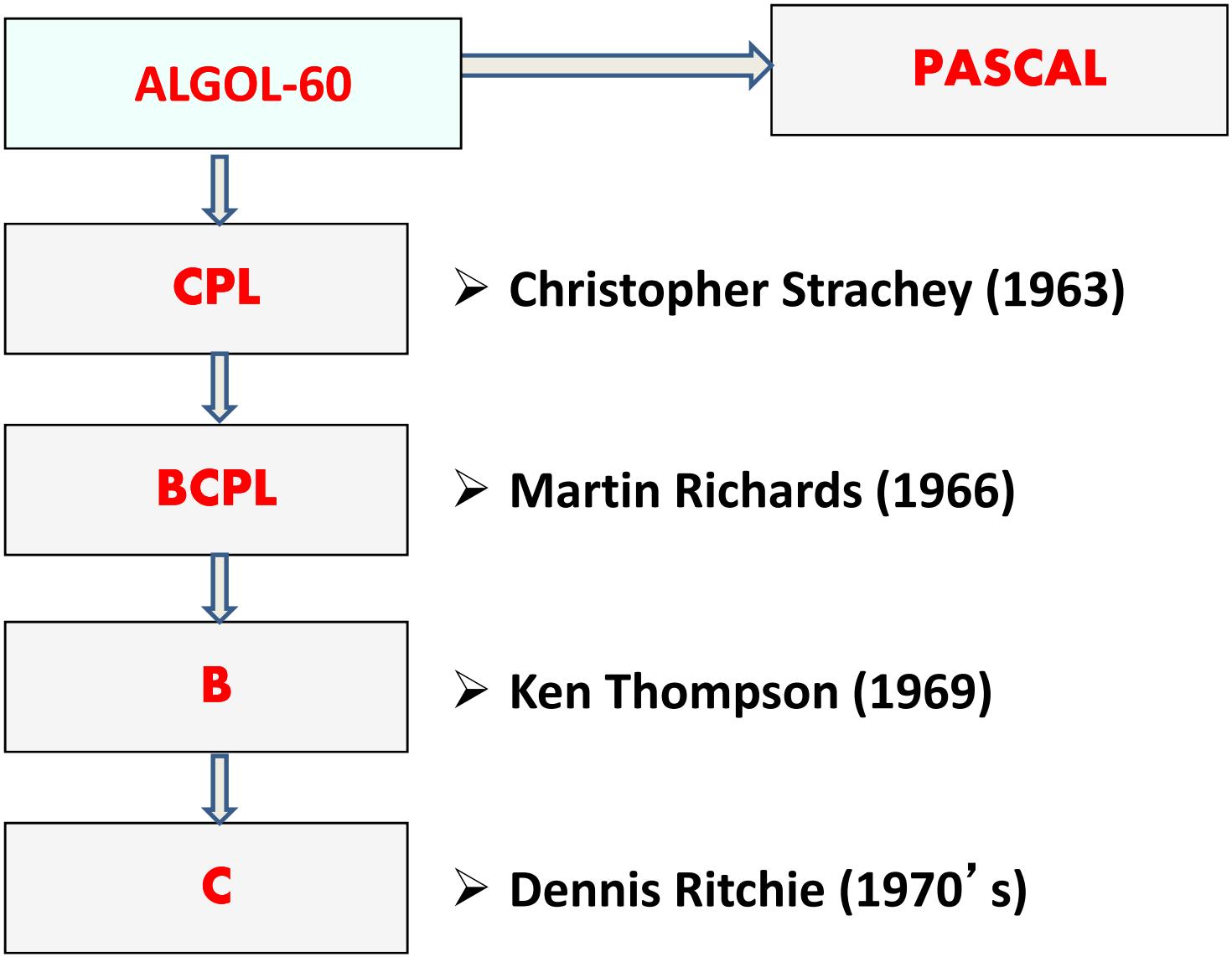
Introduction

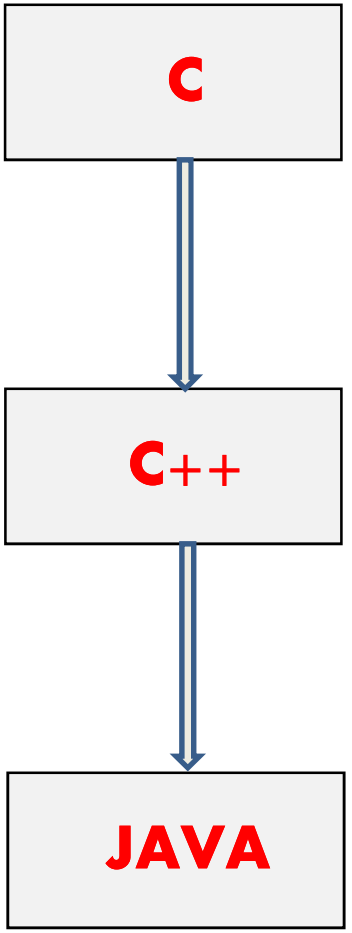
History of C..



- Christopher Strachey (1963)
- Martin Richards (1966)
- Ken Thompson (1969)
- Dennis Ritchie (1970's)

History of Java..

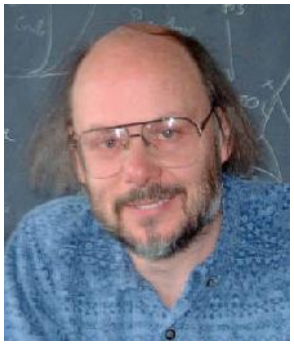
- **Dennis Ritchie**



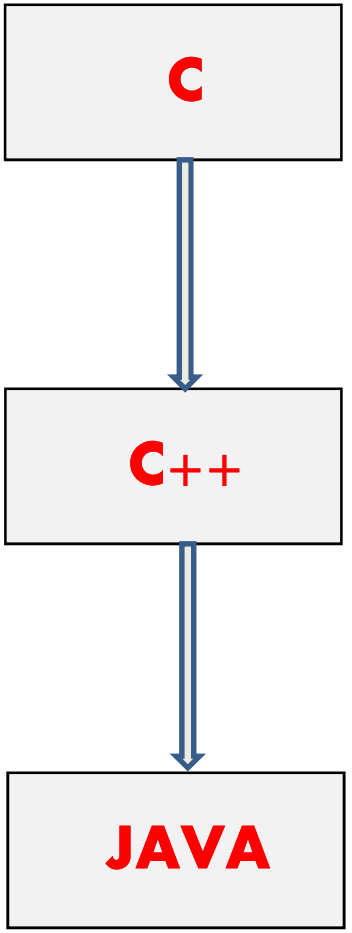
1970's Ruled by C ..

- **Increased complexity**
