JAVA Quick Reference

Author: Jialong He
Jialong_he@bigfoot.com
http://www.bigfoot.com/~jialong_he

Simple Data Types

byte	8	-128..127
short	16	-32,768..32,767
int	32	-2,147,483,648.. 2,147,483,647
long	64	-9,223,372,036,854,775,808.. 9,223,372,036,854,775,807
float	32	3.4e-0.38.. 3.4e+0.38
double	64	1.7e-308.. 1.7e+308
char	16	Complete Unicode Character Set
Boolean		true, false

decare a variable:
type identifier [= *value*];

Arrays

int *array_name*[ ]; *array_name*=new int[100];	declare an integer array and allocate 100 elements of memory.
int *array_name*[ ] = new int [100];	declare and allocate an integer array in one statement.
int *array_name* [ ] = {1, 2, 3, 4};	
int *array_name*[] = new int[10][20];	multidimensional array.
null	for reference type (class, array).

Class

```
{public|final|abstract} class name
{  [class_variable_declarations]
   public static void main{String[] args) {statements}
   [methods]
}
```

this, super

Methods

{**public** | **private** } [**static**] {*type* | **void**} name(*arg, ..., arg*) {statements}

Variable

{**public** | **private** } [**static**] *type* name [= *expression*];

Arithmetic Operators

+, -, *, /	addition, subtraction, multiplication , division
%, ++, --	modulus, increment, decrement.

Relational Operators

==, !=, >, <,	equal, not equal, greater, less,
>=, <=	greater or equal, less or equal.

Logical Operators

&, |, !, ^, ||, &&,	AND, OR, NOT, XOR, short -circuit OR, AND

Bitwise Operators

&, |, ~, ^,	AND, OR, NOT, XOR
>>, >>>, <<	shift right, shift right zero fill, shift left.

Comments

```
// rest of line
/* multiline comment */
/** documentation comment */
```

Compile and Run

```
javac nameOfFile.java
java nameOfFile
```

CLASSPATH must set correctly.
The name of the file has to match exactly the name of the class.

Flow Control

```
if (Boolean-expression) statement1; [ else statement2; ]
while loop
[initialization]
while (termination -clause) {
body;
[iteration;]
}
do while loop
do {
body;
[iteration;]
} while (termination-clause);
for loop
for (initialization; termination -clause; iteration)
body;
```

```
Program Structure
class className {
public static void main (String args[ ]) {
statements;
}
method definition1
…
method definitionN
}
```

Key words

abstract	boolean	break	byte	byvalue
case	cast	catch	char	class
const	continue	default	do	double
else	extends	false	final	finally
float	for	future	generic	goto
if	implements	import	inner	instanceof
int	interface	long	native	new
null	operator	outer	package	private
protected	public	rest	return	short
static	super	switch	synchronized	this
throw	throws	transient	true	try
var	void	volatile	while	

Java packages

java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transf erring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to fonts.
java.awt.geom	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spi	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.
java.awt.image.renderable	Provides classes and interfaces for producing rendering-independent images.
java.awt.print	Provides classes and interfaces for a general printing API.
java.beans	Contains classes related to developing *beans* – components based on the JavaBeans[TM] architecture.

Package	Description
java.beans.beancontext	Provides classes and interfaces relating to bean context.
java.io	Provides for system input and output through data streams, serialization and the file system.
java.lang	Provides classes that are fundamental to the design of the Java programming language.
java.lang.ref	Provides reference-object classes, which support a limited degree of interaction with the garbage collector.
java.lang.reflect	Provides classes and interfaces for obtaining reflective information about classes and objects.
java.math	Provides classes for performing arbitrary-precision integer arithmetic (BigInteger) and arbitrary-precision decimal arithmetic (BigDecimal).
java.net	Provides the classes for implementing networking applications.
java.nio	Defines buffers, which are containers for data, and provides an overview of the other NIO packages.
java.nio.channels	Defines channels, which represent connections to entities that are capable of performing I/O operations, such as files and sockets; defines selectors, for multiplexed, non-blocking I/O operations.
java.nio.channels.spi	Service-provider classes for the java.nio.channels package.
java.nio.charset	Defines charsets, decoders, and encoders, for translating between bytes and Unicode characters.
java.nio.charset.spi	Service-provider classes for the java.nio.charset package.
java.rmi	Provides the RMI package.
java.rmi.activation	Provides support for RMI Object Activation.
java.rmi.dgc	Provides classes and interface for RMI distributed garbage-collection (DGC).
java.rmi.registry	Provides a class and two interfaces for the RMI registry.
java.rmi.server	Provides classes and interfaces for supporting the server side of RMI.
java.security	Provides the classes and interfaces for the security framework.
java.security.acl	The classes and interfaces in this package have been superseded by classes in the java.security package.
java.security.cert	Provides classes and interfaces for parsing and managing certificates, certificate revocation lists (CRLs), and certification paths.
java.security.interfaces	Provides interfaces for generating RSA (Rivest, Shamir and Adleman AsymmetricCipher algorithm) keys as defined in the RSA Laboratory Technical Note PKCS#1. and DSA (Digital Signature Algorithm) keys as defined in NIST's FIPS-186.
java.security.spec	Provides classes and interfaces for key specifications and algorithm parameter specifications.
java.sql	Provides the API for accessing and processing data stored in a data source (usually a relational database) using the Java™ programming language.
java.text	Provides classes and interfaces for handling text, dates, numbers, and messages in a manner independent of natural languages.
java.util	Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array).
java.util.jar	Provides classes for reading and writing the JAR (Java ARchive) file format, which is based on the standard ZIP file format with an optional manifest file.
java.util.logging	Provides the classes and interfaces of the Java™ 2 platform's core logging facilities.
java.util.prefs	This package allows applications to store and retrieve user and system preference and configuration data.
java.util.regex	Classes for matching character sequences against patterns specified by regular expressions.
java.util.zip	Provides classes for reading and writing the standard ZIP and GZIP file formats.

