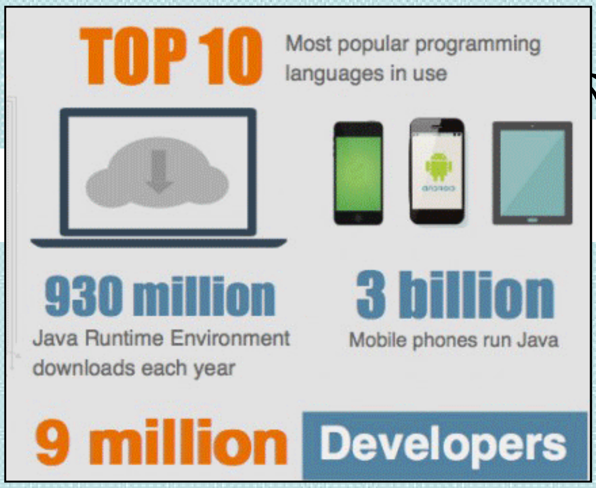


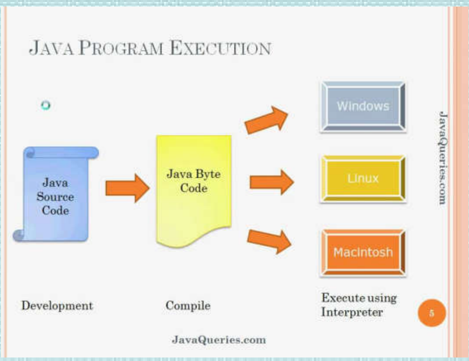
TOP 10 Most popular programming languages in use

930 million Java Runtime Environment downloads each year

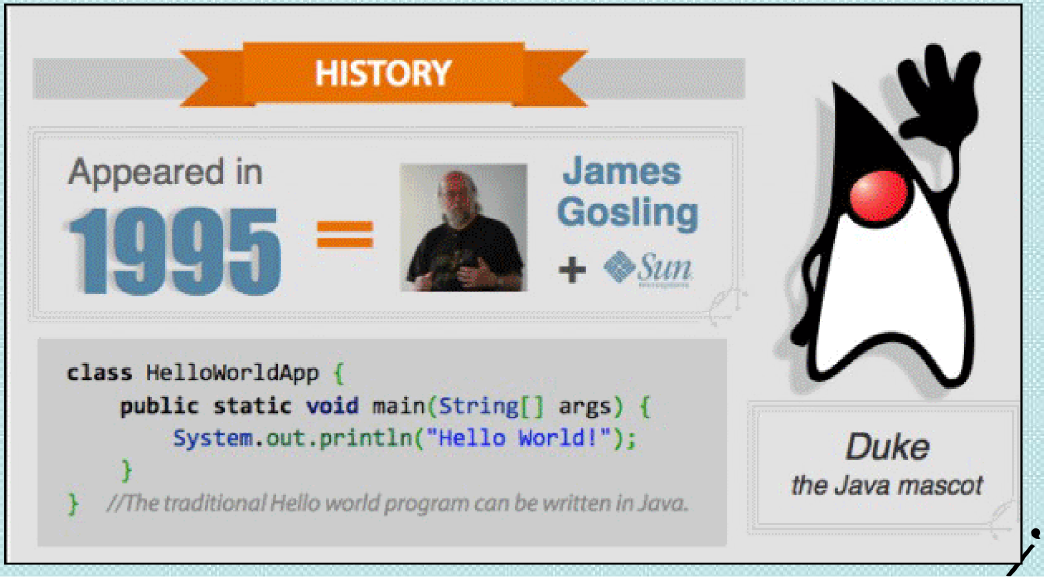
3 billion Mobile phones run Java

9 million Developers

Тема 1.1 Введение в программирование на языке *JAVA*. История



JAVA PROGRAM EXECUTION



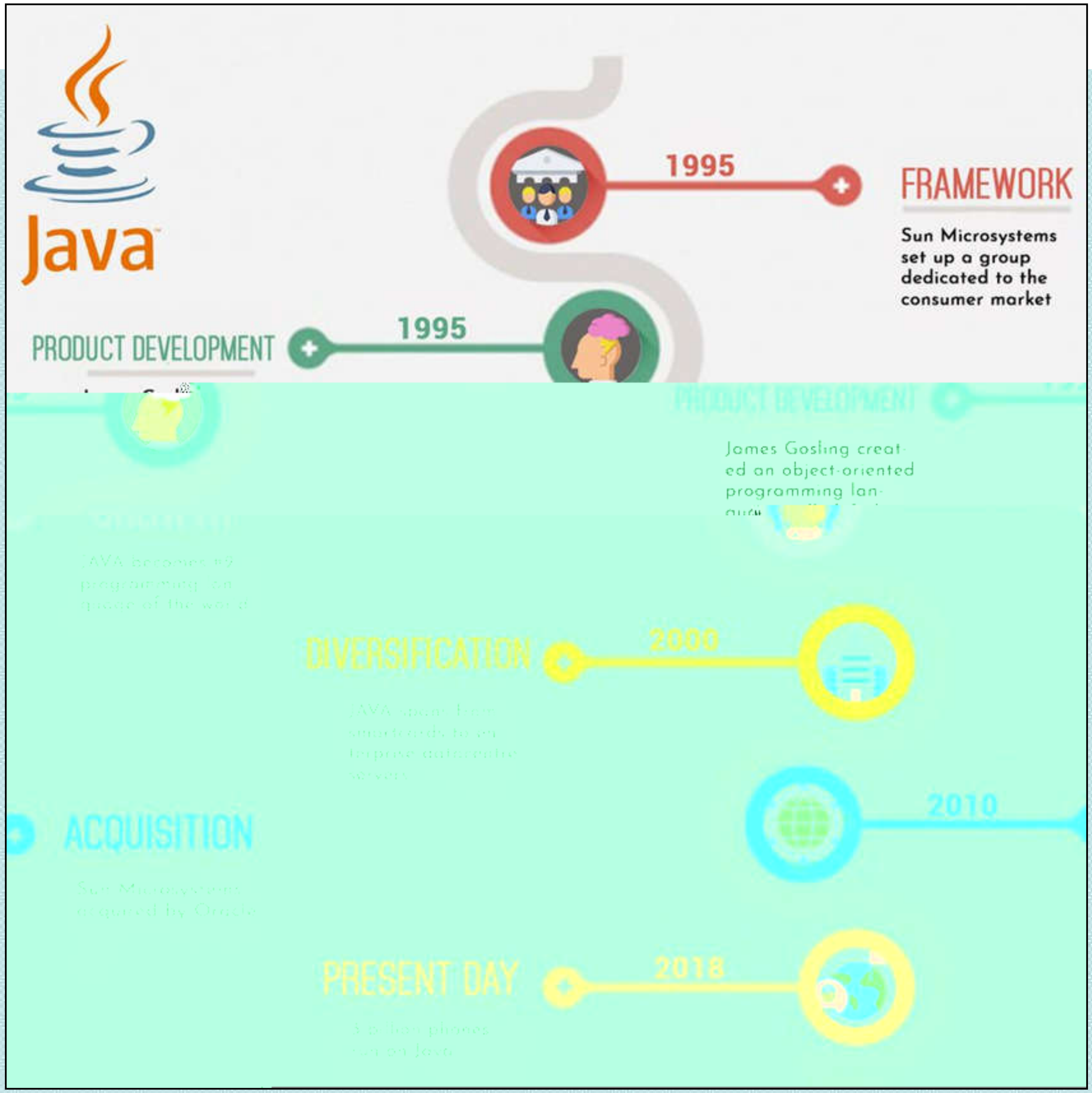
HISTORY

Appeared in 1995 = James Gosling + Sun

```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
} //The traditional Hello world program can be written in Java.
```

Duke
the Java mascot

HISTORY


Java
1995
FRAMEWORK
Sun Microsystems set up a group dedicated to the consumer market
PRODUCT DEVELOPMENT
1995
PRODUCT DEVELOPMENT
James Gosling created an object-oriented programming language
JAVA becomes #9 programming language of the world
DIVERSIFICATION
2000
JAVA spans from smartcards to enterprise automotive servers
ACQUISITION
2010
Sun Microsystems acquired by Oracle
PRESENT DAY
2018
3 billion phones run on Java

JDK 1.1 · (Feb 19, 1997)

- JavaBeans
- Improved AWT
- JDBC, RMI, Reflection
- Inner classes
- JIT, for Windows only (by Symantec)

J2SE 1.2 · Playground · (Dec 8, 1998)

- J2SE, J2EE, J2ME
- 3x · 1520 classes in 59 packages
- Sun's JIT compiler
- Collections framework
- Integrated Swing API
- `strictfp` keyword
- Java plug-in
- Java IDL/for CORBA

HISTORY

J2SE 1.3 · Kestrel · (May 8, 2000)

- HotSpot JVM
- Synthetic (Dynamic) proxy classes
- JNDI included
- Debugger Architecture (JPDA)
- RMI + CORBA
- JavaSound

J2SE 1.4 · Merlin · (Feb 6, 2002)

- JCP · JSR 59
- `assert` keyword
- Exception Chaining
- RegEx
- NIO · IPv6 · Logging
- Image API
- JAXP
- JCE · JSSE · JAAS
- Java Web Start
- Preferences API

J2SE 5.0 · Tiger · (Sep 30, 2004)

- Generics
- @Annotations
- Autoboxing
- `enum` keyword
- Varargs
- `for each` loop
- Static imports
- Mem Model Fix
- RMI auto stubs
- java.util.concurrent

Java SE 6 · Mustang · (Dec 11, 2006)

- Perfor

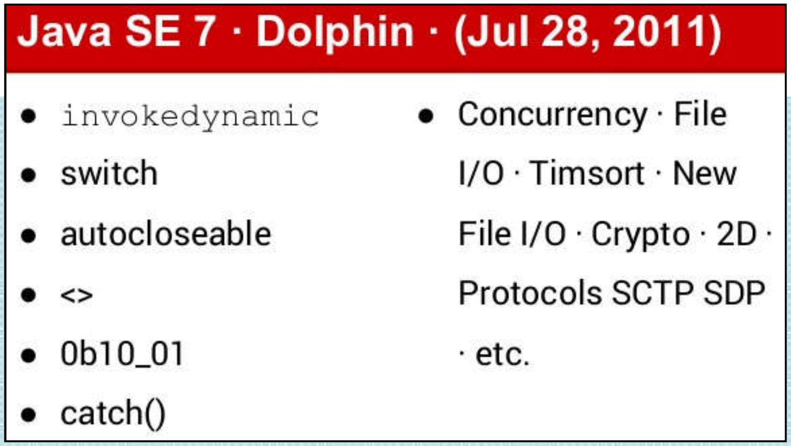
e impr. • JAX-WS • Performance

XB 2.0 · StAX • JVM/GC im

ggable annotations Support • Plu

/projectlombok.org/) • Java

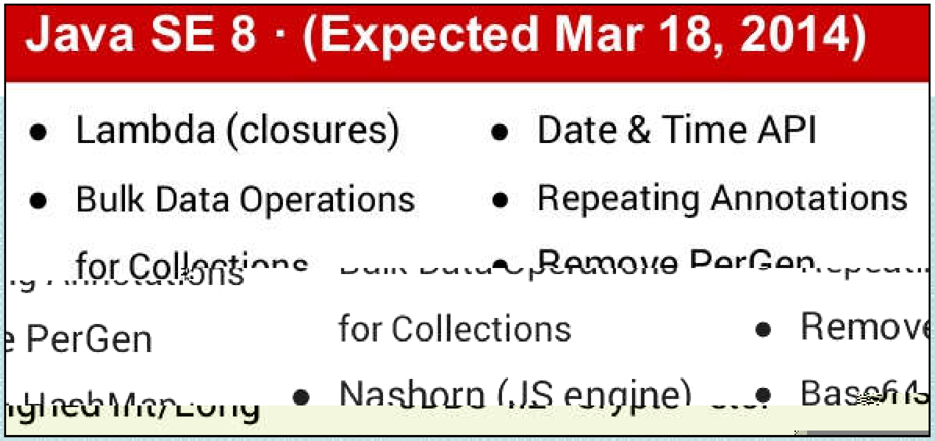
HISTORY



Java SE 7 · Dolphin · (Jul 28, 2011)

- `invokedynamic`
- switch
- autocloseable
- <>
- 0b10_01
- catch()

- Concurrency · File I/O · Timsort · New File I/O · Crypto · 2D · Protocols SCTP SDP · etc.



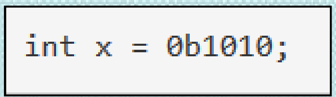
Java SE 8 · (Expected Mar 18, 2014)

- Lambda (closures)
- Bulk Data Operations for Collections
- Date & Time API
- Repeating Annotations
- Remove PerGen

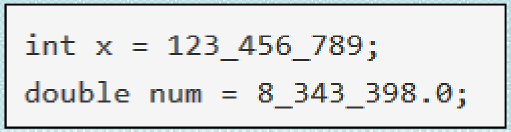
e PerGen

for Collections

- Remove
- Nashorn (JS engine)
- Base64



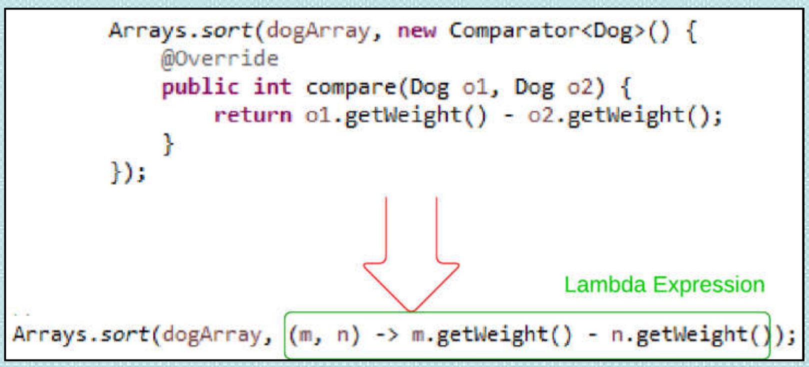
```
int x = 0b1010;
```



```
int x = 123_456_789;
double num = 8_343_398.0;
```

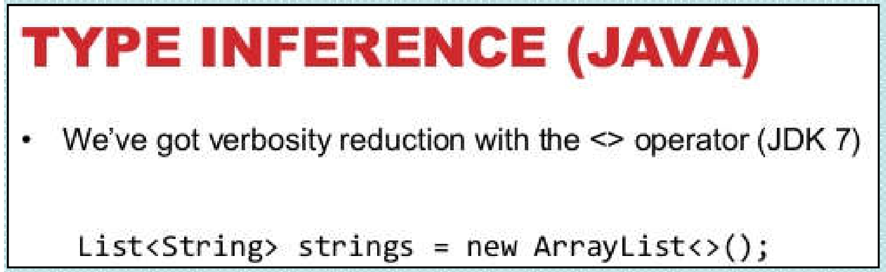
Exception Handling

Multi-catch Block



```
Arrays.sort(dogArray, new Comparator<Dog>() {
    @Override
    public int compare(Dog o1, Dog o2) {
        return o1.getWeight() - o2.getWeight();
    }
});
```

```
Arrays.sort(dogArray, (m, n) -> m.getWeight() - n.getWeight());
```

