

Tổng quan về công nghệ Java

Nội dung

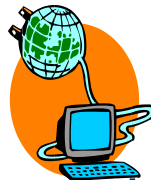
- Lịch sử phát triển
- Công nghệ Java
- Các dạng chương trình Java
- Đặc điểm của Java
- Máy ảo Java (Java Virtual Machine)
- Viết, dịch, thực thi chương trình HelloWorld
- Môi trường, công cụ: giới thiệu một số IDE phổ biến

Lịch sử phát triển

- 1991: Sun Microsystems phát triển OAK nhằm mục đích viết phần mềm điều khiển (**phần mềm nhúng**) cho các sản phẩm gia dụng.



- 1995: internet bùng nổ, phát triển mạnh. Sun phát triển OAK và giới thiệu ngôn ngữ lập trình mới tên Java



- Java là ngôn ngữ hướng đối tượng tựa C, C++



Lịch sử phát triển Java Development Kit (JDK)

- ▶ Môi trường phát triển và thực thi do Sun Microsystems cung cấp (<http://java.sun.com>)
- ▶ Bao gồm phần mềm và công cụ giúp compile, debug and execute ứng dụng.
 - ▶ JDK 1.0 - 1996
 - ▶ JDK 1.1 - 1997
 - ▶ JDK 1.2 (Java 2) - 1998
 - ▶ JDK 1.3 - 2000
 - ▶ Java 1.4 - 2002
 - ▶ Java 5 (1.5) - 2004
 - ▶ Java 6 - 2006
 - ▶ Java SE 7 - 2011
 - ▶ Java SE 8 - 2014
 - ▶ Java 9 - 2017
 - ▶ Java 10 - 2018
 - ▶ Java 11 - 2018
 - ▶ Java 12 - 3/2019
 - ▶ JDK 13 - 9/2019
 - ▶ JDK 14 - 3/2020
 - ▶ JDK 15 - 9/2020

Java Development Kit (JDK)

► Bao gồm

- *javac* Chương trình dịch chuyển mã nguồn sang bytecode
- *java* Bộ thông dịch: Thực thi java application
- *appletviewer* Bộ thông dịch: Thực thi java applet mà không cần sử dụng trình duyệt như Netscape, hay IE, v.v.
- *javadoc* Bộ tạo tài liệu dạng HTML từ mã nguồn và chú thích

Java Development Kit (JDK)

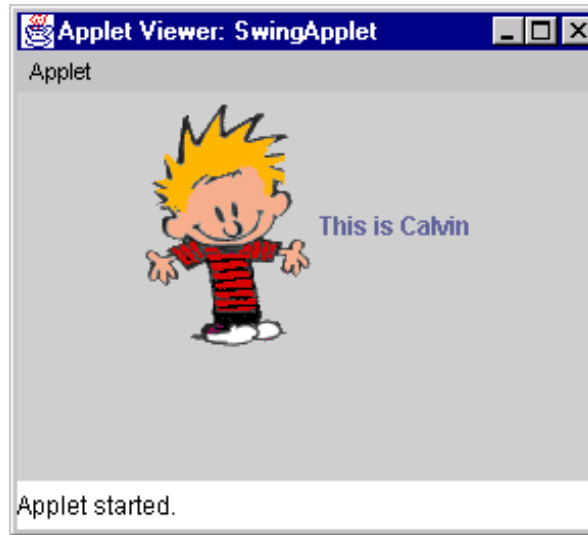
- ▶ Bao gồm
 - ▶ *jdb* Bộ gỡ lỗi (java debugger)
 - ▶ *javap* Trình dịch ngược bytecode

Công nghệ Java

- ▶ Công nghệ:
 - ▶ Ngôn ngữ lập trình
 - ▶ Môi trường thực thi và triển khai
 - ▶ Môi trường phát triển
- ▶ Công nghệ J2SE (Java 2 Standard Edition)
- ▶ Công nghệ J2EE (Java 2 Enterprise Edition)
- ▶ Công nghệ J2ME (Java 2 Micro Edition)

Các dạng chương trình Java

► Applets



Các dạng chương trình Java (tt)

