

Introduction to Java

JAVA

```

void compileFile(final SyntaxNode sn) throws CodeException {
    for (Iterator ite=sn.getChildren().createIterator(); ite.hasNext(); ) {
        final SyntaxNode cn = (SyntaxNode) ite.next();
        final Rule rule = cn.getRule();
        if (RULE_PACKAGE == rule) {
            cn.setCharByRule(RULE_REFERENCE, getTokensChars());
        } else if (RULE_IMPORT == rule) {
            //TODO handle static and *
            final SyntaxNode n = cn.getCharByRule(RULE_IMPORT);
            final Char fullN = cn.getTokensChars();
            final Char[] parts = fullN.split('.');
        }
    }
}

```

Introduction to Java

Java

Language Characteristics

- Syntax – easy Syntax
 - Strongly typed
 - Pure OOP
 - Size Of primitives is laid down
 - Garbage Collector
 - Multithreading Language
 - Support Exceptions
-

Introduction to Java

Java

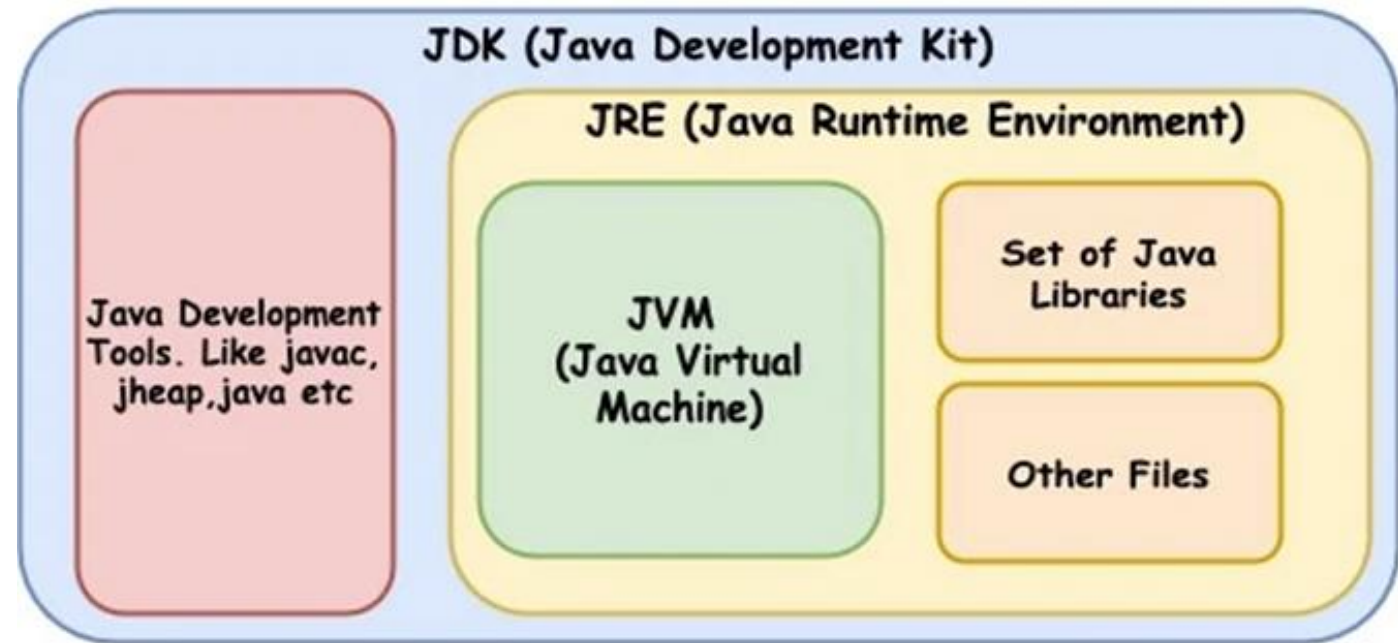
Java Does NOT Have

- Memory address (pointers) arithmetic
 - Preprocessor
 - Automatic type conversion
 - Global functions and variables
 - Typedefs
 - Operator overloading
 - Multiple inheritance
-