- **Structured programming only not enough ..**
- **Need for better ways to manage**

History of Java..



- **Bjarne Stroustrup**



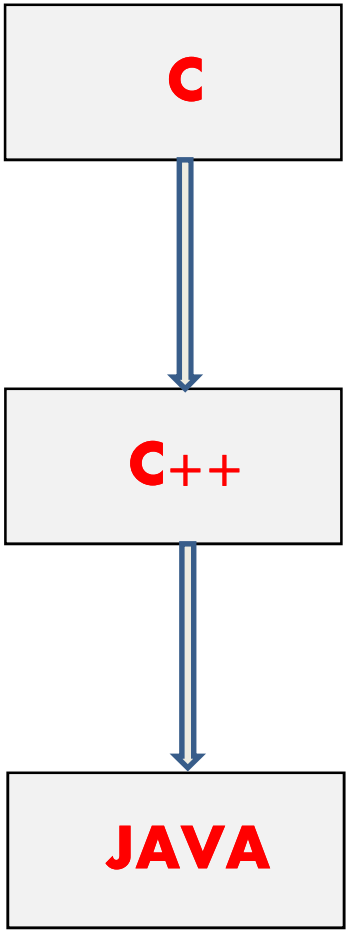
1980's Ruled by C++ ..

- **Object oriented and less complex.**
- **Require full compiler for new Architecture.**
- **Compilers are expensive and time consuming.**
- **Need Portable language**

History of Java..