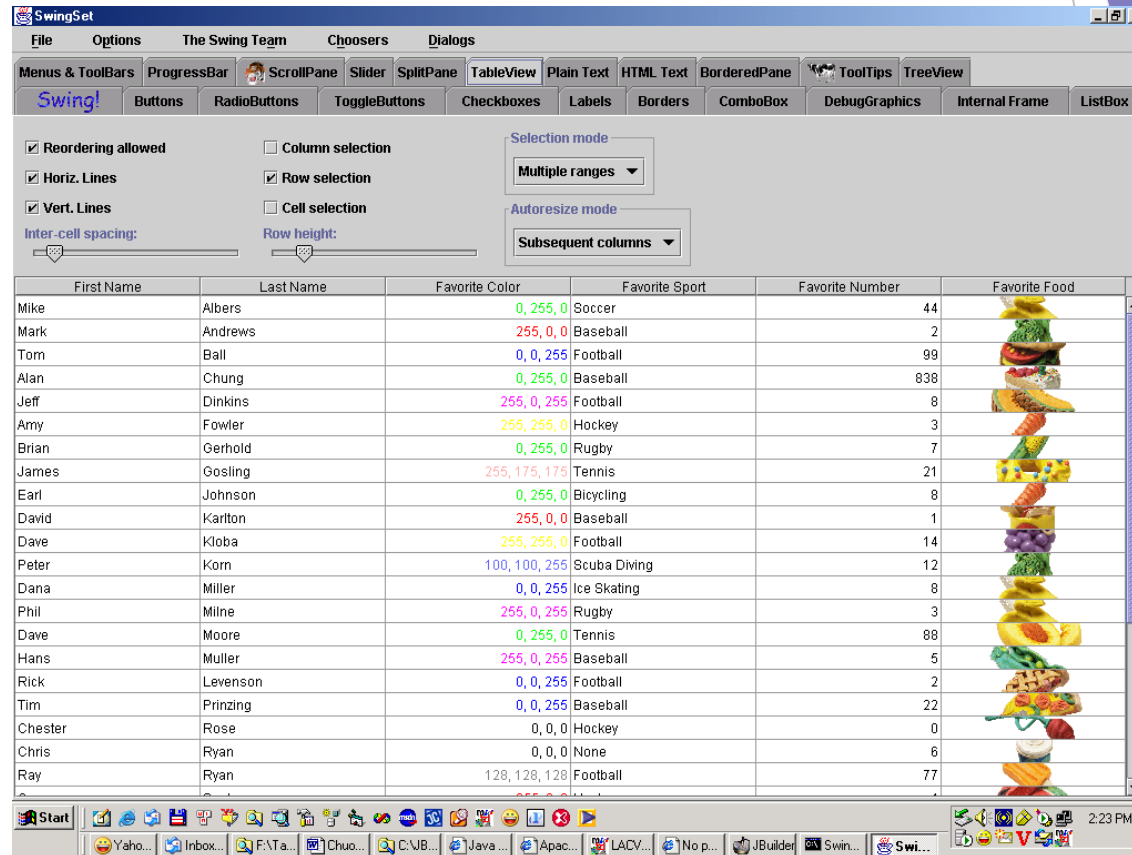
► Console Applications



A screenshot of a Windows command prompt window titled "C:\NT\System32\cmd.exe". The window shows the command `C:\>java Arraytest` being executed. The output of the program is displayed as a list of numbers: `1`, `2`, `3`, `4`, and `5`. The prompt `C:\>_` is visible at the bottom of the window, indicating that the command has completed execution.

Các dạng chương trình Java (tt)

► Ứng dụng Desktop



Các dạng chương trình Java (tt)

► Ứng dụng Web

The screenshot displays the nhadat.com.vn website. At the top, there's a green header with the phone number 9302982 and email info@nhadat.com. Below this is a navigation bar with links like 'trang đầu', 'tìm mua/thuê', 'bán/cho thuê', and 'săn lùng mặt tiền'. The main content area features a large banner for 'Đón lộc KIM CƯƠNG Thịnh Vượng KIMHONG' with an image of a building. To the left, there's a sidebar with a search form and a list of services. To the right, there's another sidebar with a search form and a list of services. The bottom section contains several smaller listings, including 'Nhà đất Giá Tốt Nhất', 'Nhà xây sẵn trong khu Đô Thị Mới', and 'Biệt Thự Cho Thuê Q.Thủ Đức'.