Introduction to Java

JDK

Java



Introduction to Java

Program Development steps

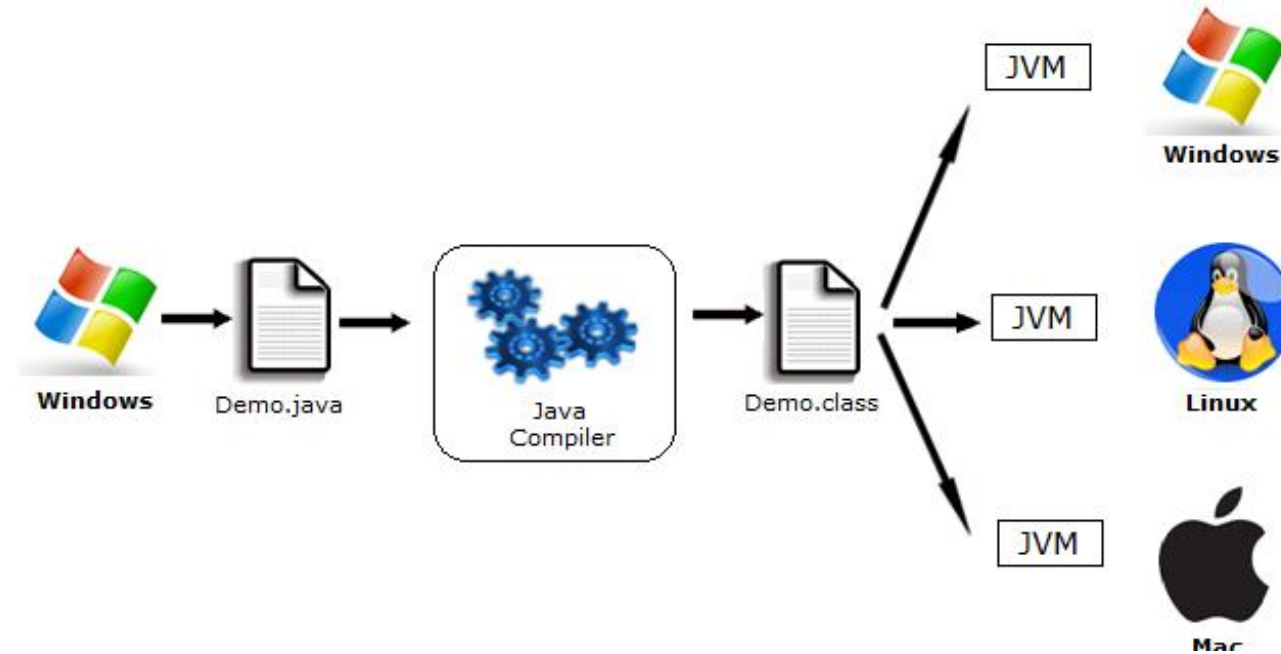
Java

- Write – The programmer is writing the code as text, using the programming language keywords and syntax.
 - Compile – Using a compiler program, the code is converted from text form to binary form so machines can run it.
 - Run – Executing the program.
-