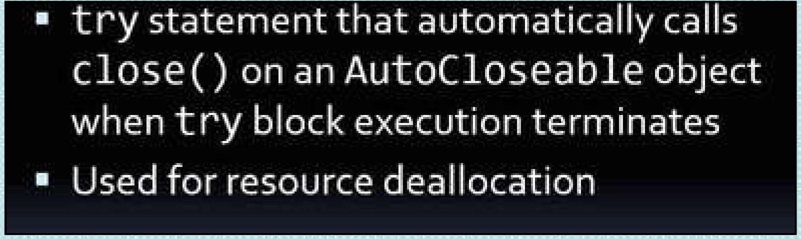


TYPE INFERENCE (JAVA)

- We've got verbosity reduction with the <> operator (JDK 7)

```
List<String> strings = new ArrayList<>();
```

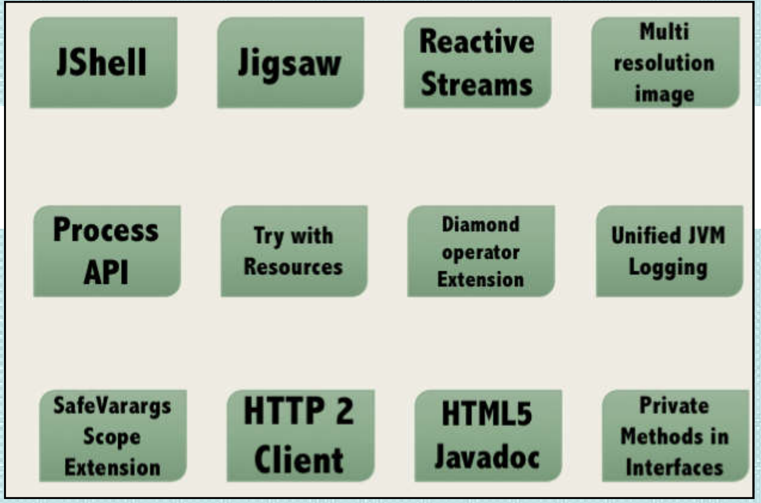
Interface Default Method



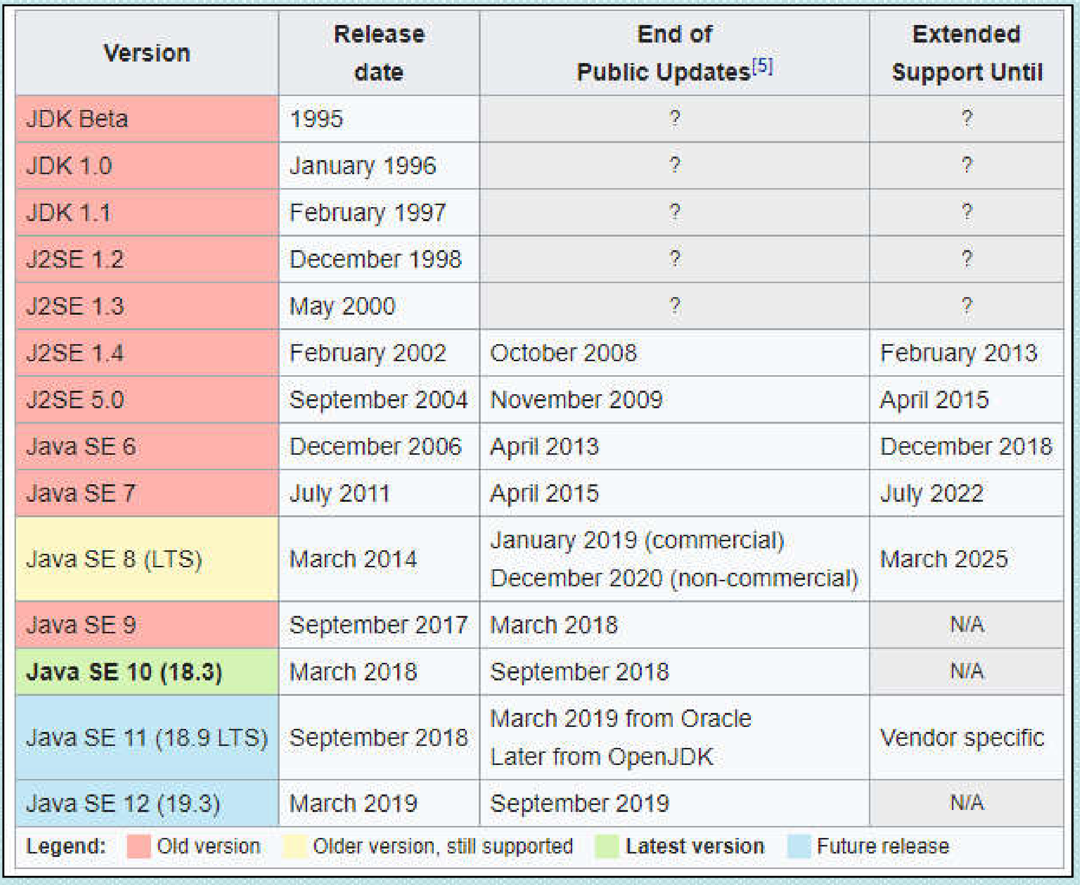
- `try` statement that automatically calls `close()` on an `AutoCloseable` object when `try` block execution terminates
- Used for resource deallocation

JavaFx

HISTORY

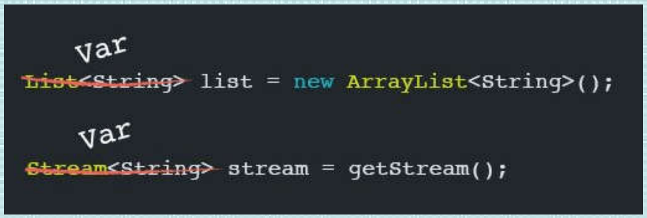


- JShell
- Jigsaw
- Reactive Streams
- Multi resolution image
- Process API
- Try with Resources
- Diamond operator Extension
- Unified JVM Logging
- SafeVarargs Scope Extension
- HTTP 2 Client
- HTML5 Javadoc
- Private Methods in Interfaces

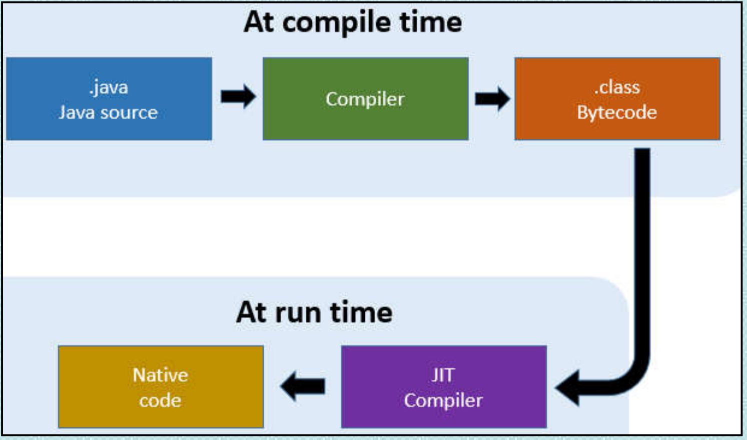


Version	Release date	End of Public Updates[5]	Extended Support Until
JDK Beta	1995	?	?
JDK 1.0	January 1996	?	?
JDK 1.1	February 1997	?	?
J2SE 1.2	December 1998	?	?
J2SE 1.3	May 2000	?	?
J2SE 1.4	February 2002	October 2008	February 2013
J2SE 5.0	September 2004	November 2009	April 2015
Java SE 6	December 2006	April 2013	December 2018
Java SE 7	July 2011	April 2015	July 2022
Java SE 8 (LTS)	March 2014	January 2019 (commercial) December 2020 (non-commercial)	March 2025
Java SE 9	September 2017	March 2018	N/A
Java SE 10 (18.3)	March 2018	September 2018	N/A
Java SE 11 (18.9 LTS)	September 2018	March 2019 from Oracle Later from OpenJDK	Vendor specific
Java SE 12 (19.3)	March 2019	September 2019	N/A

Legend: 🟥 Old version 🟨 Older version, still supported 🟩 **Latest version** 🟦 Future release



```
  var
~~List<String>~~ list = new ArrayList<String>();

  var
~~Stream<String>~~ stream = getStream();
```