James Gosling

- **James Gosling (1995)**



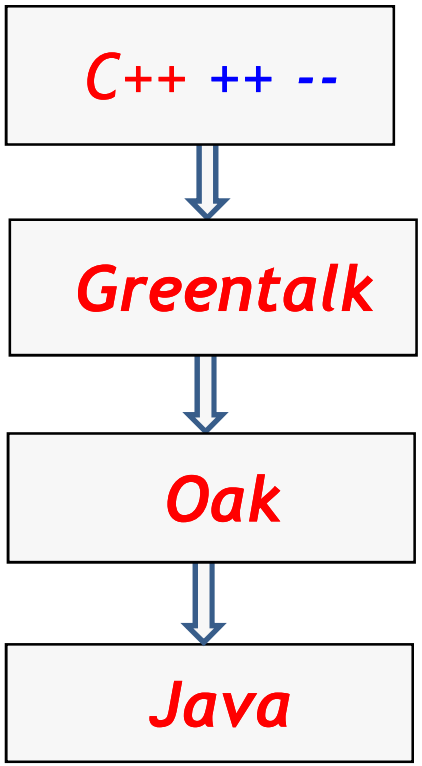
1990's Ruled by JAVA ..

- **Platform independent**
- **Best suited for Web**

History of Java..? 1990s

- **Green Team of Sun Microsystems:**

James Gosling, Mike Sheridan and Patrick Naughton,



Other contenders:

“Dynamic", “Revolutionary", "Silk", "jolt", "DNA”, etc.

Features of Java

1) Simple
2) Object-Oriented
3) Platform independent
4) Architecture neutral
5) Portable
6) Secured
7) Robust
8) Dynamic
9) Compiled and Interpreted
10) High Performance
11) Multithreaded
12) Distributed

Features of Java

1) Simple

- *Syntax is based on C++*
- *No pointers, No goto, No operator overloading, No Preprocessor, No global variables.*

2) Object-Oriented

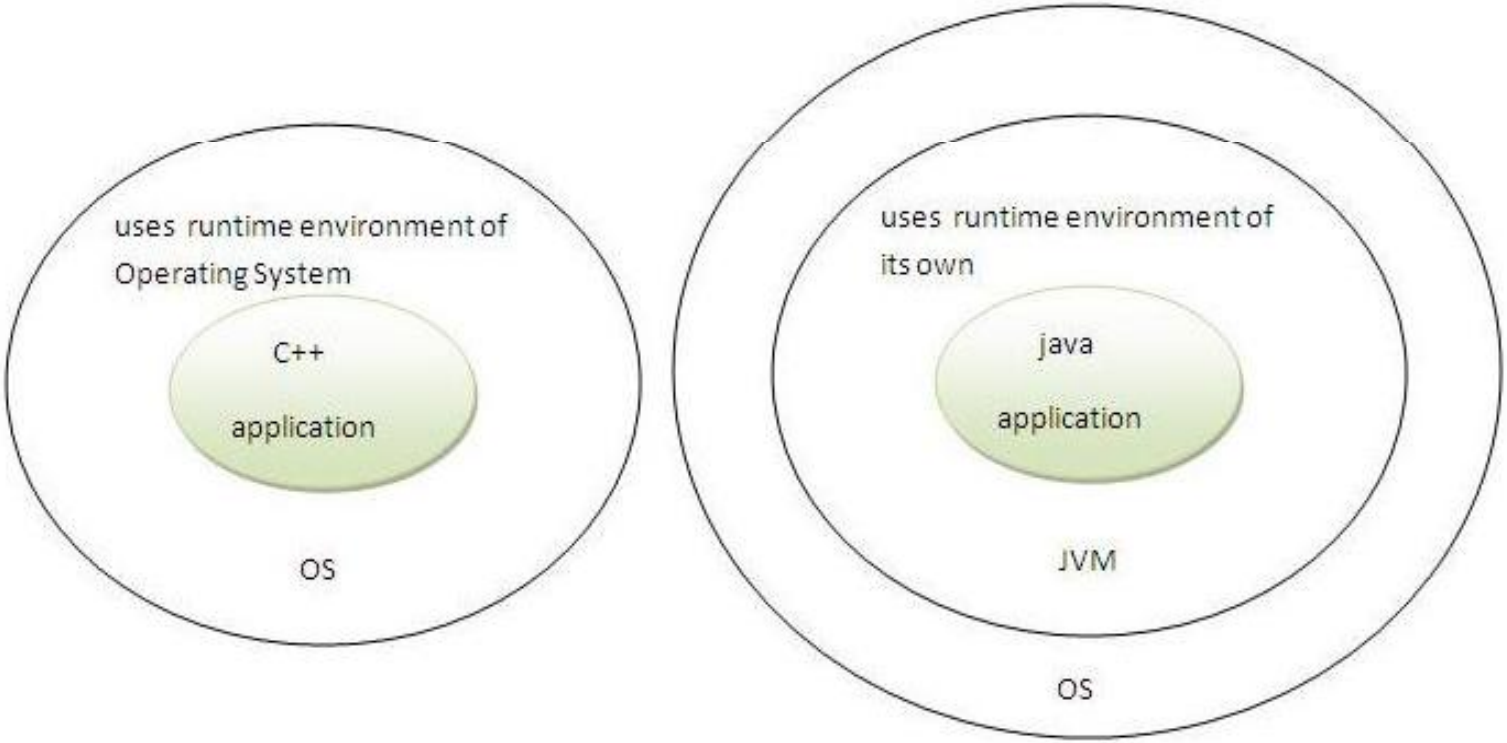
- *Uses OOPs concepts (Inheritance, Polymorphism, Encapsulation, etc)*

