file. See Archive.

**Deprecated**

An API item that is considered obsolete and on its way out, usually in favor of something better. Usually, though the item may have been originally included as part of an API, the use of it is no longer advised, and slowly support for the item is phased out.

**Design**

Process to find and describe a way to implement the system's requirements.

**Developer**

A computer programmer whose responsibility is the development of new software rather than the maintenance and updating of existing software.

**Development**

The process of developing software.

**Dialog**

A box on the computer screen that lets the user communicate with the computer. A dialog box can be used to enter information, set options, or give commands to the computer. The dialog box gives the user choices (such as open file, delete, save) which can be selected by clicking with the mouse.

**Digital Signing**

An attempt to mimic the offline act of a person applying their signature to a paper document.

**Directory**

On Macintosh and Windows 95 screens, files can be organized by placing them into folders that look like office file folders. These file folders are directories.

**Distributed System**

A system in which various components occur on different systems usually connected via a network.

**Documentation**

Instructions that come with a software program, which may include paper or electronic manuals, README files, and online help.

**Download**

To transfer files or data from one computer to another. To download means to receive; to upload means to transmit

**Drive**

A device that spins disks or tapes in order to read and write data; for example, a hard drive, floppy drive, CD-ROM drive, or tape drive.

## Glossary

**Editor**

A program that is used to make changes to a file; for example a text editor or an image editor.

**Element**

A block in a markup document like a tag in an HTML document.

**Embedded**

An object, such as a graphic, which has been placed in a document from another file.

**Enumeration**

An object that assists in iterating over a set of objects.

**Environment Variable**

# A variable that specifies how an operating system or another program runs, or the devices that the operating system recognizes.

**Error**

A mistake made by a software developer that introduces a defect in a computer program.

**Exception**

Refers to an "exceptional condition" which is an abnormal situation that alters the flow of an application.

**Executable**

A program that a computer can directly execute. See Binaries.

**Expression**

An expression which uses symbols to represent numbers or abstract concepts for use in operations similar to arithmetic.

**Extract**

To deflate an archive. See Archive.

**Fatal Error**

An error that causes a program to stop executing. See Error and Application Failure.

**File**

A block of information in the form of bytes, stored together on a computer or external digital storage medium, and given a name. A file may be a program, a document, a database, or some other collection of bytes.

**Fileset**

An abstract representation of a set of files. For instance "\*" . java" refers to all files with a ".java" extension.

**Font**

A complete set of type characters in a particular style and size.

**Framework**

A Skeletal software component that performs functions required by a system and which is incorporated into the design of such systems.

**Function**

A method that performs some form of processing in order to return a result. For example, a function to calculate the sum of two integers.

**Functionality**

The capabilities or behaviors of a program, application, or system; the total set of its features.

**Generate**

The automated creation of something.

**Getter**

A method that is responsible for relaying information about a class's state by returning the value of one of its variables.

**Hard Coded**

Hard-code refers to the software development practice of embedding data directly into the source code of a program or other executable object, instead of obtaining that data from external sources such as a configurations file or command-line parameters.

**HTML**

*Hypertext Markup Language*, The language used to create World Wide Web pages, with hyperlinks and markup for text formatting.

**IDE**

*Integrated Development Environment*, an IDE combines the editor, compiler and other useful tools in the same software package. Its advantage is that when a program with syntax errors is compiled, the programmer sees the error messages and the original program at the same time -- this makes debugging much easier.

**Import**

The act of including another source file's declarations for use in one's own source file.

**Initialize**

Set a variable to a starting value.

**Input**

Something put into a system or expended in its operation to achieve output or a result.

**Install**

To load and configure a piece of software on a computer.

**Interface**

Publically vsiible operations provided by an API.

**Internet**

A network of computer networks which operates world-wide using a common set of communications protocols.

**Iterator**

A data type used to mark a position in a collection of data (eg a linked list) and to move from item to item within the collection.

**Jar**

*Java Archive file*, An archive (like a ZIP file) containing Java class files and images. JAR files are used to package Java applications for deployment.

**Java**

Java is an object-oriented programming language developed initially by James Gosling and colleagues at Sun Microsystems. The language, initially called Oak (named after the oak trees outside Gosling's office), was intended to replace C++, although the feature set better resembles that of Objective C. Java should not be confused with JavaScript, which shares only the name and a similar C-like syntax. Sun Microsystems currently maintains and updates Java regularly.

**Javadoc**

Pertaining to the tool that parses the declarations and documentation comments in a set of source files and produces a set of HTML pages describing the classes, inner classes, interfaces, constructors, methods, and fields. The Javadoc tool from Sun Microsystems is used for generating API documentation in HTML format from doc comments in source code.