Java extension packages

Package	Description
javax.accessibility	Defines a contract between user-interface components and an assistive technology that provides access to those components.
javax.crypto	Provides the classes and interfaces for cryptographic operations.
javax.crypto.interfaces	Provides interfaces for Diffie-Hellman keys as defined in RSA Laboratories' PKCS #3.
javax.crypto.spec	Provides classes and interfaces for key specifications and algorithm parameter specifications.
javax.imageio	The main package of the Java Image I/O API.
javax.imageio.event	A package of the Java Image I/O API dealing with synchronous notification of events during the reading and writing of images.
javax.imageio.metadata	A package of the Java Image I/O API dealing with reading and writing metadata.
javax.imageio.plugins.jpeg	Classes supporting the built-in JPEG plug-in.
javax.imageio.spi	A package of the Java Image I/O API containing the plug-in interfaces for readers, writers, transcoders, and streams, and a runtime registry.
javax.imageio.stream	A package of the Java Image I/O API dealing with low-level I/O from files and streams.
javax.naming	Provides the classes and interfaces for accessing naming services.
javax.naming.directory	Extends the javax.naming package to provide functionality for accessing directory services.
javax.naming.event	Provides support for event notification when accessing naming and directory services.
javax.naming.ldap	Provides support for LDAPv3 extended operations and controls.
javax.naming.spi	Provides the means for dynamically plugging in support for accessing naming and directory services through the javax.naming and related packages.
javax.net	Provides classes for networking applications.
javax.net.ssl	Provides classes for the secure socket package.
javax.print	Provides the principal classes and interfaces for the JavaTM Print Service API.
javax.print.attribute	Provides classes and interfaces that describe the types of JavaTM Print Service attributes and how they can be collected into attribute sets.
javax.print.attribute.standard	Package javax.print.attribute.standard contains classes for specific printing attributes.
javax.print.event	Package javax.print.event contains event classes and listener interfaces.
javax.rmi	Contains user APIs for RMI-IIOP.
javax.rmi.CORBA	Contains portability APIs for RMI-IIOP.
javax.security.auth	This package provides a framework for authentication and authorization.
javax.security.auth.callback	This package provides the classes necessary for services to interact with applications in order to retrieve information (authentication data including usernames or passwords, for example) or to display information (error and warning messages, for example).
javax.security.auth.kerberos	This package contains utility classes related to the Kerberos network authentication protocol.
javax.security.auth.login	This package provides a pluggable authentication framework.
javax.security.auth.spi	This package provides the interface to be used for implementing pluggable authentication modules.
javax.security.auth.x500	This package contains the classes that should be used to store X500 Principal and X500 Private Credentials in a Subject.
javax.security.cert	Provides classes for public key certificates.
javax.sound.midi	Provides interfaces and classes for I/O, sequencing, and synthesis of MIDI (Musical Instrument Digital Interface) data.

javax.sound.midi.spi	Supplies interfaces for service providers to implement when offering new MIDI devices, MIDI file readers and writers, or sound bank readers.
javax.sound.sampled	Provides interfaces and classes for capture, processing, and playback of sampled audio data.
javax.sound.sampled.spi	Supplies abstract classes for service providers to subclass when offering new audio devices, sound file readers and writers, or audio format converters.
javax.sql	Provides the API for server side data source access and processing from the JavaTM programming language.
javax.swing	Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms.
javax.swing.border	Provides classes and interface for drawing specialized borders around a Swing component.
javax.swing.colorchooser	Contains classes and interfaces used by the JColorChooser component.
javax.swing.event	Provides for events fired by Swing components.
javax.swing.filechooser	Contains classes and interfaces used by the JFileChooser component.
javax.swing.plaf	Provides one interface and many abstract classes that Swing uses to provide its pluggable look-and-feel capabilities.
javax.swing.plaf.basic	Provides user interface objects built according to the Basic look and feel.
javax.swing.plaf.metal	Provides user interface objects built according to the Java look and feel (once codenamed Metal), which is the default look and feel.
javax.swing.plaf.multi	Provides user interface objects that combine two or more look and feels.
javax.swing.table	Provides classes and interfaces for dealing with javax.swing.JTable.
javax.swing.text	Provides classes and interfaces that deal with editable and noneditable text components.
javax.swing.text.html	Provides the class HTMLEditorKit and supporting classes for creating HTML text editors.
javax.swing.text.html.parser	Provides the default HTML parser, along with support classes.
javax.swing.text.rtf	Provides a class (RTFEditorKit) for creating Rich-Text-Format text editors.
javax.swing.tree	Provides classes and interfaces for dealing with javax.swing.JTree.
javax.swing.undo	Allows developers to provide support for undo/redo in applications such as text editors.
javax.transaction	Contains three exceptions thrown by the ORB machinery during unmarshalling.
javax.transaction.xa	Provides the API that defines the contract between the transaction manager and the resource manager,

which allows the transaction manager to enlist and delist resource objects (supplied by the resource manager driver) in JTA transactions.

javax.xml.parsers	Provides classes allowing the processing of XML documents.
javax.xml.transform	This package defines the generic APIs for processing transformation instructions, and performing a transformation from source to result.
javax.xml.transform.dom	This package implements DOM-specific transformation APIs.
javax.xml.transform.sax	This package implements SAX2-specific transformation APIs.
javax.xml.transform.stream	This package implements stream- and URI-specific transformation APIs.