3) Portable - Platform Independent – Architecture Neutral

- *"Write once, Run anywhere"*
- *Runs on Any platform that has JVM*

Features of Java

4) Secured
   - ➢No explicit pointer
   - ➢Programs run inside JVM.



Features of Java

5) Robust
   - Does automatic garbage collection, which prevents memory leaks
   - Java is strictly typed language, hence Error free.

6) Dynamic
   - Java loads in classes as they are needed,
   - it is capable of linking dynamic new classes, methods and objects.

7) Compiled and Interpreted
   - Java code is compiled to bytecode.
   - Bytecode are interpreted on any platform by JVM.
   - Java programs can be shared over the internet

Features of Java

9) High Performance

- Bytecode are highly optimised.
- JVM execute Bytecode much faster

10) Multithreaded

- Multithreading means handling more than one job at a time.
- The main advantage of multi-threading is that it shares the same memory.

11) Distributed

- Java programs can be shared over the internet

The Key Attributes of Object-Oriented Programming

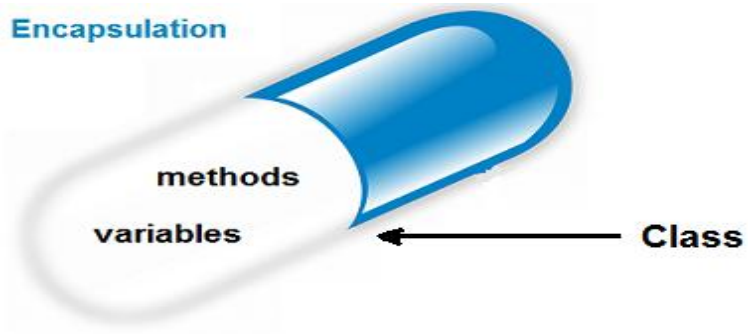
- Encapsulation
- Polymorphism
- Inheritance

The Key Attributes of Object-Oriented Programming

- Encapsulation

  "Binding the data with the code that manipulates it"

  - Why: to keep the data safe
  - *Real-time Example: Capsule*
  - Java Example for encapsulation : class




```
class Product
{
        int  product_id, Qty;
        String name;
        float  price;

      void readData() { ---- }

      float getPrice() { return price; }

      float getBill() { return Qty * price ; };
}
```

The Key Attributes of Object-Oriented Programming

- Encapsulation

Encapsulation is a practice to

bind related functionality (Methods) & Data (Variables)

in a protective wrapper (Class)

with required access modifiers (public, private, default & protected)