**JDK**

*Java Development Kit*, A software package that contains the minimal set of tools needed to write, compile, debug, and run Java applets and applications.

**JRE**

*Java Runtime Environment*, The JRE includes the Java Virtual Machine (JVM), core classes, and supporting files required to run Java programs.

**JVM**

*Java Virtual Machine*, An abstract computing machine, or virtual machine, JVM is a platform-independent programming language that converts Java bytecode into machine language and executes it.

**Keyword**

A keyword is an identifier which indicates a specific command. Many times the keyword is restricted to that core language usage, in which case it is also considered a reserved word.

**Library**

A library is a collection of subprograms used to develop software. Libraries are distinguished from executables in that they are not independent programs.

**Logger**

A component with sole responsibility of handling the logging operations in a logging framework.

**Logging**

The process of storing information about events that occurred in an application, used for debugging purposes.

**Logical Unit**

A group of code blocks or statements that together form a logical cohesive whole.

**Maintain**

The process of fixing bugs in, and adding new features to, existing software.

**Manifest**

File containing meta information incorporated into a Jar file.

**Markup**

Syntactically delimited characters added to the data of a document to represent its structure. There are four different kinds of markup: descriptive markup (tags), references, markup declarations, and processing instructions.

**Member**

Elements of a class, including methods, variables and nested classes.

**Memory**

The terms "storage" and "memory" refer to the parts of a digital computer that retain physical state (data) for some interval of time, possibly even after electrical power to the computer is turned off.

**Message**

Any information sent as a component interacts with another.

**Method**

Section of source code that performs a specific function, has a name, may be passed parameters and may return a result. Methods occur only within classes.

**Multiline**

Spanning across multiple lines.

**Namespace**

A scoping construct to subdivide the set of names and their visibility within a system. In many languages, this is tied to other computational constructs, eg, classes, procedures, modules, packages. A mechanism used to resolve naming conflicts.

**Object**

An instance, once a class is instantiated it becomes an object.

**Options**

Alternatives or choices, often refers to settings or preferences in a program that

may be set according to the users preference or taste.

**Output**

Information that has been manipulated by the central processing unit (CPU) of the computer, and displayed either on the video monitor or rendered on paper or film as hard copy, or saved on disk in a digital format.

**Package**

An entity that groups related classes together, the name must reflect the directory structure used to store the classes.

**Parameter**

Input passed to a method for processing. Parameters are a way of allowing the same method to operate on different data without re-specifying the instructions.

**Parse**

Parsing is the process of splitting up a continuous stream of characters (read from a file or keyboard input, for example) into meaningful tokens, and then building a parse tree from those tokens.

**Performance**

A major factor in determining the overall productivity of a system, performance is primarily tied to availability, throughput and response time.

**Platform**

In computing, a platform describes some sort of framework, either in hardware or software, which allows software to run. Typical platforms include a computer's architecture, operating system, or programming languages and their runtime libraries.

**Plug-In**

A small piece of software that adds features to already existing, usually large, programs.

**Popup**

A Menu that provides no visual cue to its presence, but simply pops up when a user performs a particular action. Popup Menus are associated with a particular area of the workspace, such as the client area of an application, and a user must memorize where these areas are.

**Portability**

A measure of system independence; portable programs can be moved to a new system by recompiling without having to make any other changes.

**Preferences**

See Options.

**Priority**

Precedence: status established in order of importance or urgency.

**Private**

Access modifier that renders members of a class invisible to components outside of that class.

**Process**

A process is a running instance of a program, including all variables and other state. A multitasking operating system switches between processes to give the appearance of simultaneous execution, though in fact only one process can be executing at once per CPU core.

**Program**

See Application.

**Programmer**

Someone who writes source code that is compiled into a program. See Developer.

**Project**

A project is a temporary endeavor undertaken to create a unique product or service.

**Properties**

A set of variables in the form `name=value` often saved in a file and used for configuration.

**Property**

A variable in the form `name=value`.

**Protected**

Access modifier that renders class members invisible to components outside of a class with the exception of sub-classes.

**Random Access**

The process of selecting information in a arbitrary order, not based on the physical order or sequence of its storage.

**Reader**

An object in Java that is used to read a character stream. See Stream.

**Recursive**

Referring back to itself. A method that calls itself until some base condition is true.

**Refactor**

Refactoring is the process of rewriting written material to improve its readability or structure, with the explicit purpose of keeping its meaning or behavior.

**Return**

The act of passing a result back to the caller of a method or function.

**Run**

To execute a program.