Introduction to Java

Portability

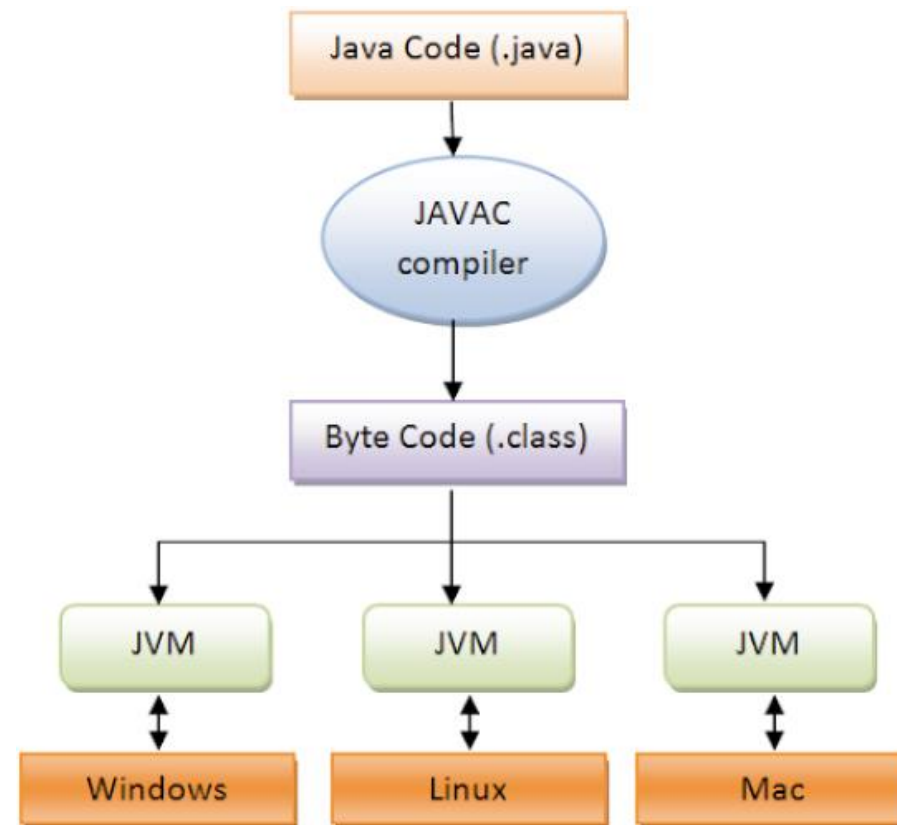
Java



Introduction to Java

Portability

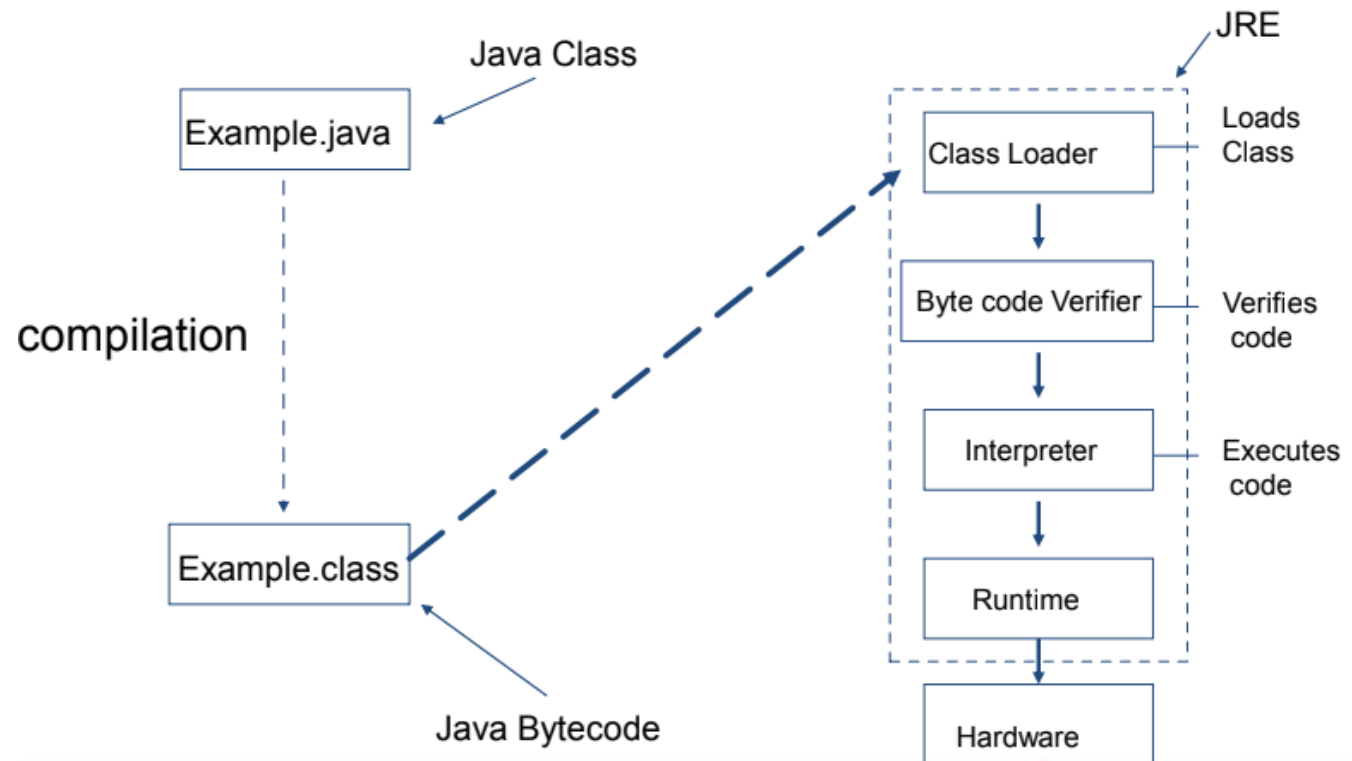
Java



Introduction to Java

JRE

Java



Introduction to Java

JRE

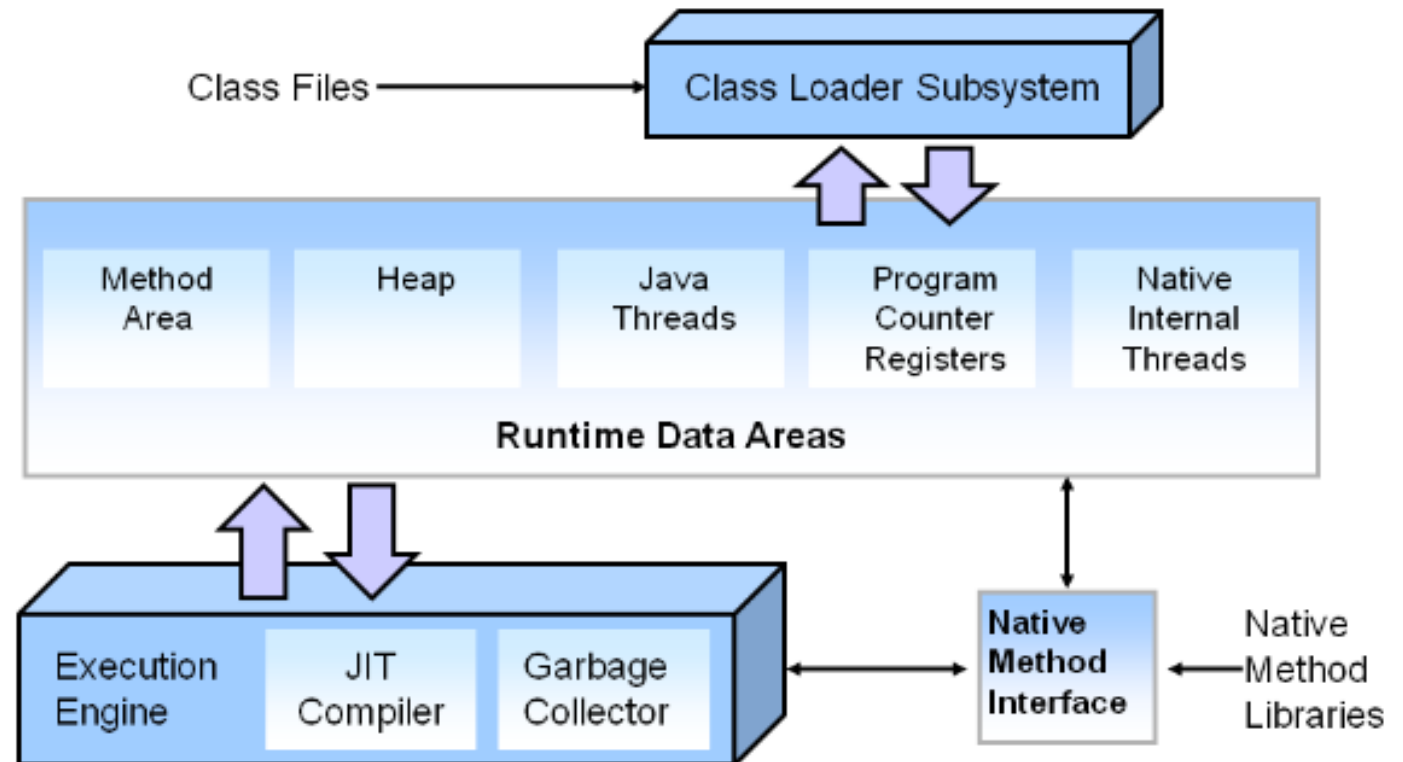
Java

- Class Loader - Loads all classes which are necessary for the execution of a program
 - Bytecode Verifier – verifies that class bytecode is legal and does not violate system integrity
 - Interpreter/JIT Compiler – Executes the code
-