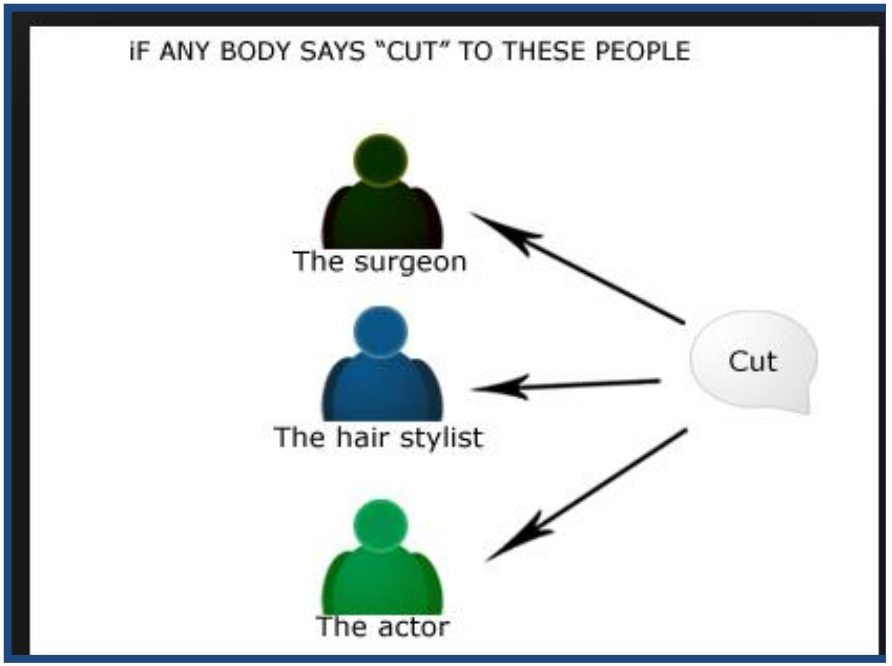
so that the code can be saved from unauthorized access by outer world.

The Key Attributes of Object-Oriented Programming

- **Polymorphism**

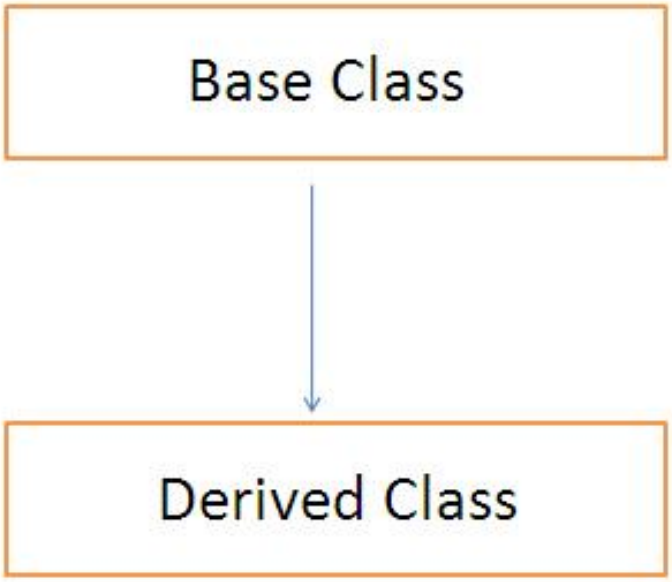
  "Same name many forms"

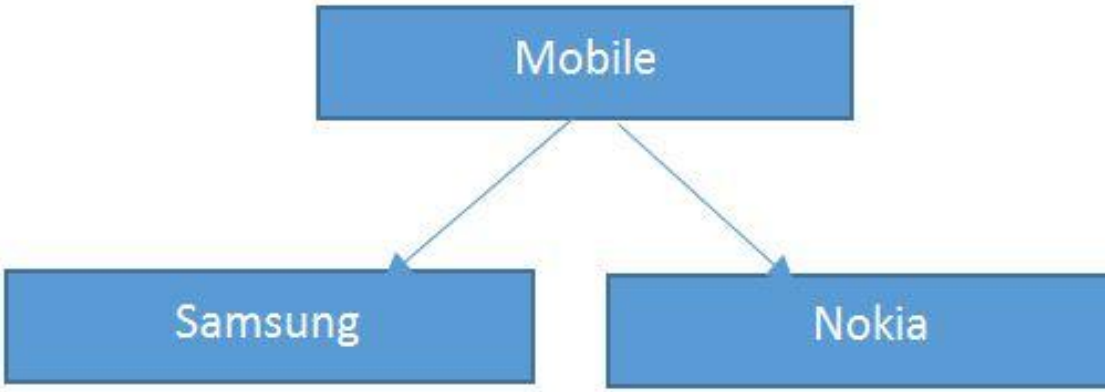
  - Why: to achive standrdization
  - Java Example for encapsulation
    - Method Overloading
    - Method Overriding