At compile time

.java
Java source

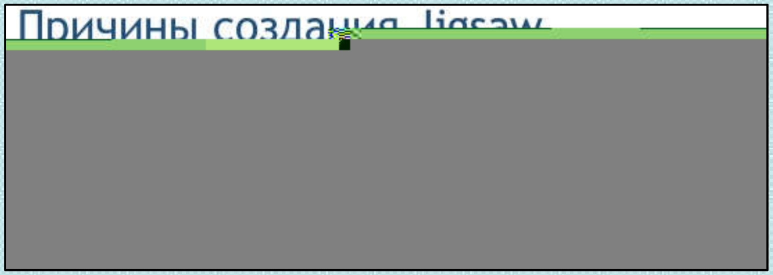
Compiler

.class
Bytecode

At run time

Native
code

JIT
Compiler

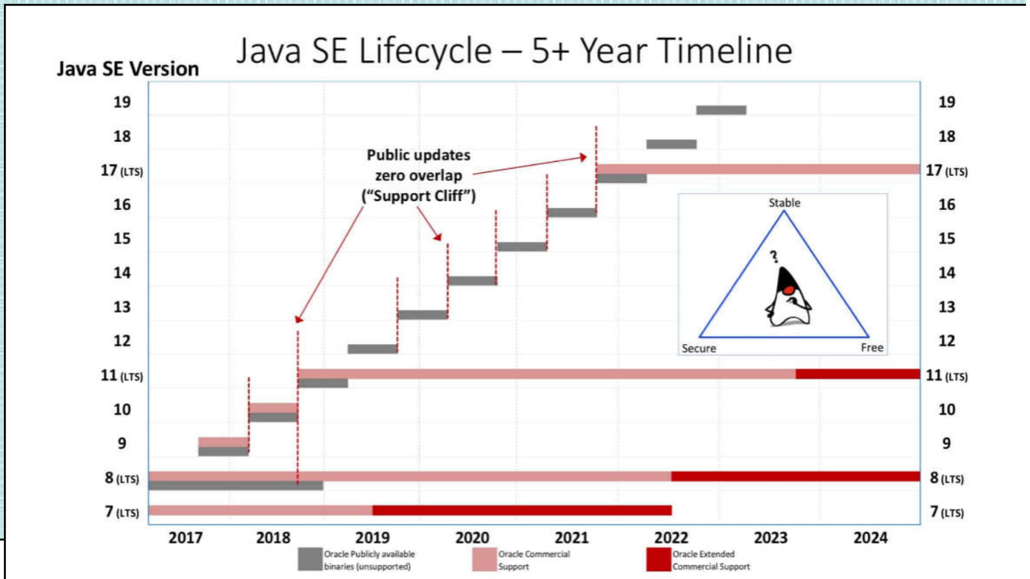


PRESENT DAY

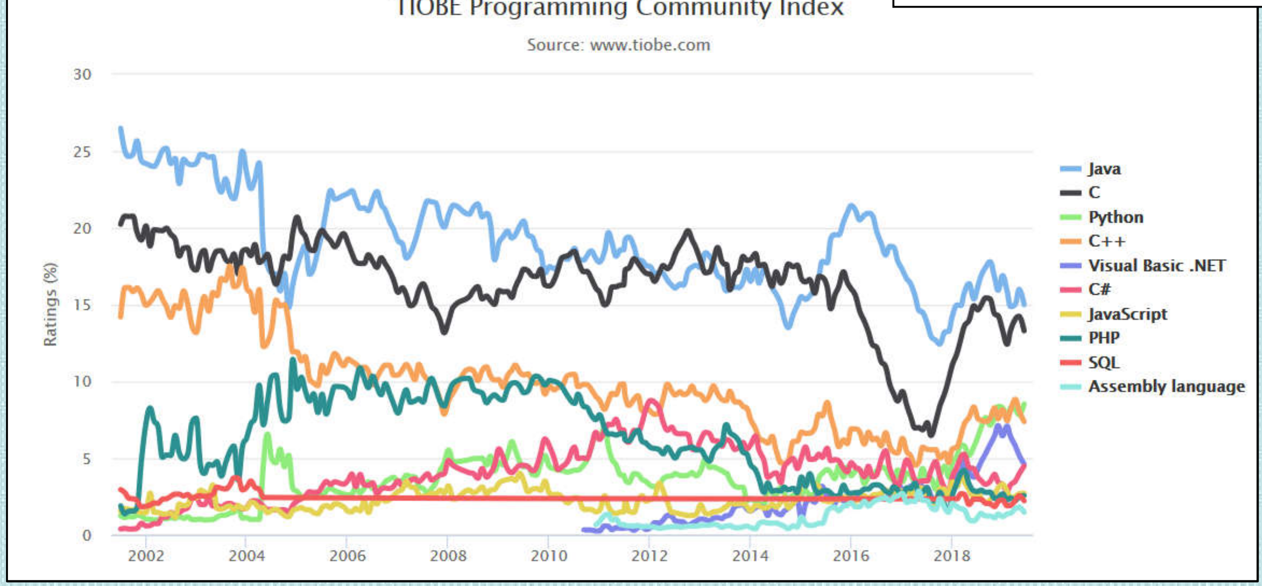
JAVA SE 12 , MARCH 2019

Java SE Downloads

Java Platform (JDK) 12



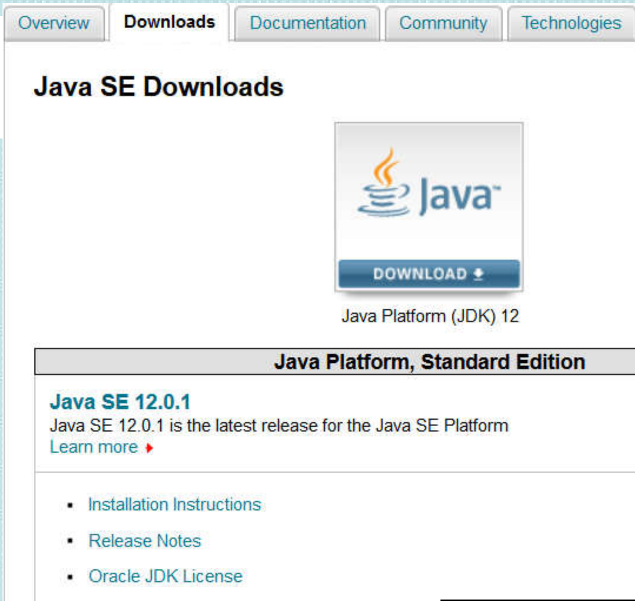
Java SE Lifecycle – 5+ Year Timeline



TIOBE Programming Community Index

Source: www.tiobe.com

НЕОБХОДИМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ



Java SE Downloads

Java Platform (JDK) 12

Java Platform, Standard Edition

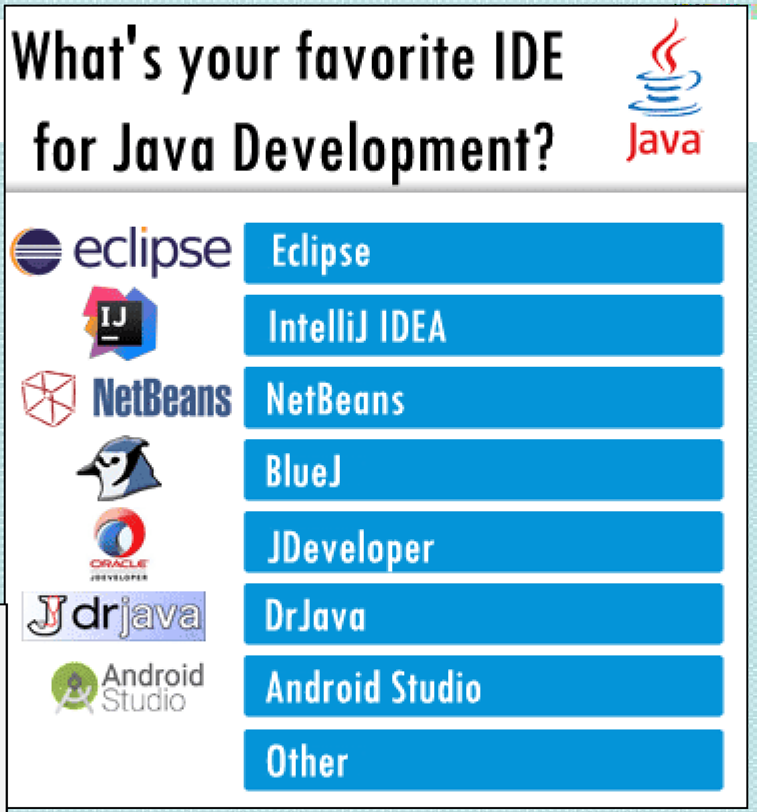
Java SE 12.0.1

Java SE 12.0.1 is the latest release for the Java SE Platform

Learn more

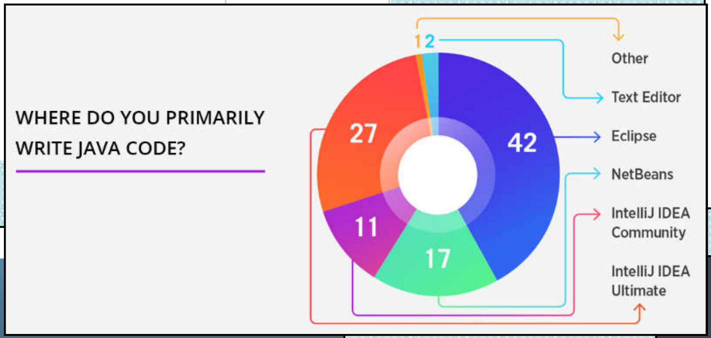
- Installation Instructions
- Release Notes
- Oracle JDK License
- Java SE Licensing Information User Manual
  - Includes Third Party Licenses
- Certified System Configurations
- Readme

Oracle JDK



What's your favorite IDE for Java Development?

- Eclipse
- IntelliJ IDEA
- NetBeans
- BlueJ
- JDeveloper
- DrJava
- Android Studio
- Other



WHERE DO YOU PRIMARILY WRITE JAVA CODE?

Java Setup - Progress

Status: Installing Java

ATMs, Smartcards, POS Terminals, Blu-ray Players, PCs
Set Top Boxes, Mobile Phones, Printers, Servers, Switches
Routers, Lottery Devices
Automobiles
Systems, Access Control Systems, Building Controls
3 Billion
Devices Run Java
Java
#1 Development Platform
ORACLE

JAVA BEST IDE'S

It's Write & Debug Java Program with easy

Java IDE or Integrated Development Environment is a software

pment Environment is a software

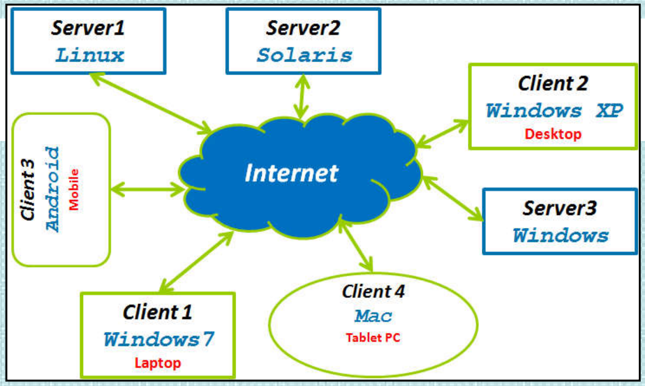
Java IDE or Integrated Develo

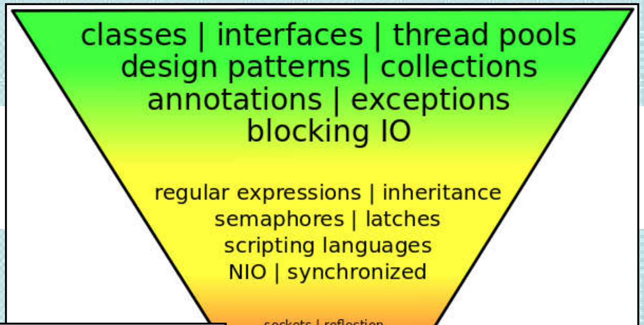
lopers to write and debug ease. A good IDE will have a number of featu

application. It allows deve

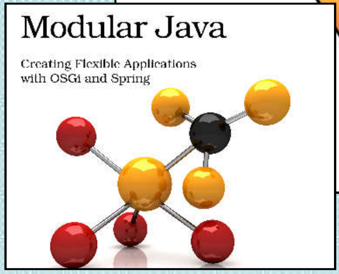
that help developers to write codes easily.

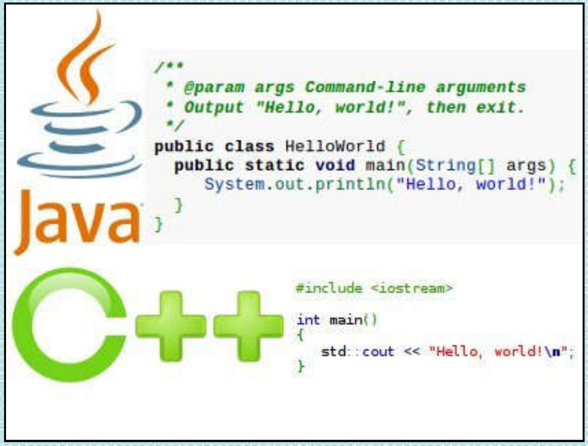
ОСНОВЫ JAVA





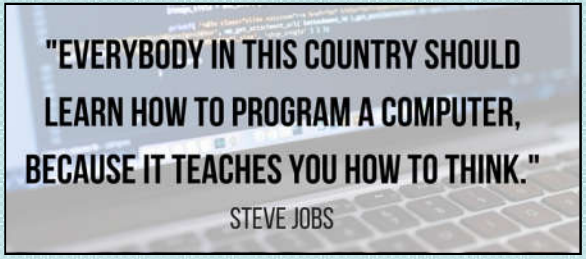
Java father
James Arthur Gosling





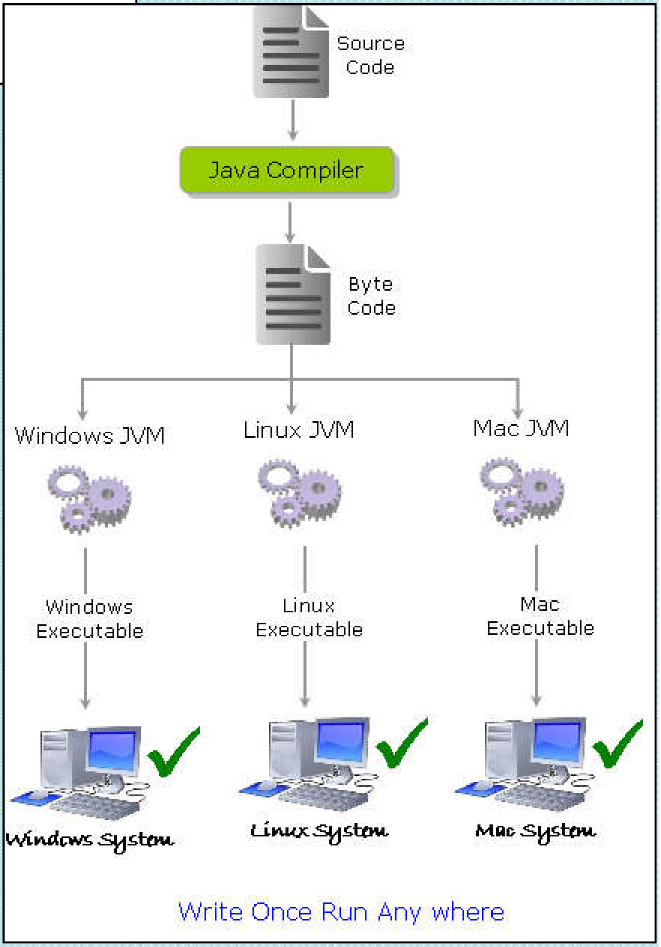
```
/**
 * @param args Command-line arguments
 * Output "Hello, world!", then exit.
 */
public class HelloWorld {
  public static void main(String[] args) {
      System.out.println("Hello, world!");
  }
}
```

```
#include <iostream>

int main()
{
   std::cout << "Hello, world!\n";
}
```

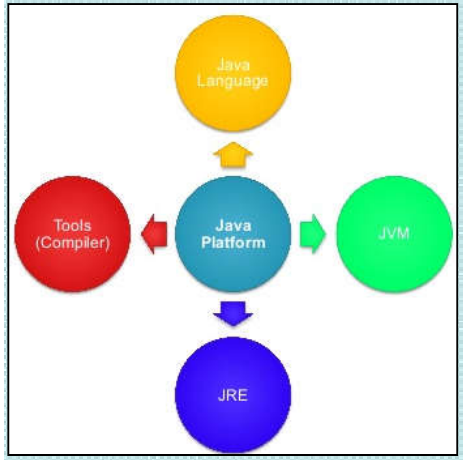


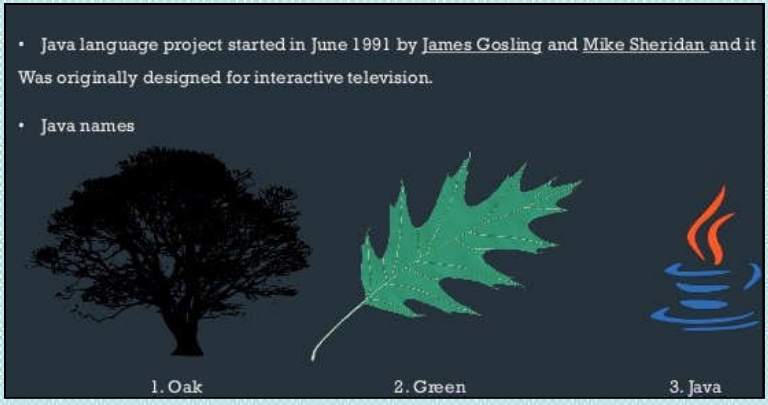
"EVERYBODY IN THIS COUNTRY SHOULD LEARN HOW TO PROGRAM A COMPUTER, BECAUSE IT TEACHES YOU HOW TO THINK."

STEVE JOBS



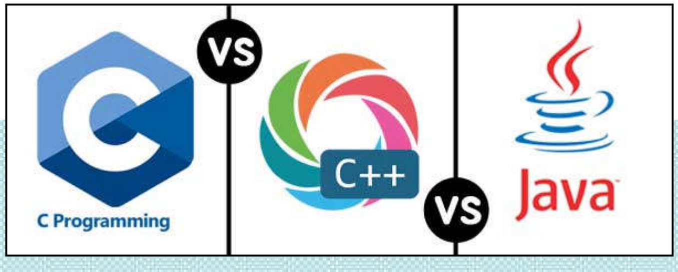
Write Once Run Any where

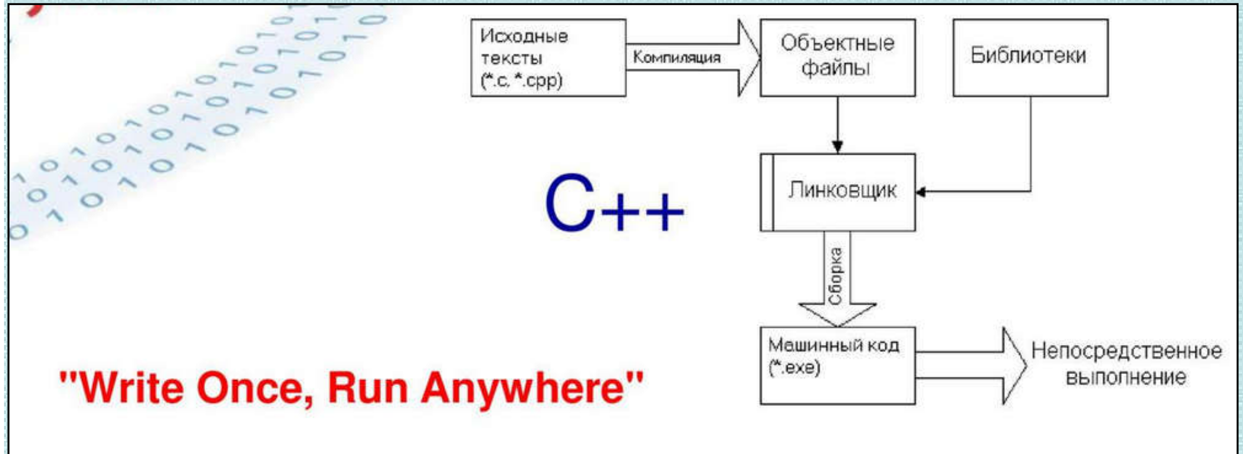




- Java language project started in June 1991 by James Gosling and Mike Sheridan and it Was originally designed for interactive television.
- Java names

ВЗАИМОСВЯЗЬ МЕЖУ JAVA И ЯЗЫКАМИ С И С++




C++

Исходные тексты (*.c, *.cpp)