Introduction to Java

JVM

Java

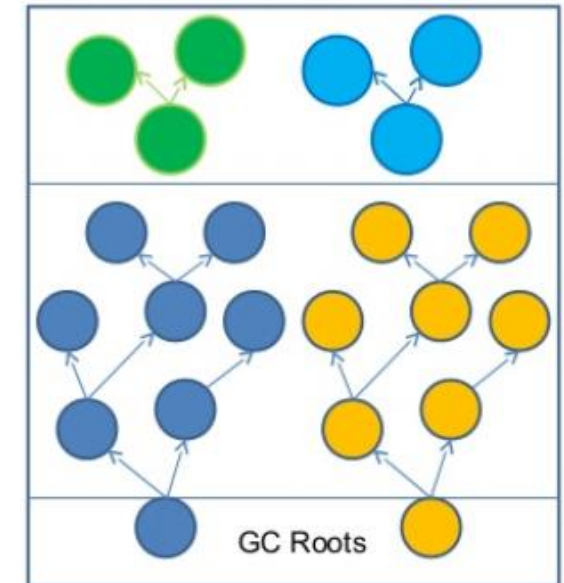


Introduction to Java

GC

Java

- The garbage collector frees all dynamically allocated memory that is no longer referenced
- An object located on the heap that is no longer referenced
- should be regarded as “garbage” and has to be removed
- Garbage collection is actually memory recycling