The Key Attributes of Object-Oriented Programming

- Inheritance
  - One object can Acquire the properties of another object





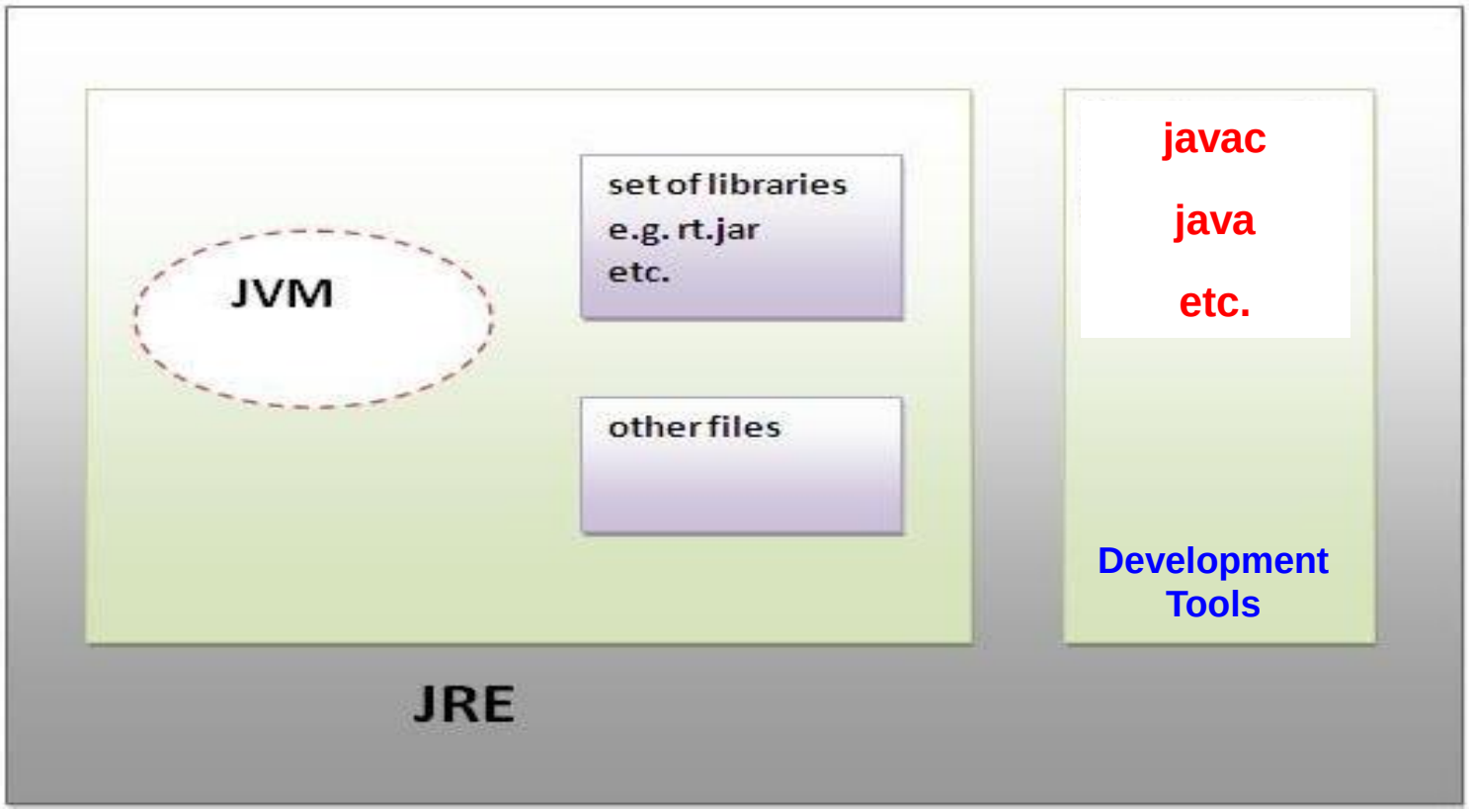
JDK

- JDK- The Java Development Kit
  - JDK is provided by Sun Microsystems
  - is a basic development environment for Java.
  - Provides environment for writing Java applets and applications.
  - JDK is collection of tools like..
    - ✓ *to compile - javac*
    - ✓ *To execute - java*
    - ✓ *To debug*
    - ✓ *To document, etc.*