Компиляция

Объектные файлы

Библиотеки

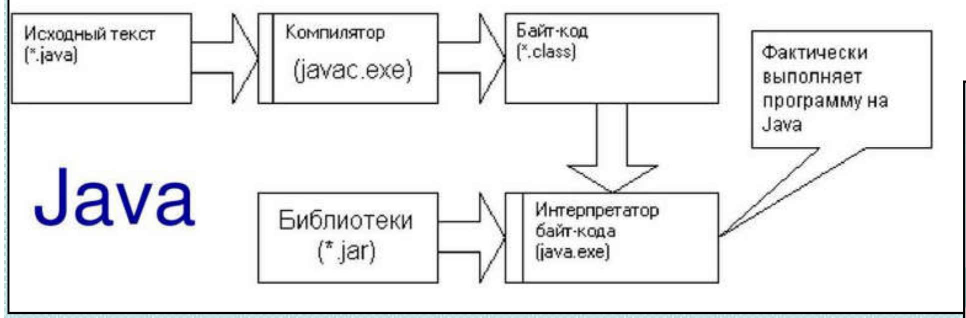
Линковщик

Сборка

Машинный код (*.exe)

Непосредственное выполнение

"Write Once, Run Anywhere"


Java

Исходный текст (*.java)

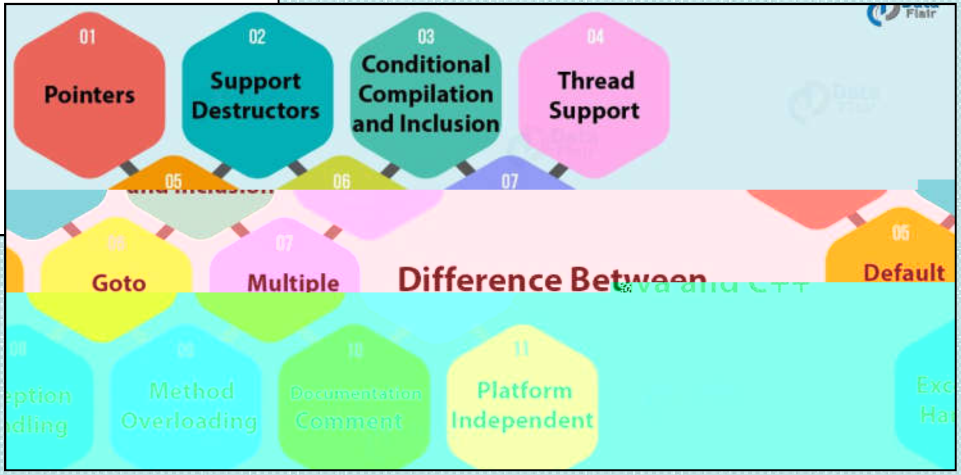
Компилятор (javac.exe)

Байт-код (*.class)

Фактически выполняет программу на Java

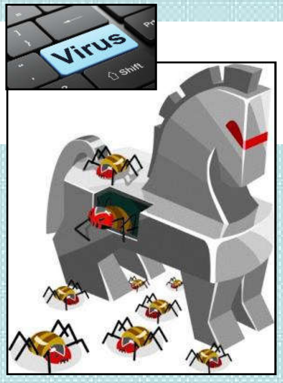
Библиотеки (*.jar)

Интерпретатор байт-кода (java.exe)

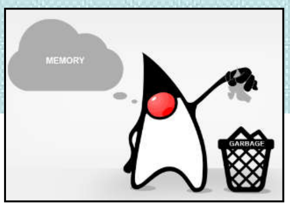


От *C* язык *Java* унаследовал синтаксис,
а от *C*++ – объектную модель

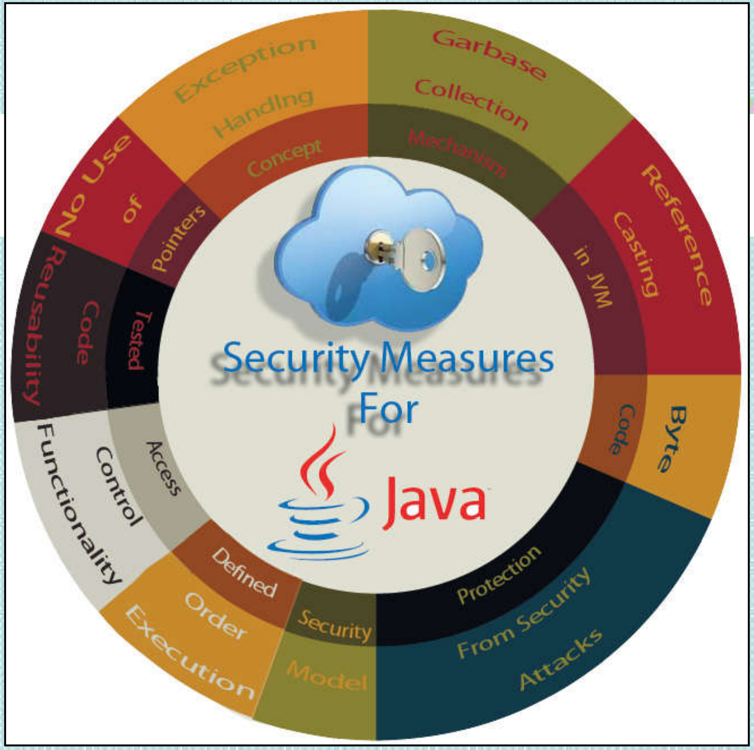
БЕЗОПАСНОСТЬ

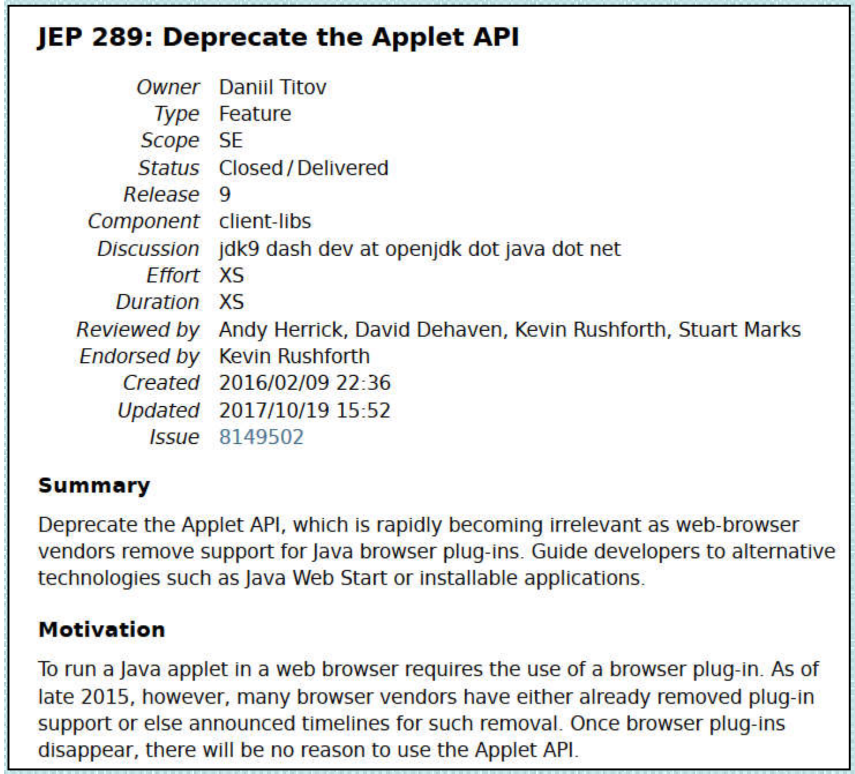


ORACLE Certified Professional



- отсутствие адресной арифметики
- технология «сборки мусора» (garbage collection)
- строгая типизация
- отсутствие множественного наследования классов
- запрет перегрузки операторов
- встроенная обработка исключений


Security Measures For Java
Exception Handling Concept
Garbage Collection Mechanism
Reference Casting in JVM
Byte Code
Protection From Security Attacks
Security Model
Defined Order Execution
Access Control Functionality
Tested Code Reusability
No Use of Pointers



JEP 289: Deprecate the Applet API

Owner	Daniil Titov
Type	Feature
Scope	SE
Status	Closed / Delivered
Release	9
Component	client-libs
Discussion	jdk9 dash dev at openjdk dot java dot net
Effort	XS
Duration	XS
Reviewed by	Andy Herrick, David Dehaven, Kevin Rushforth, Stuart Marks
Endorsed by	Kevin Rushforth
Created	2016/02/09 22:36
Updated	2017/10/19 15:52
Issue	8149502

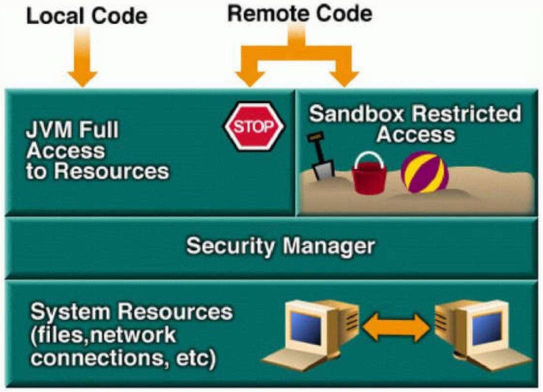
Summary

Deprecate the Applet API, which is rapidly becoming irrelevant as web-browser vendors remove support for Java browser plug-ins. Guide developers to alternative technologies such as Java Web Start or installable applications.

Motivation

To run a Java applet in a web browser requires the use of a browser plug-in. As of late 2015, however, many browser vendors have either already removed plug-in support or else announced timelines for such removal. Once browser plug-ins disappear, there will be no reason to use the Applet API.

JVM - аналог виртуального компьютера, расположенного в оперативной памяти и интерпретирующего байт-код. Все действия Java-программы замкнуты внутри этого виртуального компьютера. JVM может не допускать деструктивных действий Java-программ.



Local Code

Remote Code

JVM Full Access to Resources

STOP

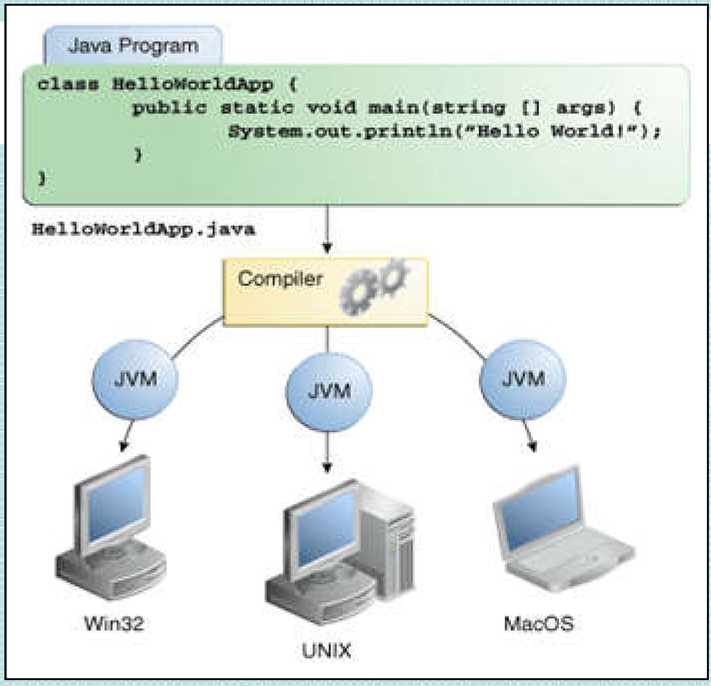
Sandbox Restricted Access

Security Manager

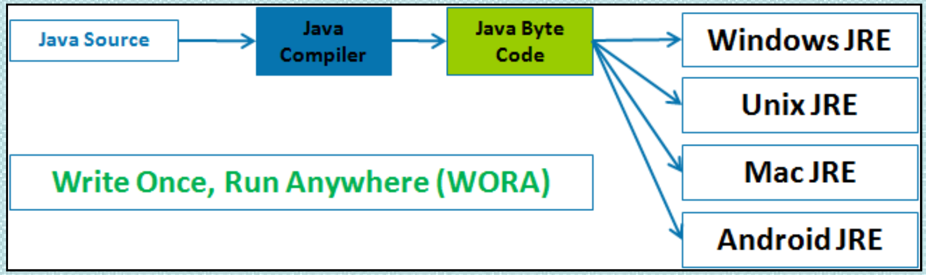
System Resources (files,network connections, etc)

ПЕРЕНОСИМОСТЬ

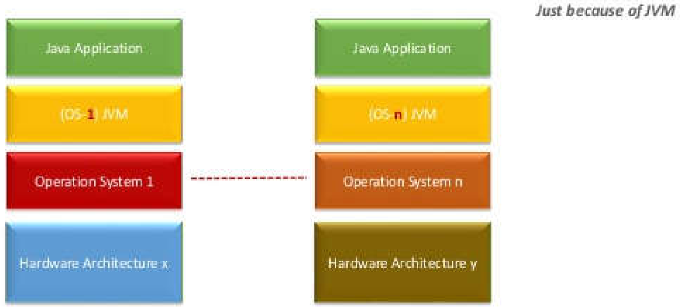
Portability: Platform Independent

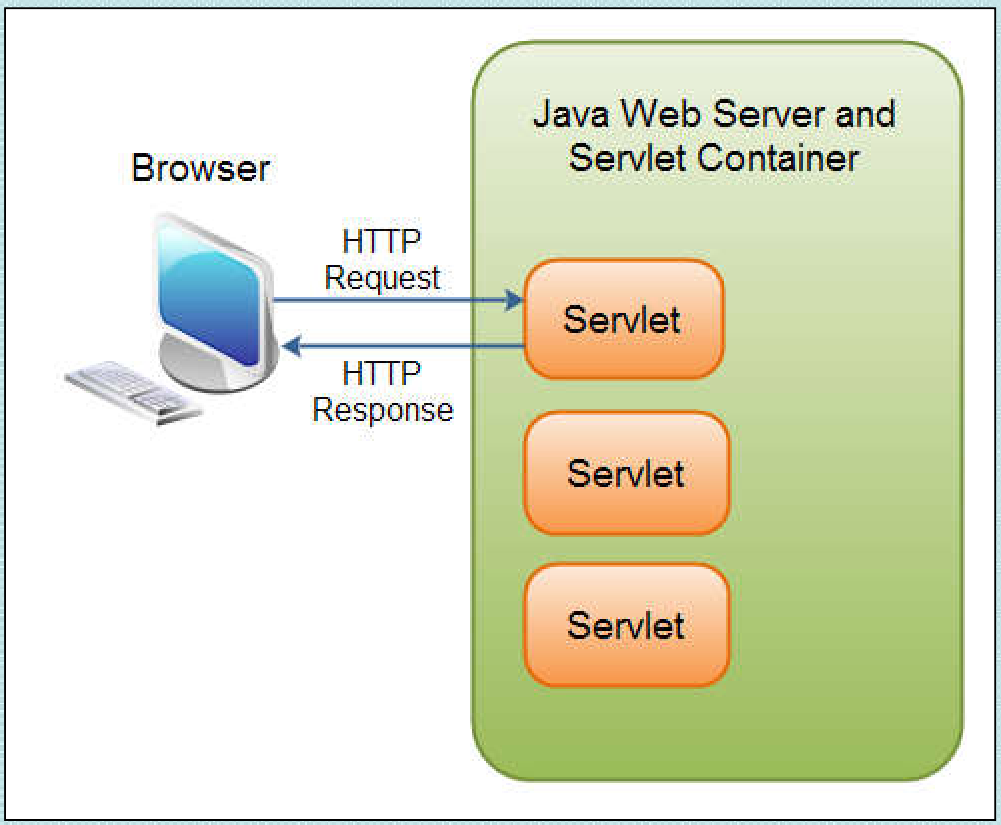


```
class HelloWorldApp {
        public static void main(string [] args) {
                System.out.println("Hello World!");
        }
}
```