nhadat.com.vn
nhà đất cho mọi người

TP Hồ Chí Minh
tỉnh thành khác

THỰC ĐƠN NHANH

- Trang nhà đất chấm com của tôi
- Trở thành đối tác của nhadat.com

Hãy điền vào địa chỉ Email của bạn để nhận được tin tức địa ốc thường xuyên

VINH TUONG

HoREA®

Khu phố yên vui
Tiện nghi hiện đại

Gọi cho chúng tôi ở 9302982 - info@nhadat.com

trang đầu | tìm mua/thuê | bán/cho thuê | săn lùng mặt tiền

Đón lộc KIM CƯƠNG Thịnh Vượng KIMHONG

chợ nhadatchamcom
Mời quý khách chọn một trong các điều kiện tìm kiếm của mình ở dưới đây:

Tin tức

- Dân hiến 65ha đất làm đường.[chi tiết]
- Ưu đãi doanh nghiệp nước ngoài cải tạo chung cư cũ![chi tiết]
- Thị trường nhà đất sau NĐ17: Vẫn chưa chuyển động![chi tiết]

Thị trường

- Nhà đất Giá Tốt Nhất
- Nhà xây sẵn trong khu Đô Thị Mới
 - Khu Nhà Biệt Thự An Khánh, Quận 2
 - Khu nhà phố Tạ Quang Bửu, Quận 8
 - Các khu quy hoạch khác

KHU DÂN CƯ TÂY NAM

Biệt Thự Cho Thuê Q.Thủ Đức
Với khuôn viên 1800m2, thiết kế sang trọng hiện đại kết hợp với nét kiến trúc cổ...[chi tiết]

Công ty Môi Giới

- Tân Kỳ Nguyên
- Phúc Đức
- Hồng Nhật
- Du Lịch Địa Ốc Ánh Dương
- Mạnh Hùng
- Anh Trung
- nhà đất ĐÔ THI MỚI
- Các nhà môi giới khác

Các dạng chương trình Java (tt)

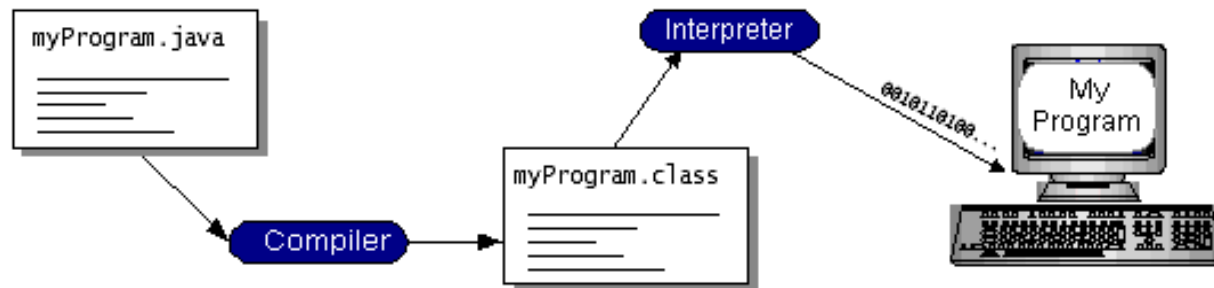
- Một dạng phần mềm nhúng



Đặc điểm Java

- ▶ Tựa C++, hướng đối tượng hoàn toàn.
- ▶ Khả chuyển, độc lập nền.
- ▶ Thông dịch (vừa biên dịch vừa thông dịch).
- ▶ Cơ chế giải phóng bộ nhớ tự động.
- ▶ An toàn, bảo mật.