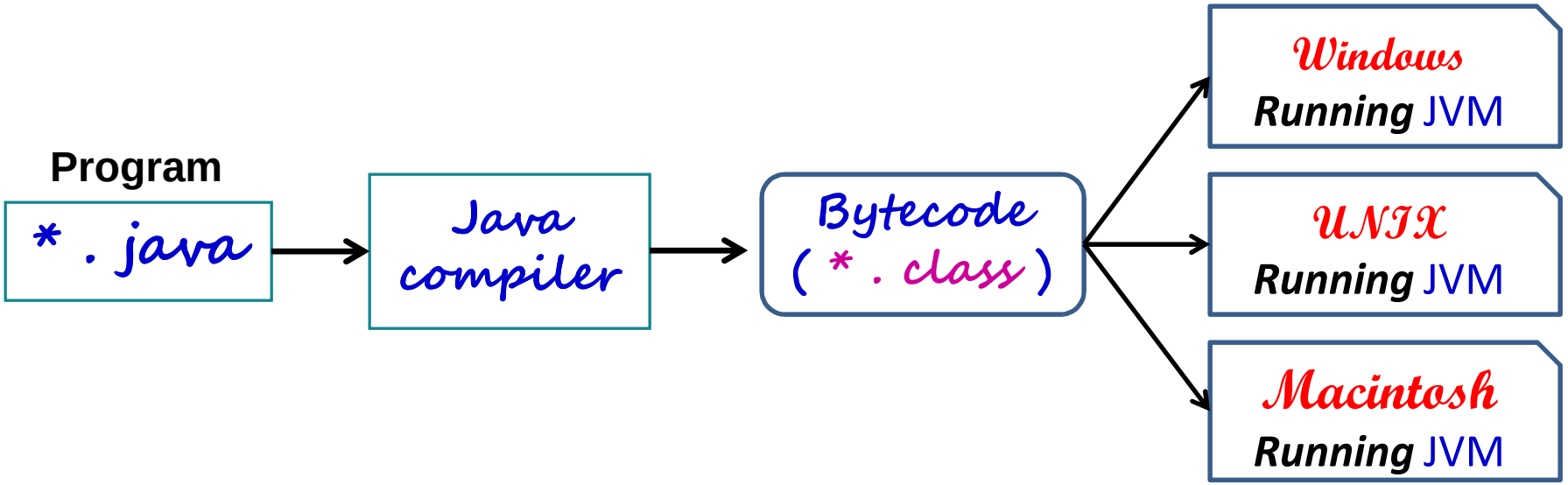
JDK, JRE and JVM

- Difference between JDK, JRE and JVM

JDK = JRE + Development Tools



Bytecode..



- **Bytecode is same no matter which computer platform.**
- **Bytecode translated into native code.**
- **Program can be executed on any computer that has the JVM.**
- **"Write once, run anywhere".**

Java programming fundamentals, Data types and operators

First Program

Hello.java

```
import  java.io.* ;
class  Hello {

     pu


     }
}
```

- **File name:**
  - *must match the class name containing main function.*
- **Program Code :**
  - All methods and code must reside inside a class.
  - *main method also reside inside a class.*

Hello.java

```
import  java.io.* ;
class  Hello {

    public static void main (String[] args) {
        System.out.println ("My First Program");
    }
}
```

- **public** : method can be called from anywhere (outside class).
- **static** : No object needed to call this method. (called by JVM)
- **void** : method has no return value.

Compiling and running..

Create Hello.java in directory

- Compiling..

  d:\yourname> *javac* Hello.java

- Running..

  d:\yourname> javaHello

Handling Syntax Errors

- The java compiler attempts to make sense out of your source code no matter what you have written.
- If you enter something incorrectly into your program, the compiler will report a syntax error message when it tries to compile it.

Keywords

- Keywords

List of Java Keywords

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

Although const and goto are reserved words, they are not currently part of the Java language.