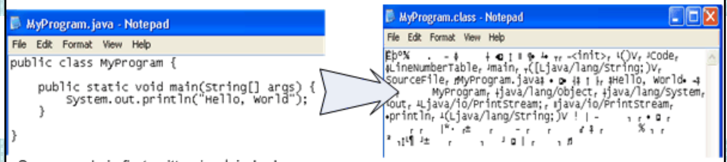


Write Once, Run Anywhere - Promise





ВОЛШЕБНЫЙ БАЙТ-КОД JAVA


MyProgram.java - Notepad
File Edit Format View Help
MyProgram.class - Notepad
File Edit Format View Help

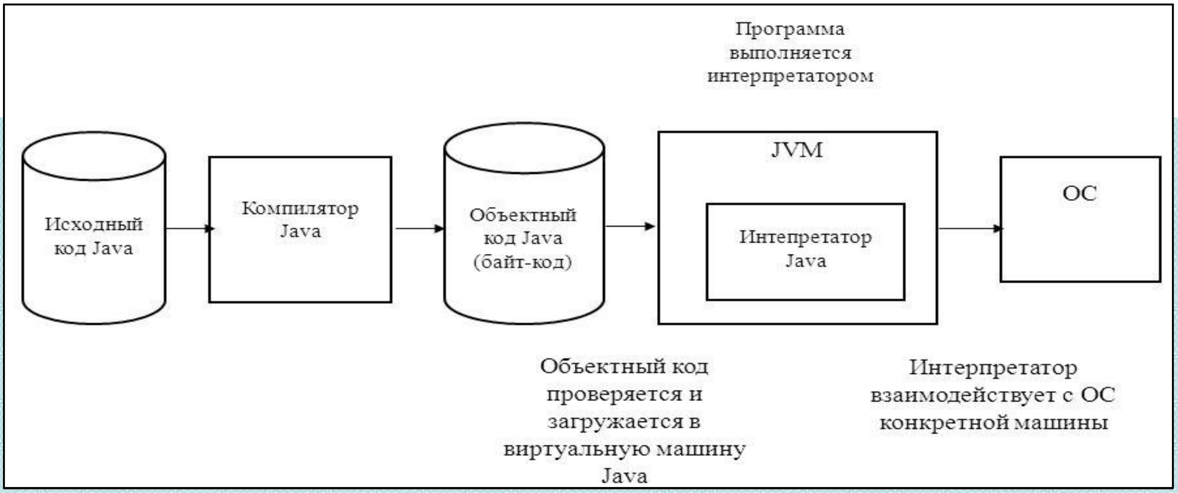
```
public class MyProgram {

    public static void main(String[] args) {
        System.out.println("Hello, World");
    }

}
```

```
Eþ°%   .  - ‡      ‡ ◘ ↕ ‼ ¶ ↓ ┬┬ -<init>, ↓()V, ♯Code,
♯LineNumberTable, ♪main, ┬([Ljava/lang/String;)V,
SourceFile, ♬MyProgram.java‡ • ► ‼ ↕ ↓, ♯Hello, World• ↵
         MyProgram, ‡java/lang/Object, ‡java/lang/System,
out, ↓Ljava/io/PrintStream;, ‡java/io/PrintStream,
•println, ↓(Ljava/lang/String;)V ! | -     ┐ ┌ • ◘ ┌
```

Source code is first written in plain text files ending with the .java extension.

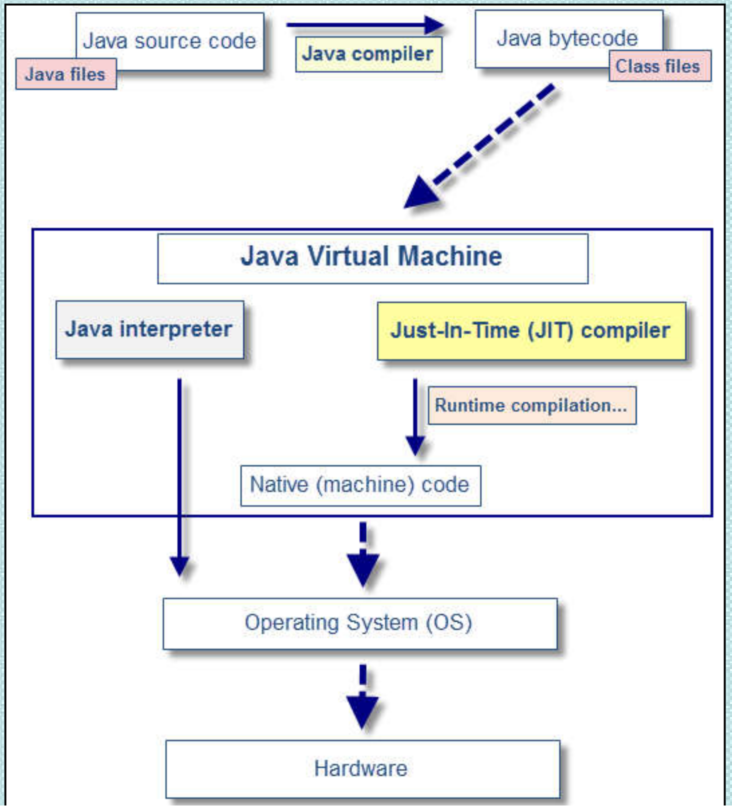
After the compilation is successful, java compiler will generate an intermediate ".class" file that contains the bytecode.

Compiled	C, C++, Objective-C
Interpreted	PHP, JavaScript
Hybrid	Java, C#, VB.NET, Python

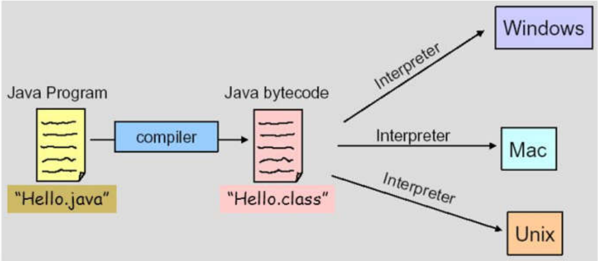

Программа выполняется интерпретатором
Исходный код Java
Компилятор Java
Объектный код Java (байт-код)
JVM
Интепретатор Java
ОС

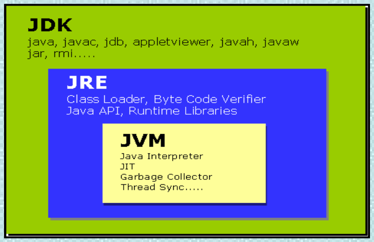
Объектный код проверяется и загружается в виртуальную машину Java

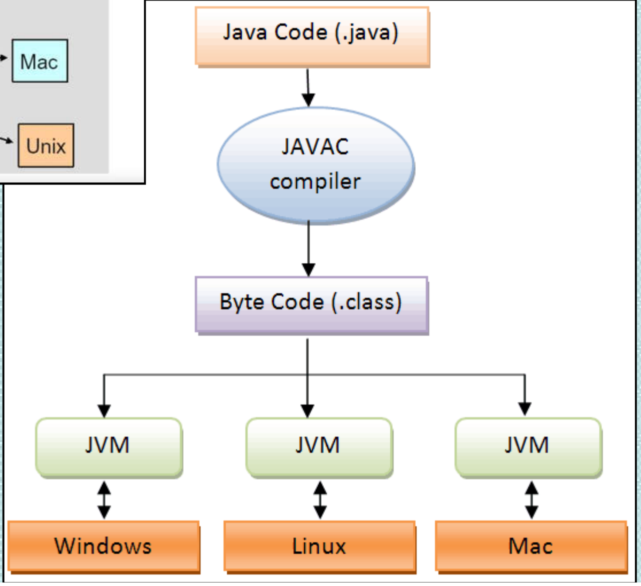
Интерпретатор взаимодействует с ОС конкретной машины



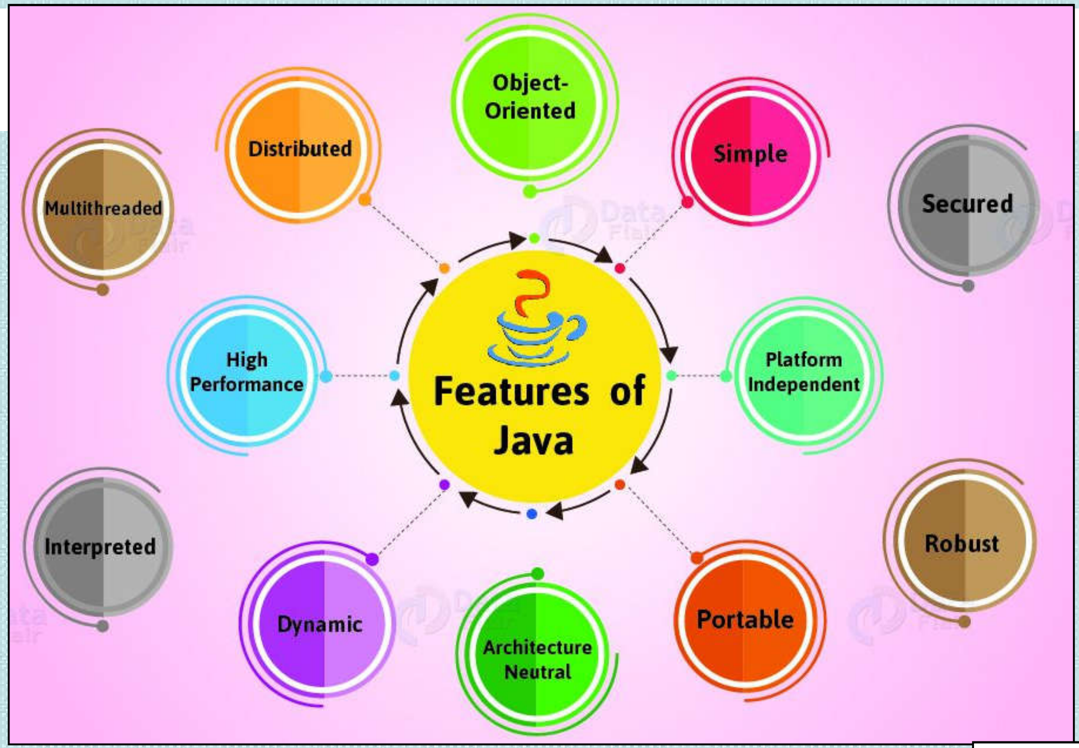
You can run bytecode on any computer that has a Java Interpreter installed




JDK
java, javac, jdb, appletviewer, javah, javaw
jar, rmi.....
JRE
Class Loader, Byte Code Verifier
Java API, Runtime Libraries
JVM
Java Interpreter
JIT
Garbage Collector
Thread Sync.....

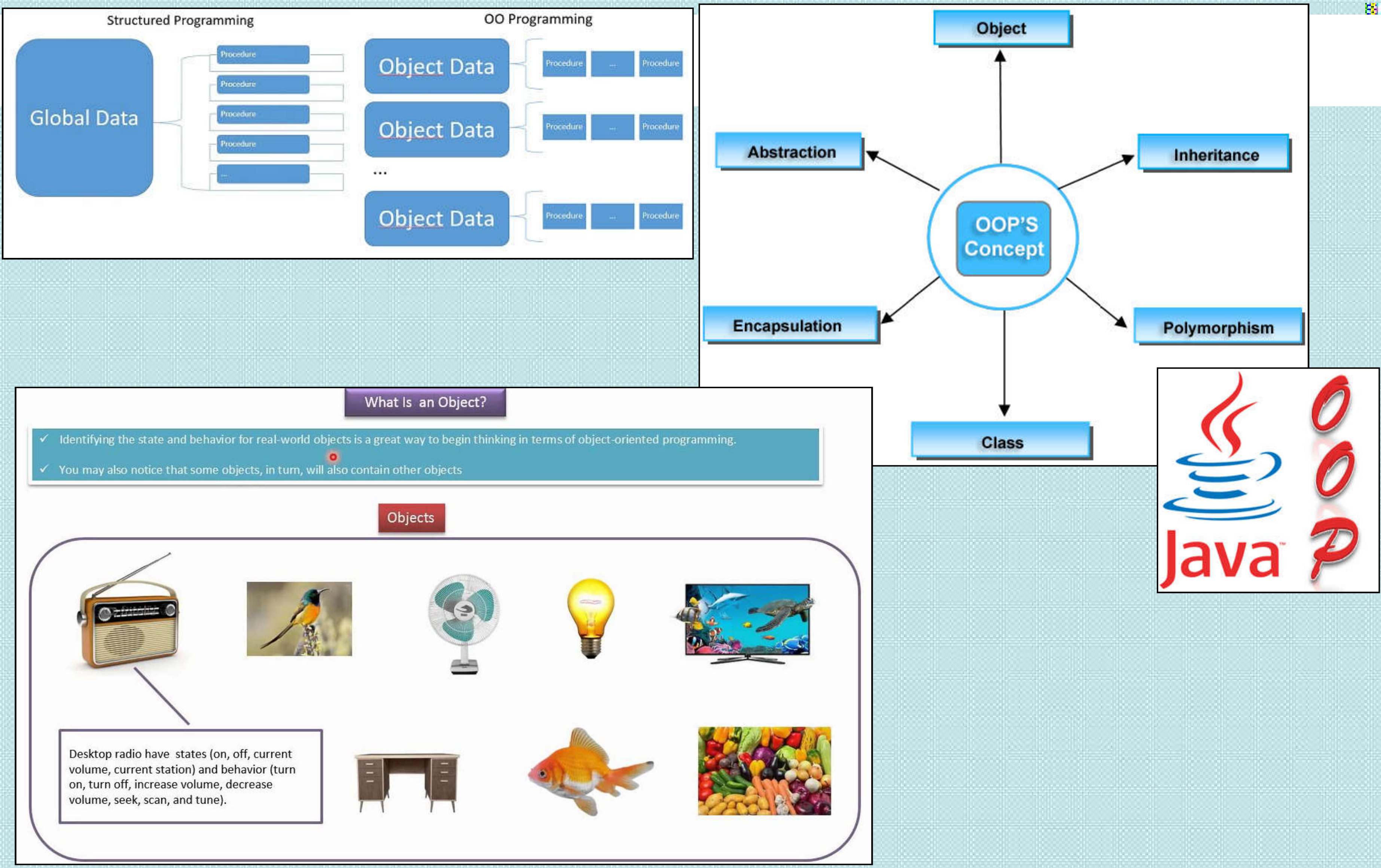


ОСНОВНЫЕ ХАРАКТЕРИСТИКИ JAVA



- архитектурная независимость и переносимость кода
- полная объектная ориентированность
- устойчивость (надежность) кода
- встроенный механизм поддержки многопоточности
- безопасность Java-программ
- встроенная структура коллекций
- удобство разработки GUI