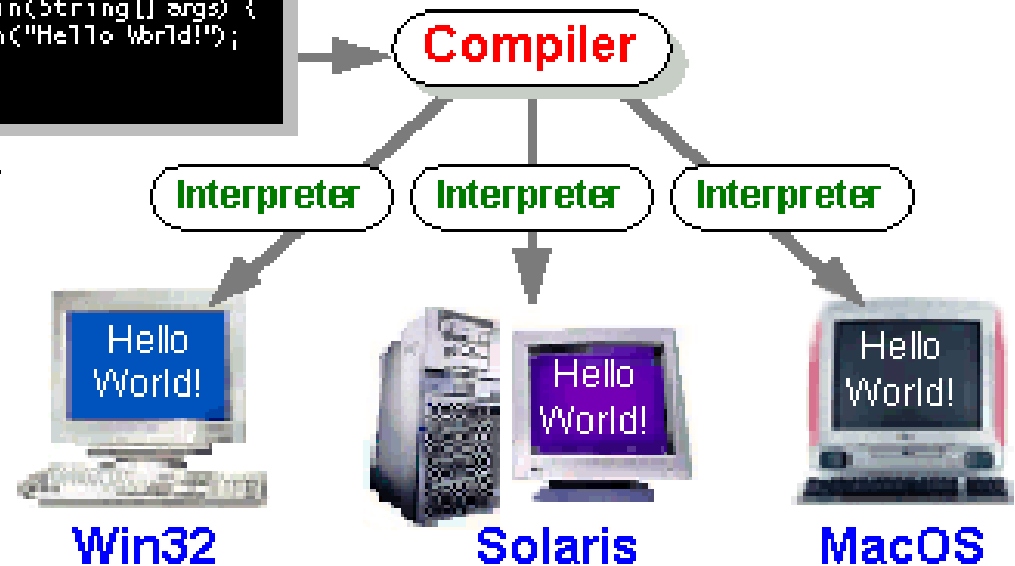
Dịch và thực thi chương trình Java



Java Program

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

HelloWorldApp.java



Java Virtual Machine

- ▶ Là phần mềm dựa trên cơ sở máy tính ảo.
- ▶ Có thể xem như 1 hệ điều hành thu nhỏ.
- ▶ Cung cấp môi trường thực thi cho chương trình java (độc lập nền)
- ▶ Hình thành 1 lớp trừu tượng:
 - Phần cứng máy tính bên dưới
 - Hệ điều hành
 - Mã đã biên dịch
- ▶ Chương trình java chỉ chạy khi có JVM
- ▶ JVM đọc và thực thi từng câu lệnh java

Giải phóng bộ nhớ (Garbage Collection)

- ▶ Java cung cấp một tiến trình mức hệ thống để theo dõi việc cấp phát bộ nhớ
- ▶ Garbage Collection
 - ▶ Đánh dấu và giải phóng các vùng nhớ không còn được sử dụng
 - ▶ Được tiến hành tự động
 - ▶ Cơ chế hoạt động phụ thuộc vào các phiên bản máy ảo

Môi trường, công cụ

- ▶ Môi trường phát triển và thực thi của Sun - JDK 10
- ▶ **IDE (Integrated Development Enviroment)**
 - ▶ Jcreator Pro 5.10
 - ▶ NetBeans 8.2
 - ▶ Eclipse 4.7
 - ▶ JBuilder 2008
 - ▶ IntelliJ IDEA Community Edition, Android Studio, Enide Studio 2014, BlueJ, jEdit, jGRASP, Jsource, Jdeveloper, DrJava,...

Chương trình Hello World

- Dùng Notepad soạn thảo đoạn lệnh bên dưới và lưu lại với tên HelloWorld.java

```
import java.io.*;
```

Khai báo thư viện java.io

```
public class HelloWorld
```

Định nghĩa lớp tên “**HelloWorld**”

```
{
```

Phương thức main

```
    public static void main(String args[])
```

```
    {
```

```
        System.out.print("Hello World");
```

```
    }
```

Bắt đầu đoạn lệnh

```
}
```

Kết thúc đoạn lệnh

Xuất ra Console thông báo

Chương trình Hello World (tt)

- ▶ **Biên dịch:** dùng chương trình **javac**

C:\> javac HelloWorld.java

- ▶ Biên dịch thành công tạo ra tập tin có đuôi **.class** (HelloWorld.class)

- ▶ **Thông dịch (thực thi):** dùng chương trình **java**

C:\> java HelloWorld

Chương trình Hello World (tt)

- ▶ **Lưu ý:** Phải khai báo đường dẫn chỉ đến thư mục cài đặt java, và thư mục chứa các class cần thực thi
- ▶ Ví dụ:

```
C:\> set path=C:\jdk1.5\bin\
```

```
C:\> set classpath = D:\ThucHanhJava\BT1\
```

Chương trình Hello World (tt)

TestGreeting.java:

```
import java.io.*;

public class TestGreeting
{
    public static void main(String[]
args)
    {
        Greeting gr = new Greeting();
        gr.greet();
    }
}
```

Greeting.java:

```
public class Greeting
{
    public void greet()
    {
        System.out.print("Hello World");
    }
}
```

Chương trình Hello World (tt)