Introduction to Java

Java APIs

Java

Java runtime

JFC

Security

JDBC

JavaBeans

Java RMI

Java Communications

JavaMail

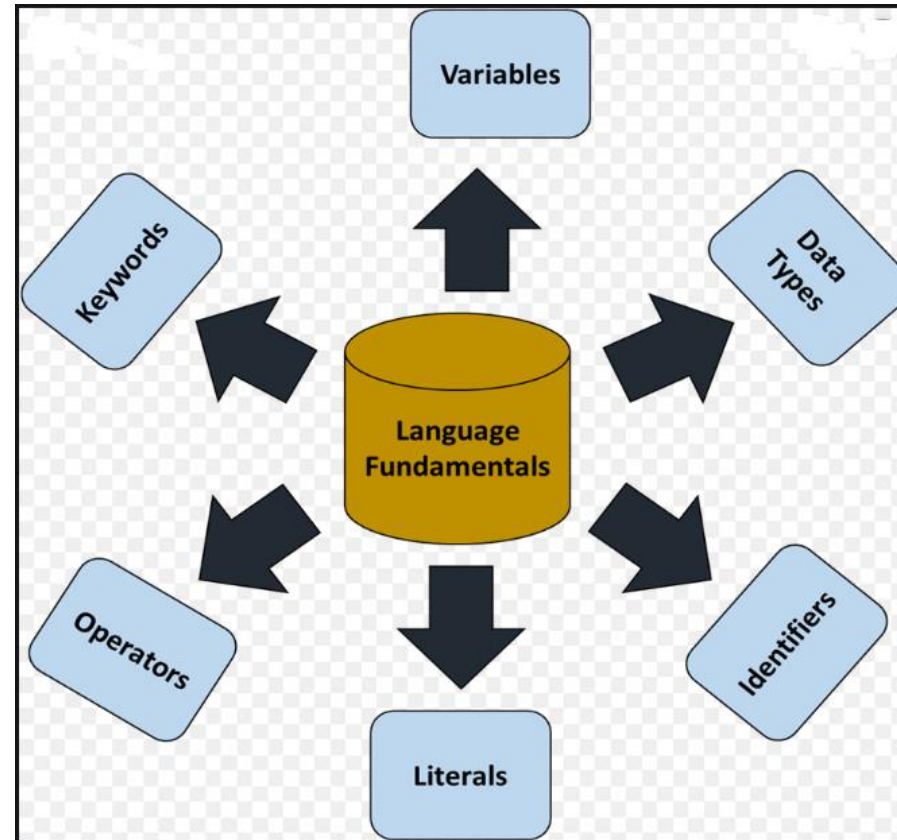
Java media

JNDI

Introduction to Java

Fundamentals

Java



Introduction to Java

Modifiers

Java

Access Specifiers in Java

		public	private	protected	default
Same Package	Class	YES	YES	YES	YES
	Sub class	YES	NO	YES	YES
	Non sub class	YES	NO	YES	YES
Different Package	Sub class	YES	NO	YES	NO
	Non sub class	YES	NO	NO	NO

Introduction to Java

Keywords

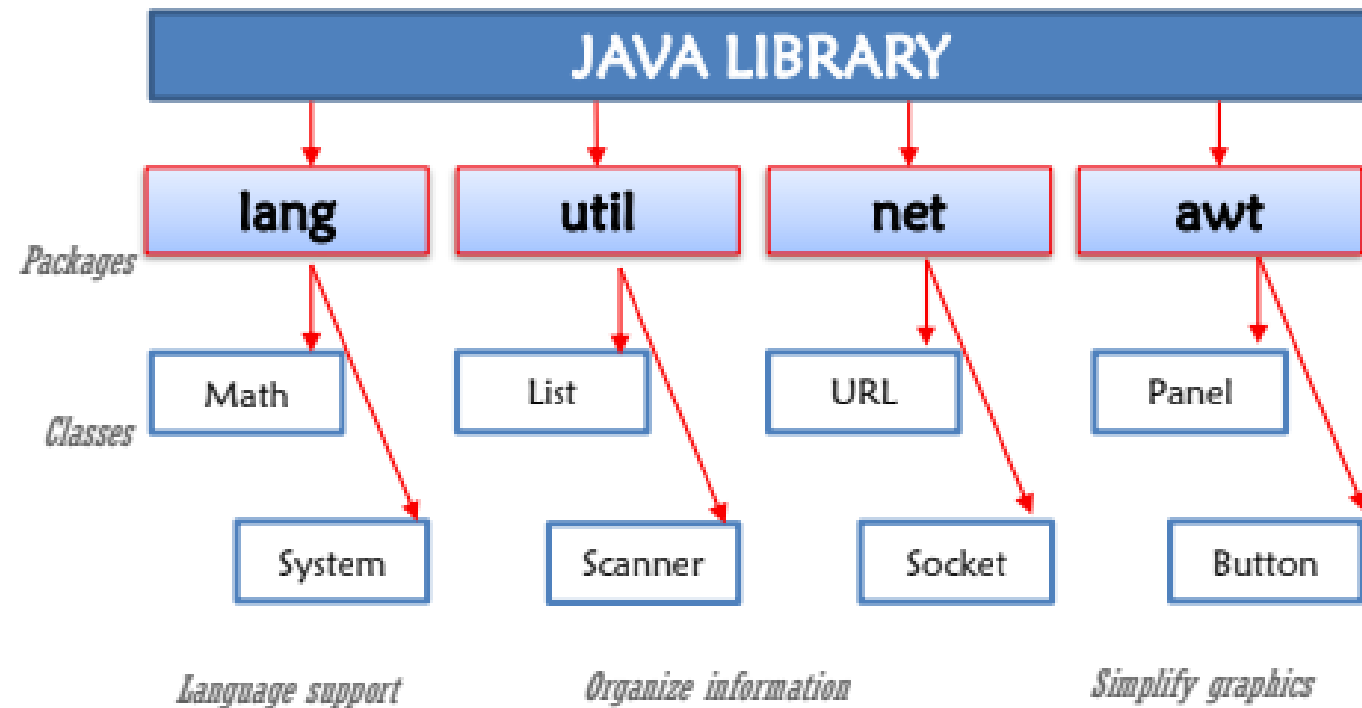
Java

abstarct	continue	for	new	switch
assert	default	goto	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