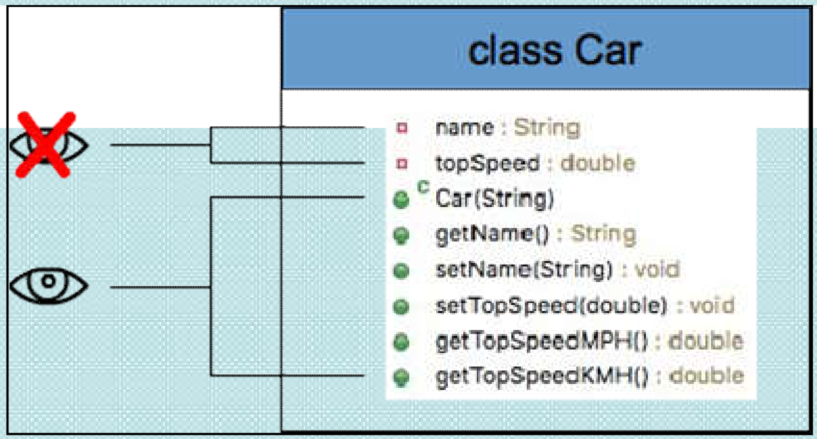
ОСНОВНЫЕ ХАРАКТЕРИСТИКИ JAVA

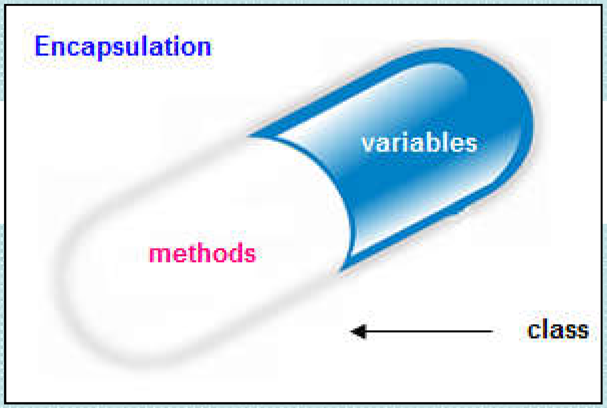


- ✓ Identifying the state and behavior for real-world objects is a great way to begin thinking in terms of object-oriented programming.
- ✓ You may also notice that some objects, in turn, will also contain other objects

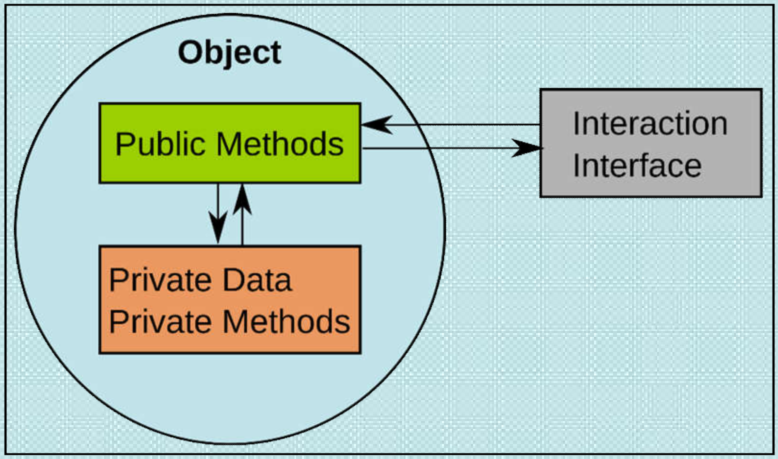
Desktop radio have states (on, off, current volume, current station) and behavior (turn on, turn off, increase volume, decrease volume, seek, scan, and tune).

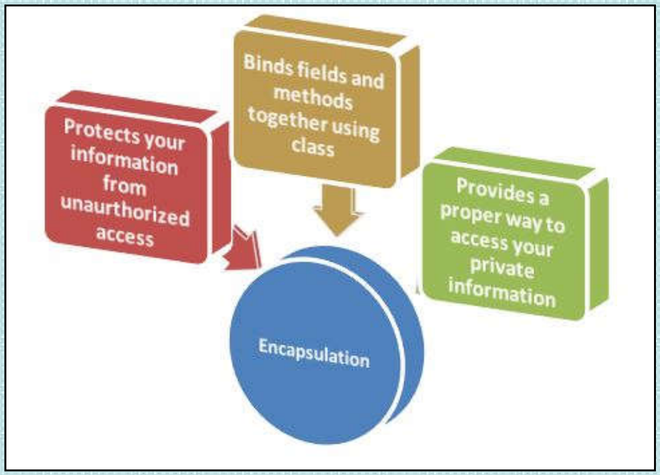
ИНКАПСУЛЯЦИЯ

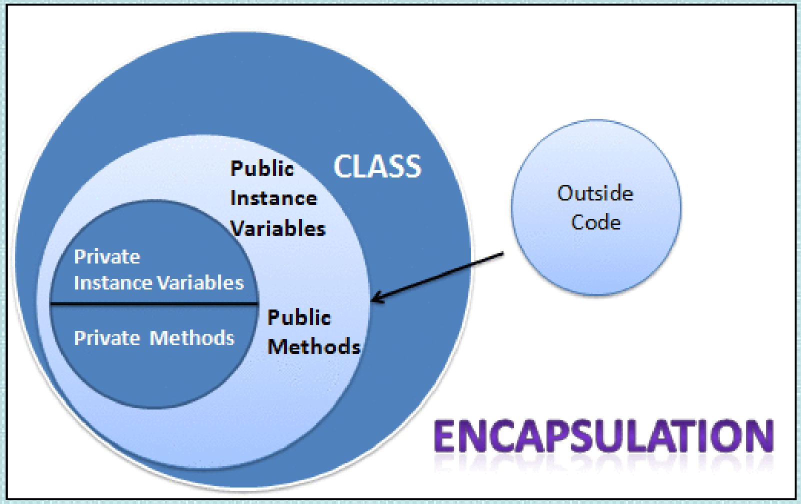




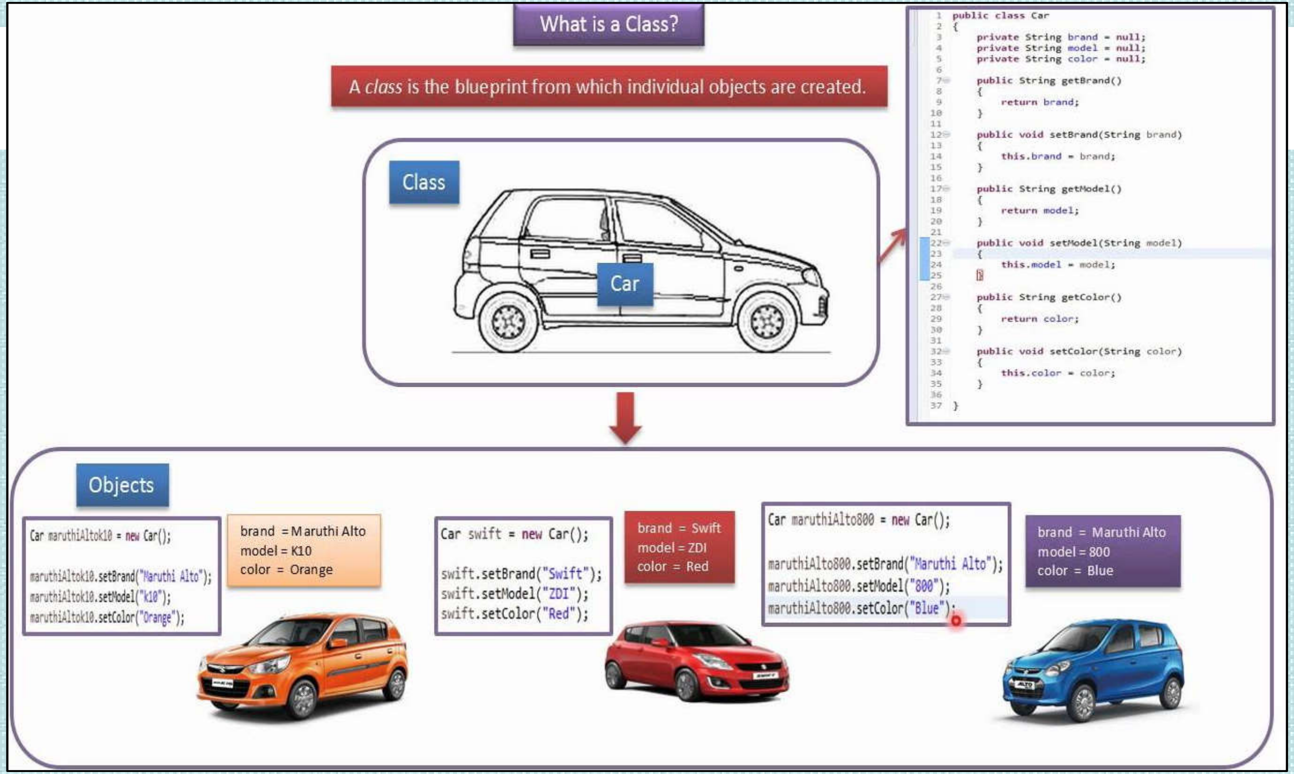
Java OOP




Protects your information from unaurthorized access
Binds fields and methods together using class
Provides a proper way to access your private information
Encapsulation



КЛАСС



What is a Class?

A *class* is the blueprint from which individual objects are created.

```
public class Car
{
    private String brand = null;
    private String model = null;
    private String color = null;

    public String getBrand()
    {
        return brand;
    }

    public void setBrand(String brand)
    {
        this.brand = brand;
    }

    public String getModel()
    {
        return model;
    }

    public void setModel(String model)
    {
        this.model = model;
    }

    public String getColor()
    {
        return color;
    }

    public void setColor(String color)
    {
        this.color = color;
    }

}
```

```
Car maruthiAltok10 = new Car();

maruthiAltok10.setBrand("Maruthi Alto");
maruthiAltok10.setModel("k10");
maruthiAltok10.setColor("Orange");
```

brand = Maruthi Alto
model = K10
color = Orange

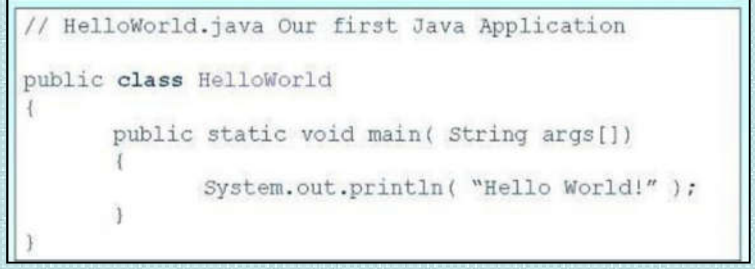
```
Car swift = new Car();

swift.setBrand("Swift");
swift.setModel("ZDI");
swift.setColor("Red");
```

brand = Swift
model = ZDI
color = Red

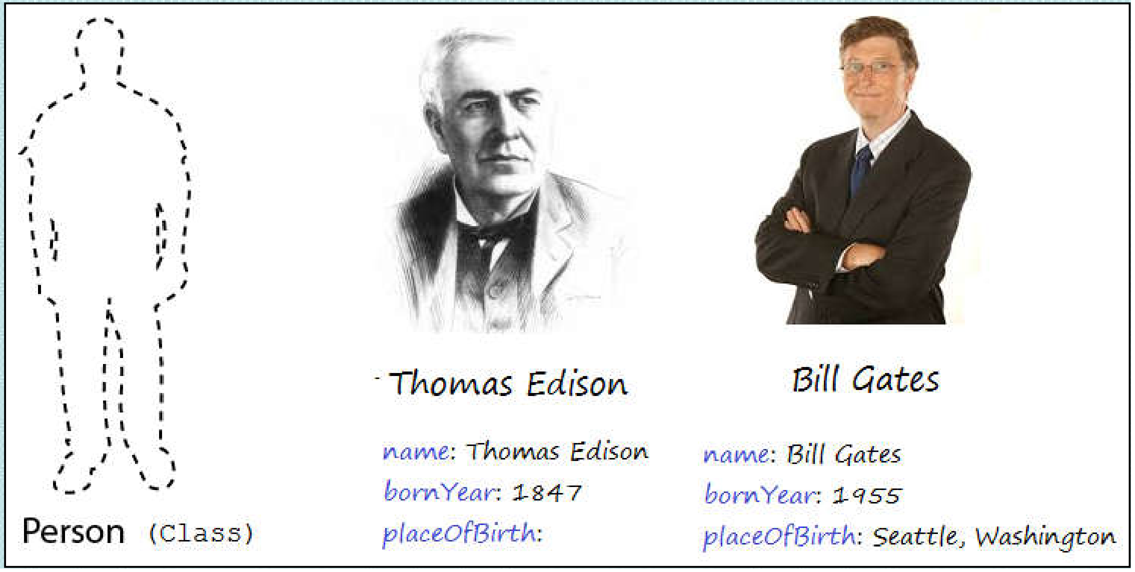
```
Car maruthiAlto800 = new Car();

maruthiAlto800.setBrand("Maruthi Alto");
maruthiAlto800.setModel("800");
maruthiAlto800.setColor("Blue");
```

brand = Maruthi Alto
model = 800
color = Blue

Основная структурная единица программы – класс, весь код Java-программы должен находиться внутри одного или нескольких классов.



```
// HelloWorld.java Our first Java Application

public class HelloWorld
{
          public static void main( String args[])
          {
                    System.out.println( "Hello World!" );
          }
}
```



Person (Class)

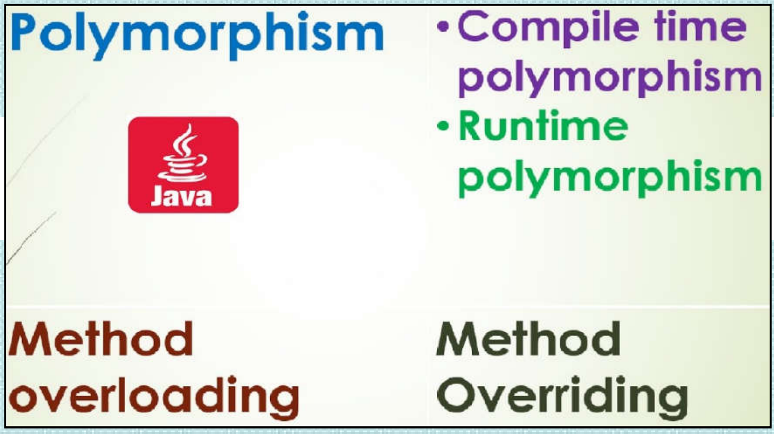
Thomas Edison

name: Thomas Edison
bornYear: 1847
placeOfBirth:

Bill Gates

name: Bill Gates
bornYear: 1955
placeOfBirth: Seattle, Washington

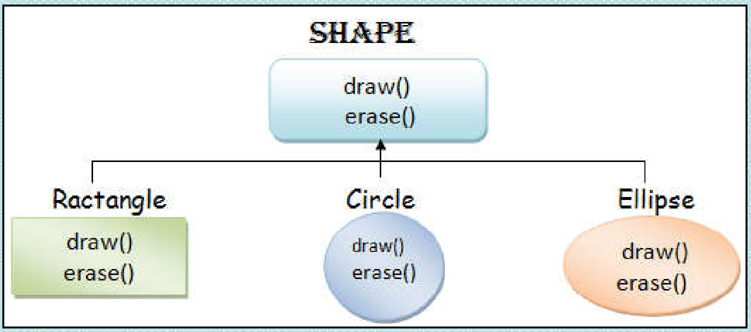
ПОЛИМОРФИЗМ

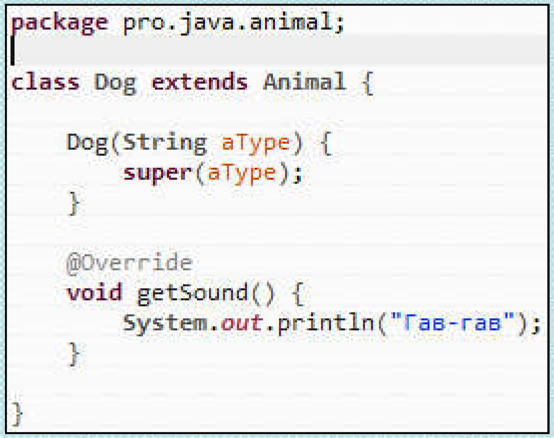

Polymorphism

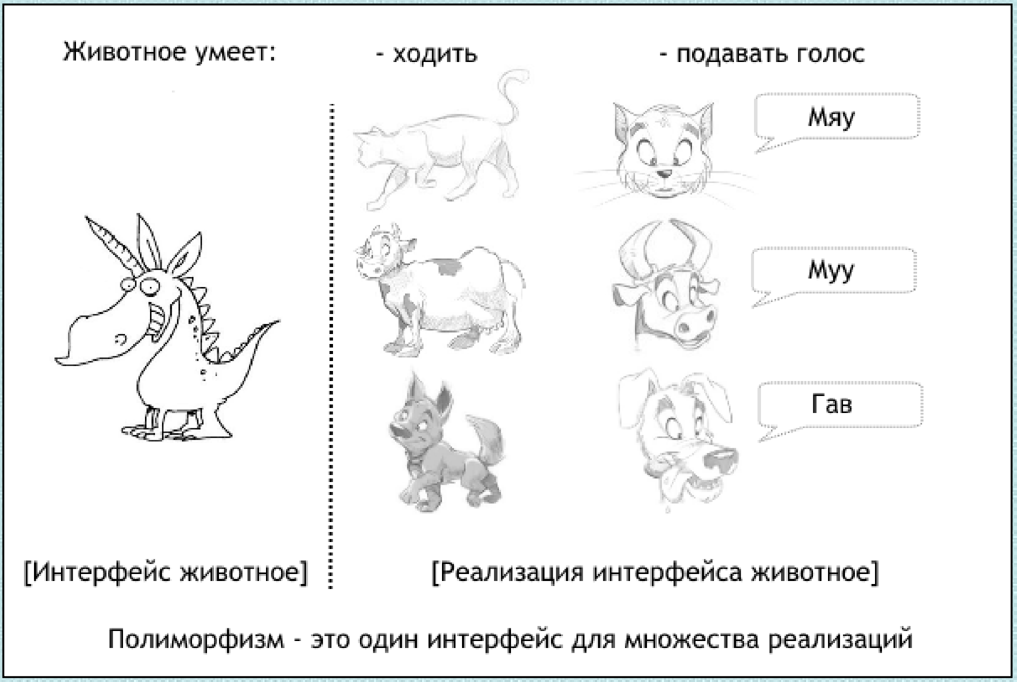
- Compile time polymorphism
- Runtime polymorphism

Method overloading

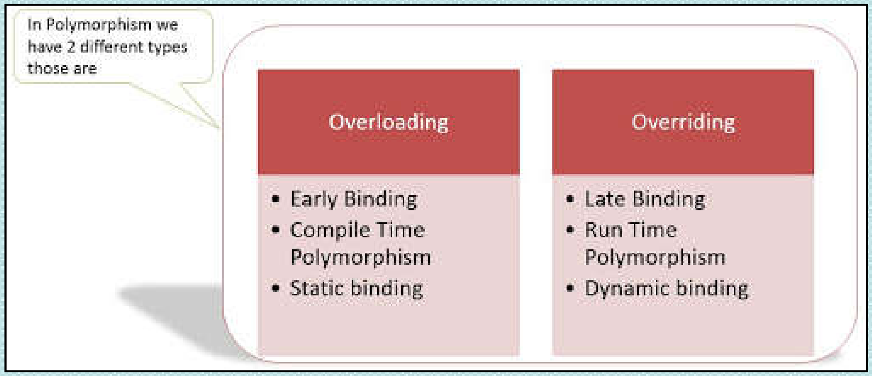
Method Overriding




```
package pro.java.animal;

class Dog extends Animal {

    Dog(String aType) {
        super(aType);
    }

    @Override
    void getSound() {
        System.out.println("Гав-гав");
    }

}
```



Полиморфизм - это один интерфейс для множества реализаций


In Polymorphism we have 2 different types those are

Overloading	Overriding
• Early Binding • Compile Time Polymorphism • Static binding	• Late Binding • Run Time Polymorphism • Dynamic binding

НАСЛЕДОВАНИЕ

Inheritance defines **is-a** relationship between a Super class and its Sub class. extends and implements keywords are used to describe inheritance in Java.

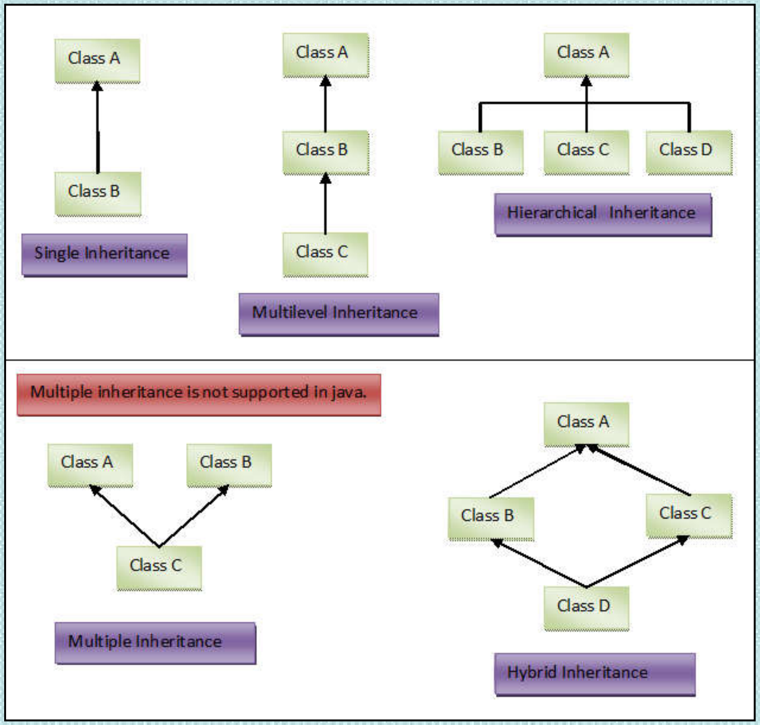
Java Inheritance(IS-A)

```
class Vehicle.
{
 ......
}
class Car extends Vehicle
{
 .......   //extends the property of vehicle class.
}
```

Vehicle is super class of **Car**.
Car is sub class of **Vehicle**.
Car IS-A **Vehicle**.

Purpose of Inheritance

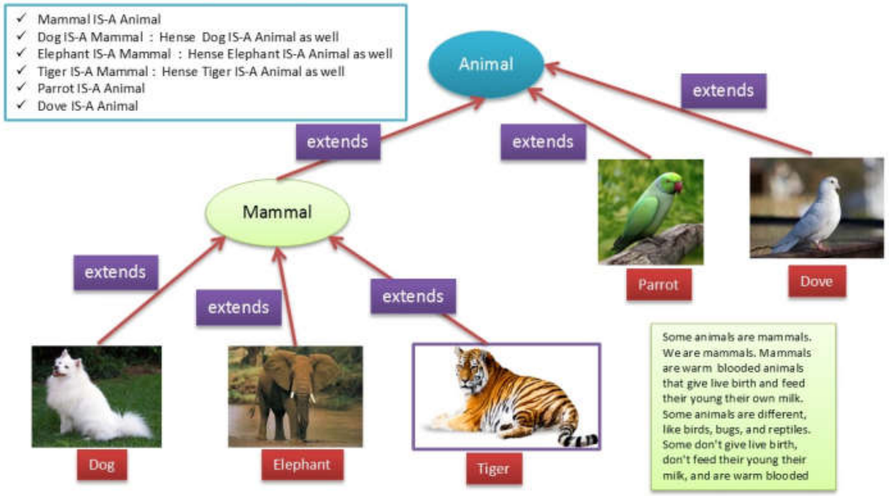
- ✓ To promote code reuse.
- ✓ To use Polymorphism.



Java Inheritance(IS-A)

IS-A is a way of saying : This object is a type of that object.

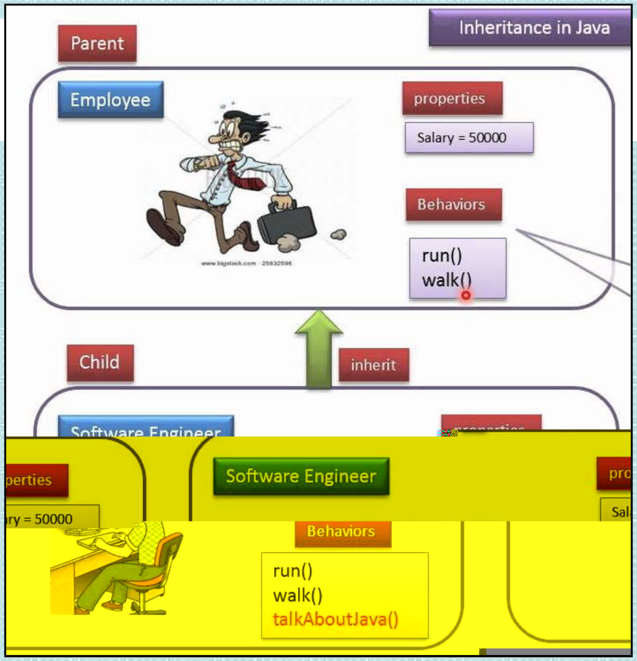
- ✓ Mammal IS-A Animal
- ✓ Dog IS-A Mammal : Hense Dog IS-A Animal as well
- ✓ Elephant IS-A Mammal : Hense Elephant IS-A Animal as well
- ✓ Tiger IS-A Mammal : Hense Tiger IS-A Animal as well
- ✓ Parrot IS-A Animal
- ✓ Dove IS-A Animal

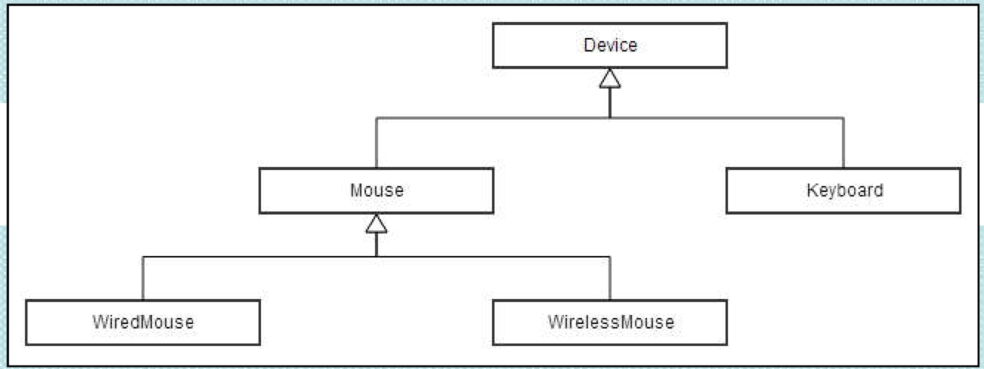


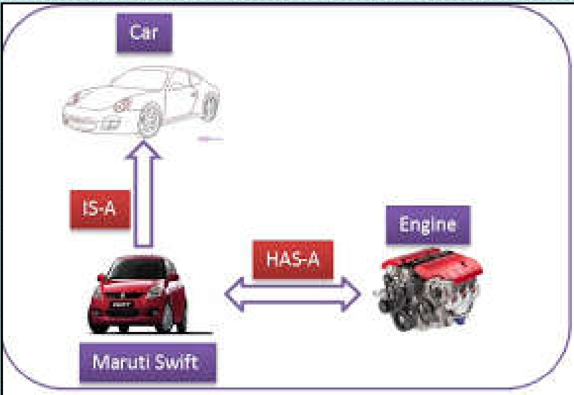
Some animals are mammals. We are mammals. Mammals are warm blooded animals that give live birth and feed their young their own milk. Some animals are different, like birds, bugs, and reptiles. Some don't give live birth, don't feed their young their milk, and are warm blooded

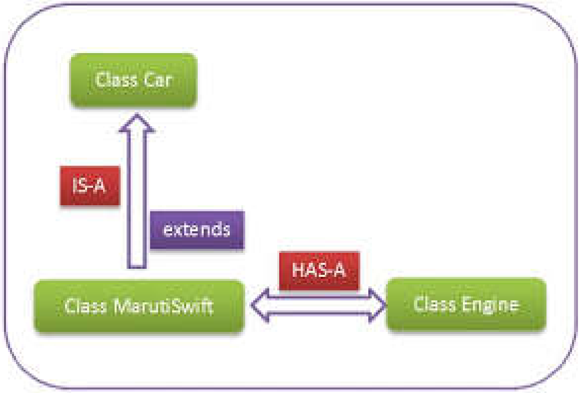
```
public class Animal
{

}
public class Mammal extends Animal
{

}
public class Dog extends Mammal
{

}
public class Elephant extends Mammal
{

}
public class Tiger extends Mammal
{

}
public class Parrot extends Animal
{

}
public class Dove extends Animal
{

}
```