- ▶ Biên dịch TestGreeting.java

javac TestGreeting.java

- ▶ Greeting.java được biên dịch tự động

- ▶ Thực hiện

java TestGreeting

- ▶ Kết quả

Hello World

Java Applets

- ▶ Được nhúng trong một ứng dụng khác (web browser)
- ▶ Có giao diện hạn chế (đồ họa)
- ▶ Không truy cập được tài nguyên của client

Applet đơn giản

Welcome.java:

```
// Java packages
import java.awt.Graphics;
import java.applet.Applet;
public class Welcome extends Applet
{
    public void paint(Graphics g)
    {
        // call superclass version of method paint
        super.paint(g);
        // draw a String
        g.drawString("Welcome to Java programming!", 25, 25);
    }
}
```


Nhúng vào trang web

```
<html>
```

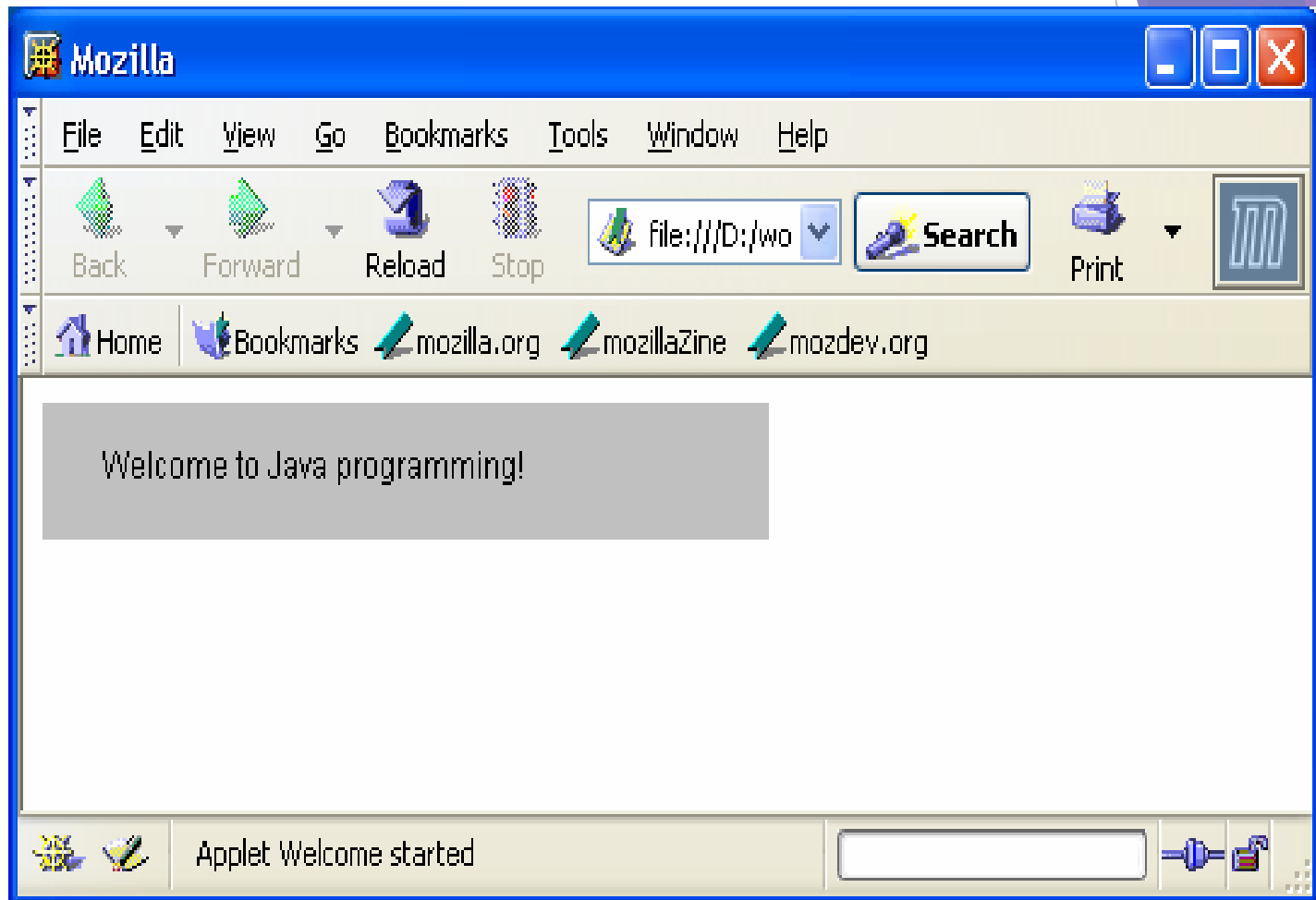
```
  <applet code = "Welcome.class"
```

```
    width = "300" height = "45">
```

```
  </applet>
```