Introduction to Java

JAVA LIBS

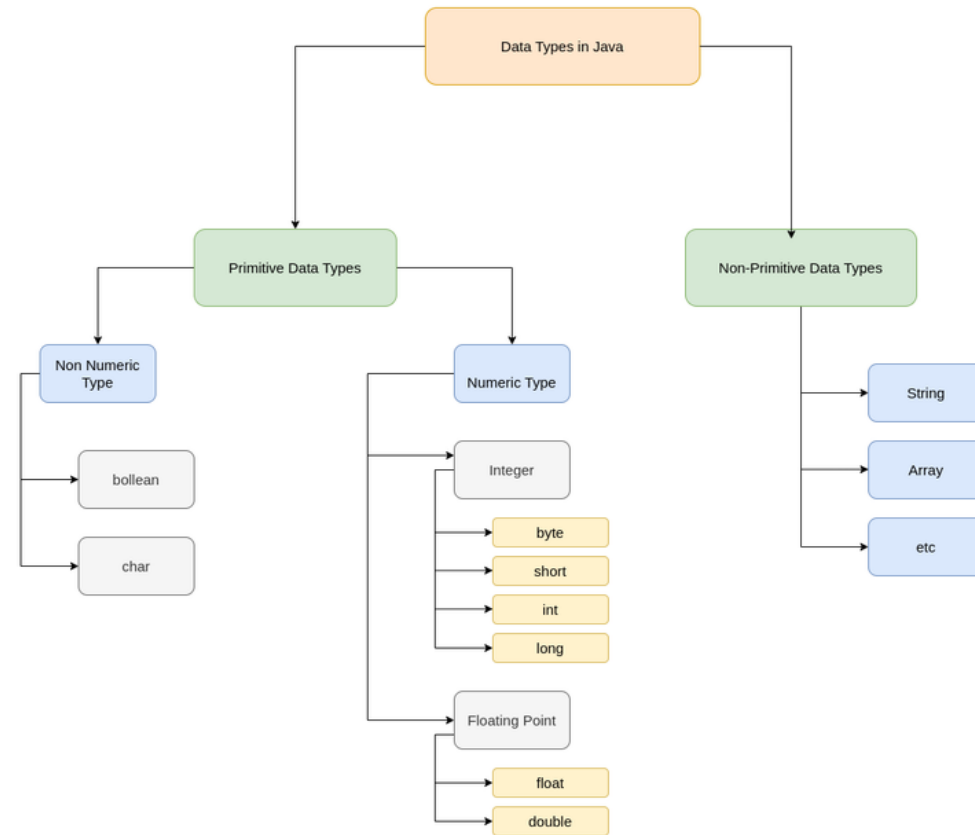
Java



Introduction to Java

Data Types

Java



Introduction to Java

Java

Components

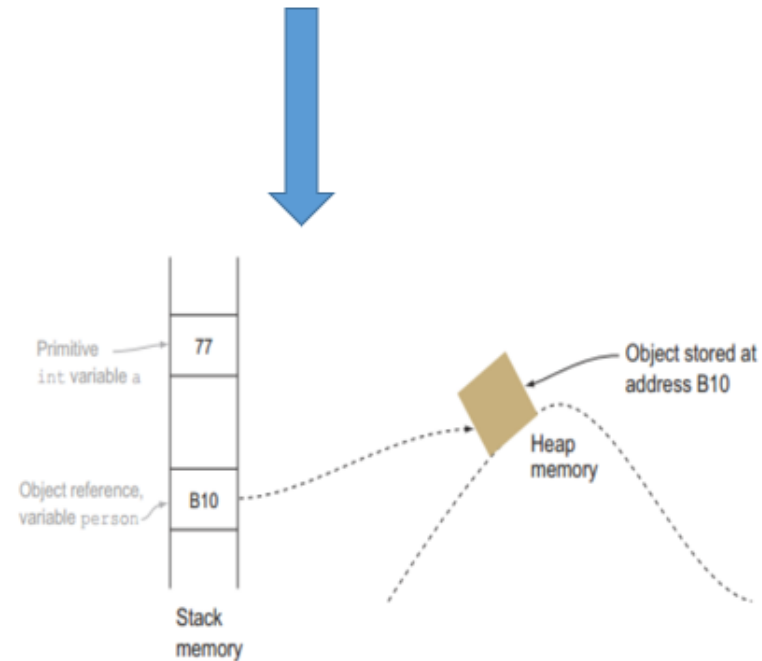
- **Class** : A blueprint source code for instantiating objects
 - **Object** : An instance of a class
 - **Attribute** : (Class Data Member, Data Field)
 - **Method** : (Class Function) A behavioral element of a class
 - **Constructor** : A method that is called whenever an object of class is instantiated.
 - **Package** : A grouping of classes (directory)
-