Identifiers

❑ Identifiers

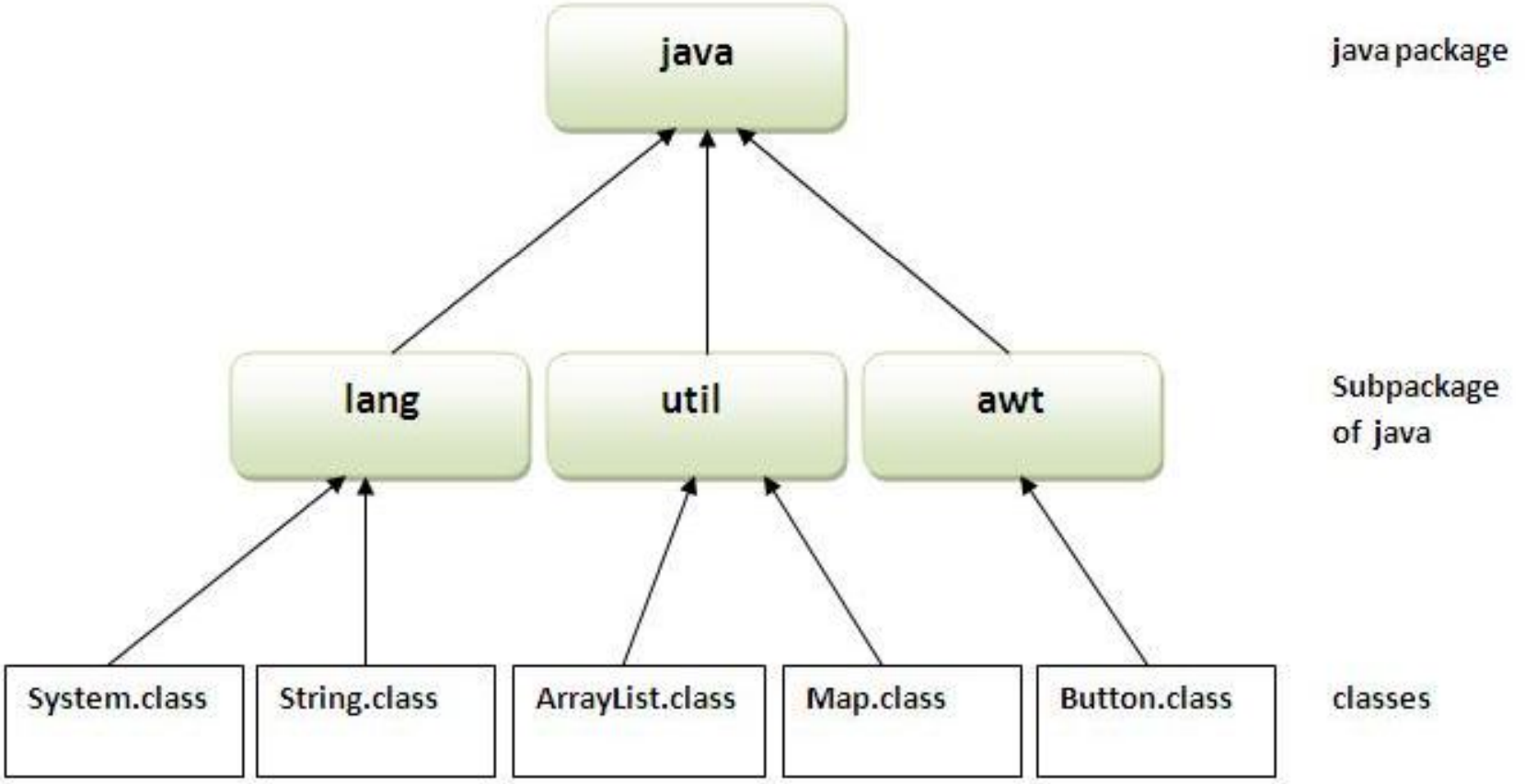
- An identifier is a name given to anything that you want to identify in program. like
  - ➢ *a package, class, interface, method, or variable.*

- **Rules…**
  - ✓ *Can start with* the underscore (_) or dollar sign ($).
  - ✓ *reserved words* cannot be used.
  - ✓ *cannot start with* a digit but digits can be used after first character.
  - ✓ are case-sensitive and have no maximum length
    - ➢ VALUE is a different from Value.

Java class libraries

- A **java package** is a group of similar types of classes, interfaces and sub-packages.
- Package in java can be categorized in two form, built-in package and user-defined package.
- There are many built-in packages such as java, **lang, awt, javax, swing, net, io, util, sql etc.**

Contd....


java
java package
lang
util
awt
Subpackage
of java
System.class
String.class
ArrayList.class
Map.class
Button.class
classes