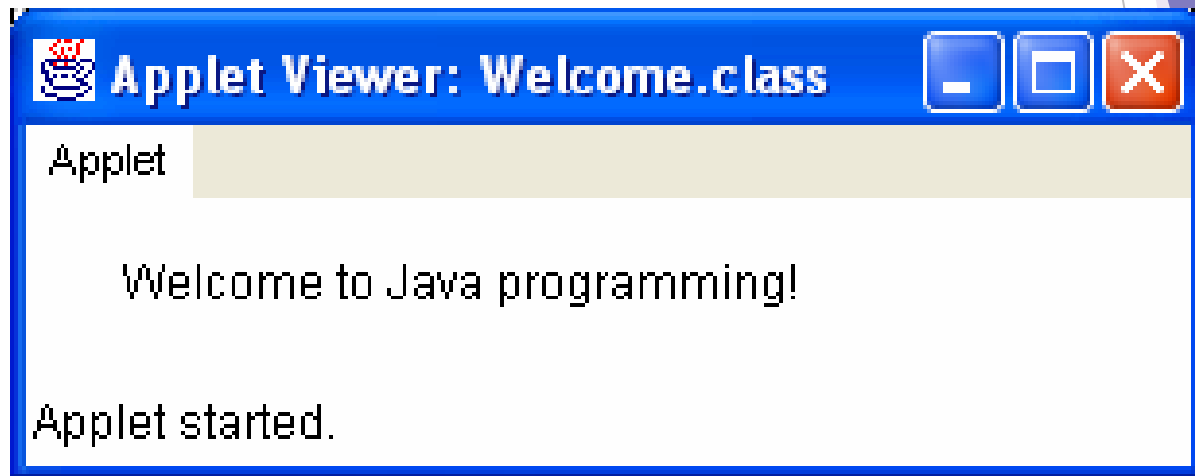
```
</html>
```

Thực hiện (trong web browser)



Thực hiện

appletviewer Welcome.html



Căn bản về ngôn ngữ Java

Nội dung

- Biến & Hằng
- Kiểu dữ liệu (kiểu cơ sở, kiểu tham chiếu)
- Toán tử, biểu thức
- Các cấu trúc điều khiển (chọn, rẽ nhánh, lặp)
- Lớp bao kiểu cơ sở
- Phương thức và cách sử dụng
- Một số ví dụ minh họa

Biến

- Biến là một vùng nhớ lưu các giá trị của chương trình
- Mỗi biến gắn với 1 kiểu dữ liệu và 1 định danh duy nhất là tên biến
- Tên biến phân biệt chữ hoa và chữ thường. Tên biến bắt đầu bằng 1 dấu `_`, `$`, hay 1 ký tự, không được bắt đầu bằng 1 ký số.
- ▶ **Khai báo**
`<kiểu dữ liệu> <tên biến>;`
`<kiểu dữ liệu> <tên biến> = <giá trị>;`
- ▶ **Gán giá trị**
`<tên biến> = <giá trị>;`
- ▶ **Lưu ý:** trong java nếu lúc khai báo không khởi tạo giá trị cho biến thì nó sẽ nhận 1 giá trị mặc định. Mỗi kiểu dữ liệu có 1 kiểu dữ liệu mặc định khác nhau.

Hằng

- Là một giá trị bất biến trong chương trình
- Tên đặt theo qui ước như tên biến
- Được khai báo dùng từ khóa **final**, và thường dùng tiếp vĩ ngữ đối với các hằng số (l, L, d, D, f, F)
- Ví dụ:

final int x = 10; // khai báo hằng số nguyên x = 10

final long y = 20L; // khai báo hằng số long y = 20

- Hằng ký tự: đặt giữa cặp nháy đơn “ ”
- Hằng chuỗi: là một dãy ký tự đặt giữa cặp nháy đôi “ ”