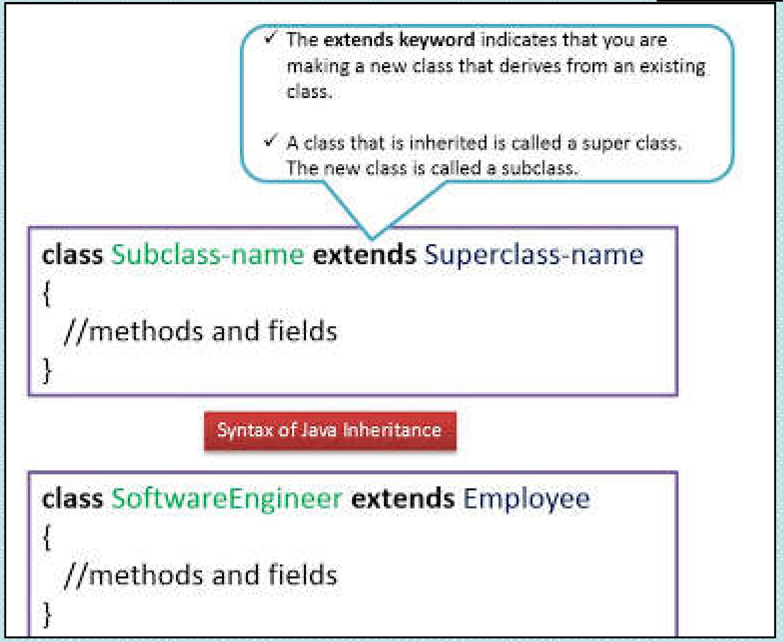
НАСЛЕДОВАНИЕ











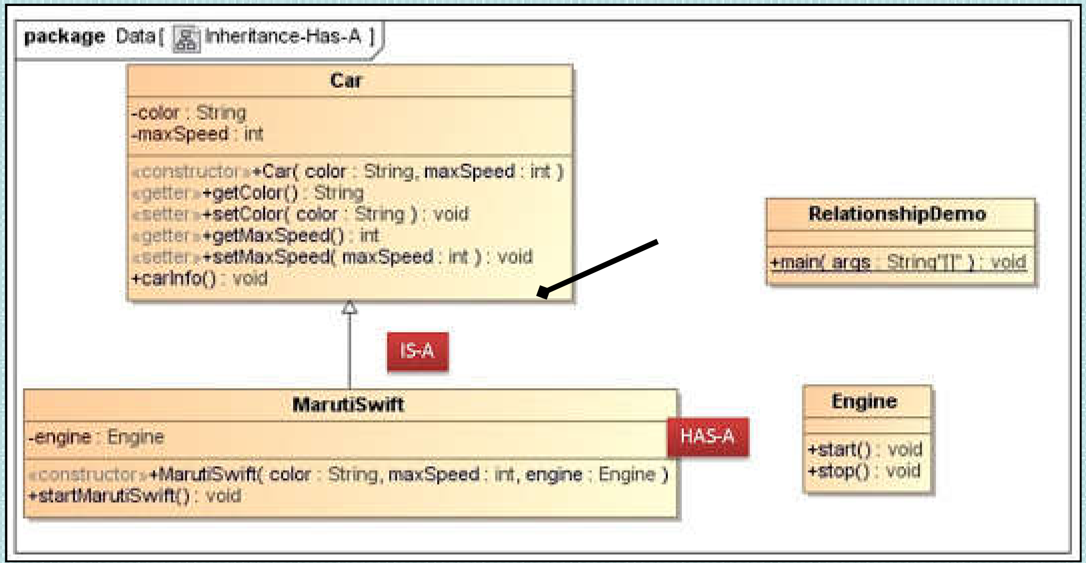
✓ The **extends keyword** indicates that you are making a new class that derives from an existing class.

✓ A class that is inherited is called a super class. The new class is called a subclass.

```
class Subclass-name extends Superclass-name
{
  //methods and fields
}
```

Syntax of Java Inheritance

```
class SoftwareEngineer extends Employee
{
  //methods and fields
}
```



package Data[Inheritance-Has-A]

Car

-color : String
-maxSpeed : int

«constructor»+Car(color : String, maxSpeed : int)
«getter»+getColor() : String
«setter»+setColor(color : String) : void
«getter»+getMaxSpeed() : int
«setter»+setMaxSpeed(maxSpeed : int) : void
+carInfo() : void

RelationshipDemo

+main(args : String"[]") : void

IS-A

MarutiSwift

-engine : Engine

«constructor»+MarutiSwift(color : String, maxSpeed : int, engine : Engine)
+startMarutiSwift() : void

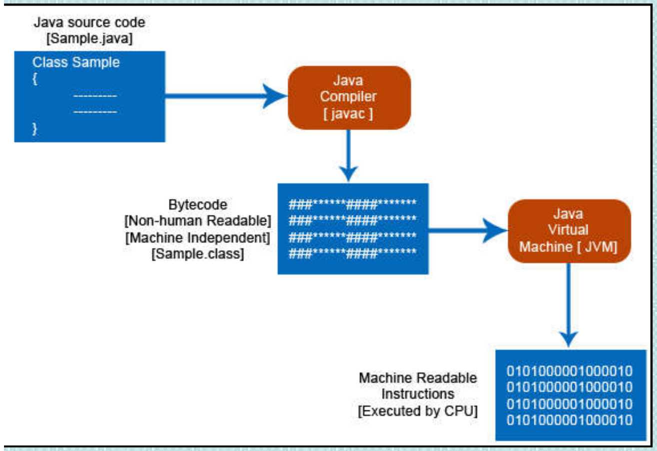
HAS-A

Engine

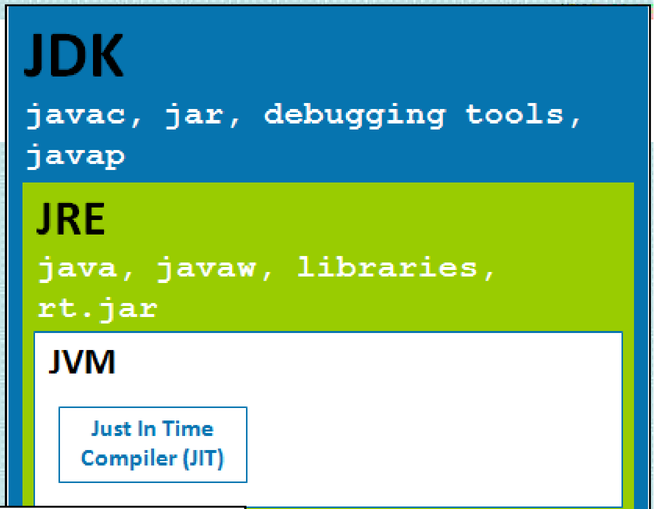
+start() : void
+stop() : void

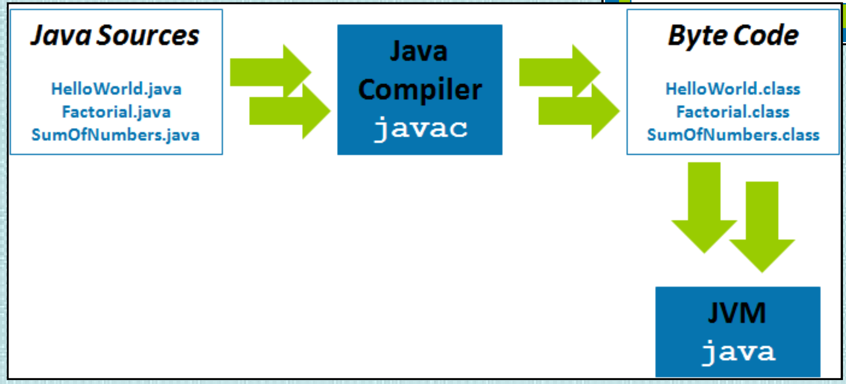
УСТАНОВКА КОМПЛЕКТА JAVA DEVELOPMENT KIT

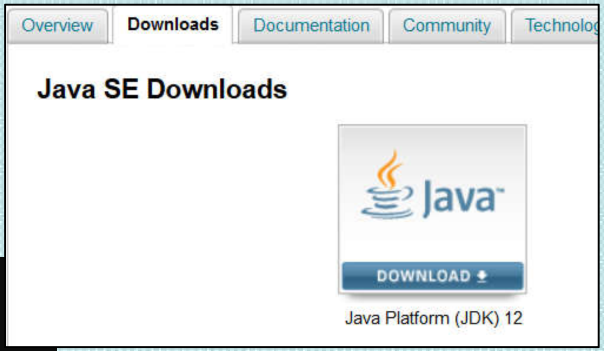
Java Development Kit
JDK



ORACLE TO BUY SUN

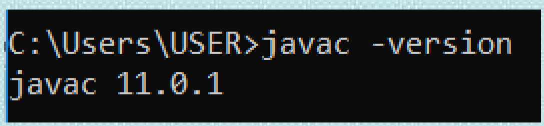




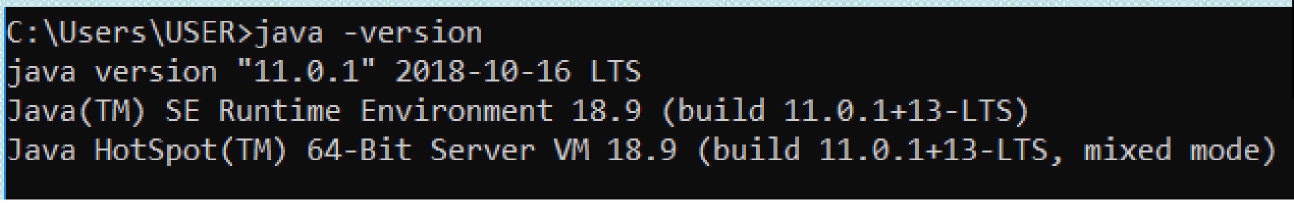


Java SE Downloads

Java Platform (JDK) 12

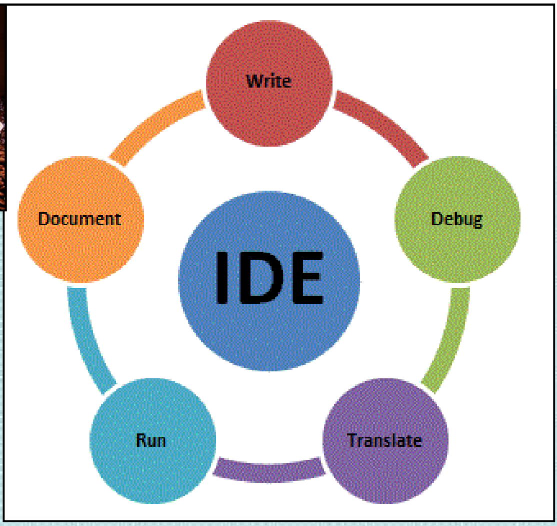


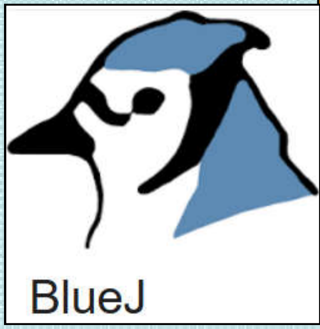
```
C:\Users\USER>javac -version
javac 11.0.1
```



```
C:\Users\USER>java -version
java version "11.0.1" 2018-10-16 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.1+13-LTS)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.1+13-LTS, mixed mode)
```

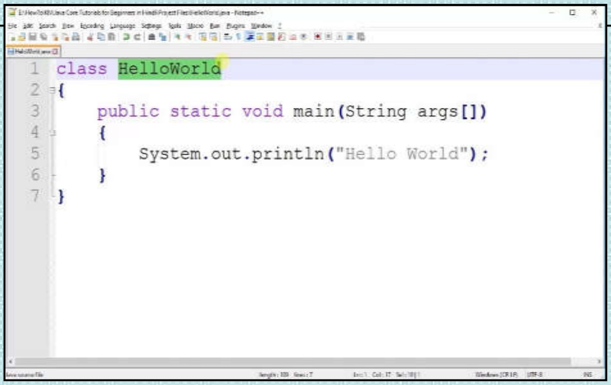
ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ





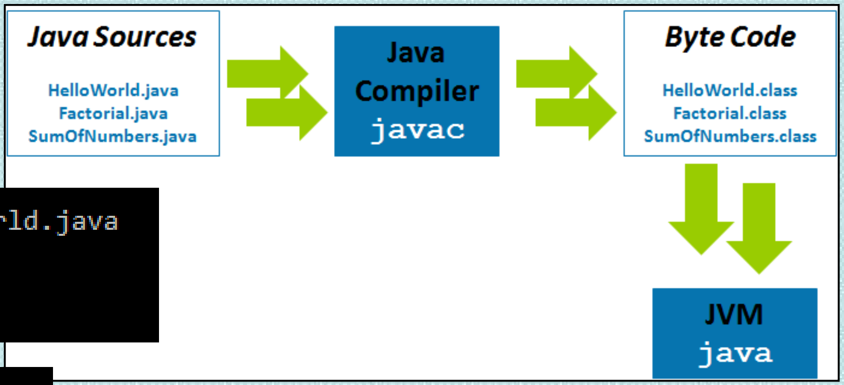
ПЕРВАЯ ПРОГРАММА НА JAVA

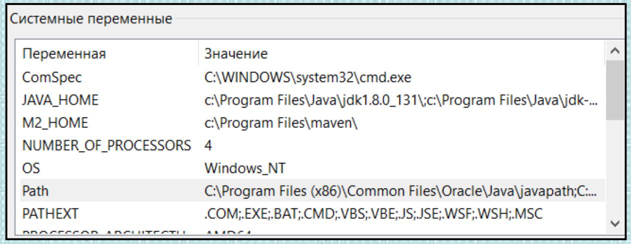
```
/*
Это пример простой программ на Java. Присвойте файлу с исходным кодом
имя HelloWorld.java
*/
class HelloWorld{
//выполнение любой программы на Java начинается с вызова метода main()
public static void main(String args[]) {
        System.out.println("Java правит Интернетом!");
        }
}
```



```
class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Hello World");
    }
}
```

1) ввести исходный код программы;
2) скомпилировать программу;
3) запустить программу на выполнение.

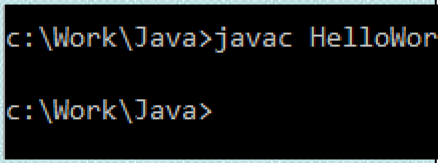




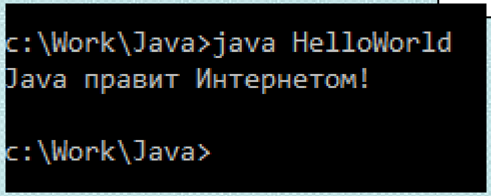
Системные переменные

Переменная	Значение
ComSpec	C:\WINDOWS\system32\cmd.exe
JAVA_HOME	c:\Program Files\Java\jdk1.8.0_131\;c:\Program Files\Java\jdk-...
M2_HOME	c:\Program Files\maven\
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

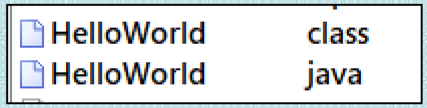
c:\Program Files\Java\jdk-10.0.1\bin



```
c:\Work\Java>javac HelloWorld.java

c:\Work\Java>
```



```
c:\Work\Java>java HelloWorld
Java правит Интернетом!

c:\Work\Java>
```



HelloWorld	class
HelloWorld	java

Спасибо за внимание!