Introduction to Java

Memory

Java

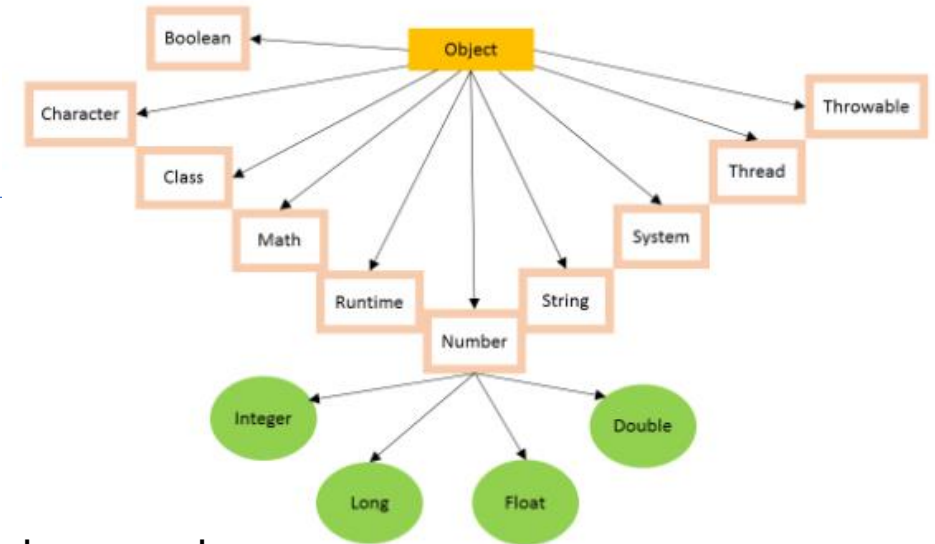
```
int a = 77;  
Person person = new Person();
```



Introduction to Java

Java

Object Class



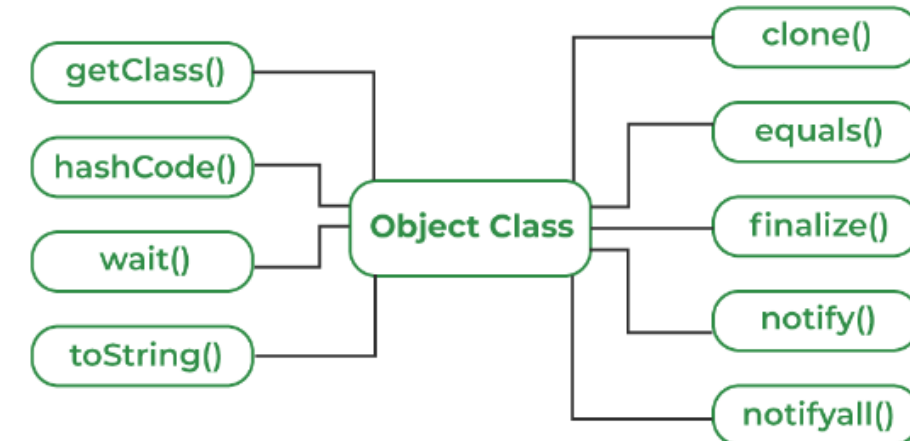
- Object class is present in java.lang package.
- Every class in Java is directly or indirectly derived from the Object class.
- Object class methods are available to all Java classes.
- Object class acts as a root of the inheritance hierarchy in any Java Program.

Introduction to Java

Object Class

The Object class provides multiple methods which are as follows:

- toString() method
- hashCode() method
- equals(Object obj) method
- finalize() method
- getClass() method
- clone() method
- wait(), notify() notifyAll() methods



Java

Introduction to Java

Java

Thank You !!