**Runtime**

## Glossary

During the time in which a program is executing.

### **Script**

A file containing operating system commands that are processed in a batch method, one at a time, until complete.

### **Sealed**

A Sealed Jar guarantees that all classes in a package come from the same code source.

### **Security**

Techniques and practices that preserve the integrity of computer systems, and digital library services and collections.

### **Serialize**

Encode a data structure as a sequence of bytes.

### **Setter**

Method that transforms the state of an object by set the value of one of its variables.

### **Software Developemnt**

A set of activities that results in software products. Software development may include new development, modification, reuse, re-engineering, maintenance, or any other activities that result in software products.

### **Software**

See Application.

### **Solution**

The result of solving a problem.

### **Standalone**

A program, function, files or system that operates on its own and has no dependencies on other programs, functions, files or systems.

### **Statement**

An entity in a programming language which is typically the smallest indivisible unit of execution.

### **Stream**

A continuous flow of data, usually digitally encoded, designed to be processed sequentially. Also called a bitstream.

### **String**

A group or sequence of characters.

### **Stylesheet**

An ASCII text document that is attached to an SGML or XML encoded document and that contains instructions to specify the formatting and display of the encoded document, or to transform it into another format.

### **Sub-class**

A class that is an extension of another class and inherits public and protected variables and methods from the other class. Also known as a derived class.

**Super Constructing**

A derived or sub class calls the base or super class's constructor in order for the super class to initialize itself properly. This is achieved through the `super` mechanism in Java.

**Super-class**

A base class from which another class derives.

**Swing**

Swing is a GUI toolkit for Java. Swing is one part of the Java Foundation Classes (JFC) and includes graphical user interface (GUI) widgets such as text boxes, buttons, split-panes, and tables.

**Syntax Highlight**

Syntax highlighting is a feature of some text editors that displays text, especially source code, in different colors and fonts according to the category of terms. This feature eases writing in a structured language such as a programming language or a markup language as both structures and syntax errors are visually distinct.

**Syntax**

The rules by which the words in a program are combined to form commands to a computer.

**Tag**

A tag is a marker embedded in a document that indicates the purpose or function of the element. Each element has a beginning tag and an end tag.

**Target**

The step in a build process that can be executed, usually by Apache Ant in a build file.

**Task**

A sequence of instructions treated as a basic unit of work.

**Template**

A set of pre-designed formats for text and graphics on which new pages and webs can be based.

**Test Case**

A set of conditions or variables under which a tester will determine if a requirement upon an application is partially or fully satisfied. It may take many test cases to determine that a requirement is fully satisfied.

**Test Driven Development**

Test-driven development (TDD) is a programming technique heavily emphasized in Extreme Programming. Essentially the technique involves writing your tests first then implementing the code to make them pass

**Test Suite**

A test suite is a set of related tests, usually pertaining to a group of features or software component.

**Text**

Data consisting of a sequence of characters, as opposed to binary numbers, images, graphics commands, executable programs, and the like.

**Throw**

In Java terms, an exception is said to be thrown if the exception is raised. A method may throw an exception if an error occurs in its processing.

**Tool**

A device that provides a mechanical or mental advantage in accomplishing a task.

**Transformation**

Transformation is a form of conversion in which a file is converted into a file format with a comparable structure (eg from XML to XML, or from SGML to HTML). Usually, this form of conversion can be carried out very well.

**Tutorial**

A mode of instruction that presents content, checks understanding or performance, and continues on to the next relevant selection of content. Tutorials may be linear or branched.

**Unit Test**

A unit test is a method of testing the correctness of a particular module of source code.

**User**

A person who requires a computer for the performance of a task or recreational activity. Also called an end-user.

**Variable**

A quantity that may assume any one of a set of values.

**Versioning**

Versioning is a mechanism that keeps track of all changes in content and code and allows any change to be 'rolled back' to any previous version. This also means that a deleted file can be recovered to its last saved state.

**Visibility**

The accessibility of methods and instance variables to other classes and packages, through the use of access modifiers: public, protected, package or private.

**Wildcard**

A special character such as an asterisk (\*) or a question mark (?) that you can use to represent one or more characters. Any character or set of characters can

replace a pattern matching character.

**Window**

A window is a visual area, usually rectangular in shape, containing some kind of user interface, displaying the the output of and allowing input for one of a number of simultaneously running computer processes. Windows are primarily associated with graphical displays, where they can be manipulated with a pointer.

**Windows**

Microsoft Windows is a range of closed source proprietary commercial operating systems for personal computers and servers.

**Zip**

To zip a file is to compress it into an archive so that it occupies less disk space.