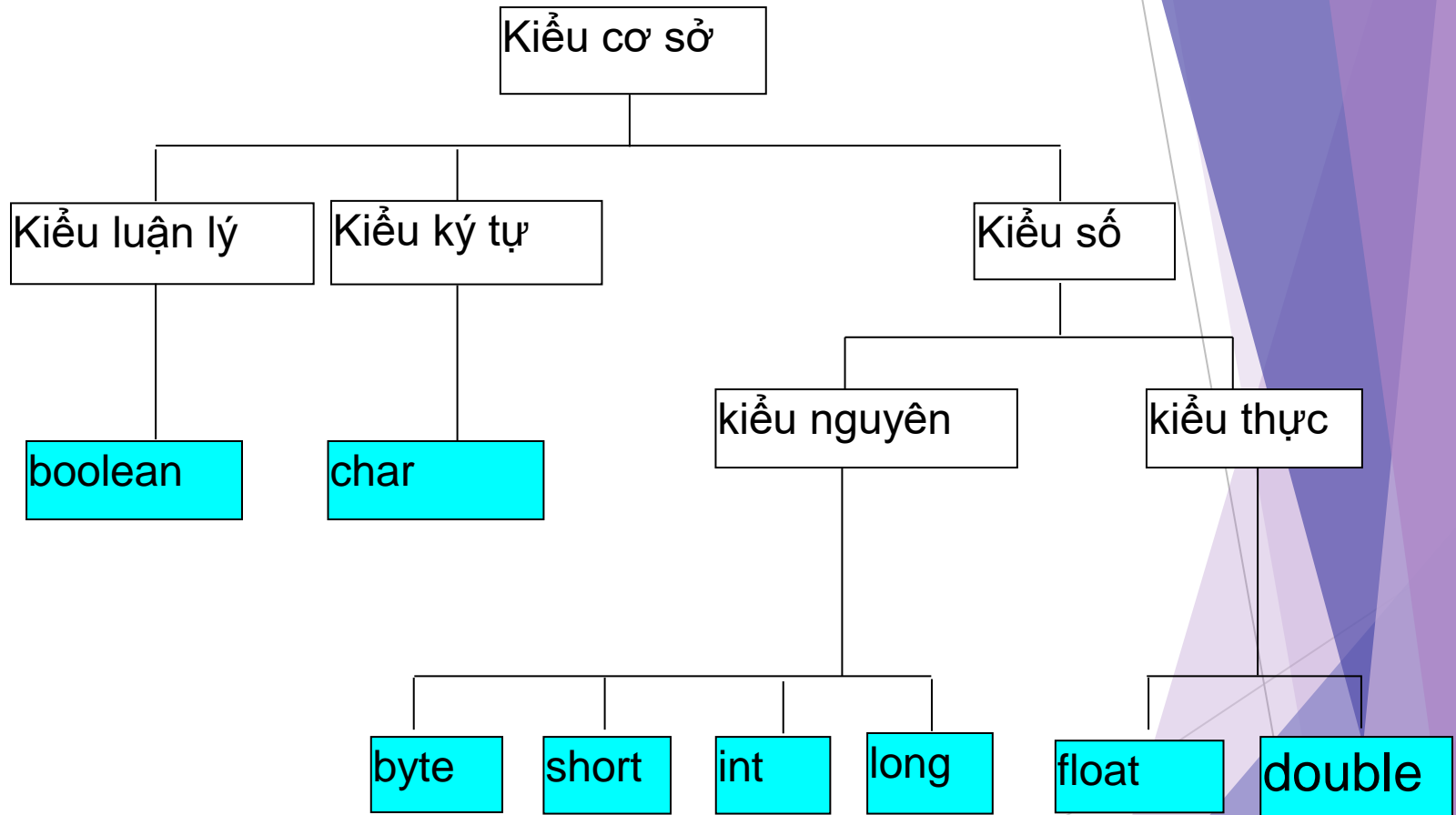
Hàng ký tự đặc biệt

Ký tự	Ý nghĩa
\b	Xóa lùi (BackSpace)
\t	Tab
\n	Xuống hàng
\r	Dấu enter
\"	Nháy kép
'	Nháy đơn
\\	\
\f	Đẩy trang
\uxxxx	Ký tự unicode

Kiểu dữ liệu

- Kiểu dữ liệu cơ sở (primitive data type)
- Kiểu dữ liệu tham chiếu (reference data type)

Kiểu dữ liệu cơ sở

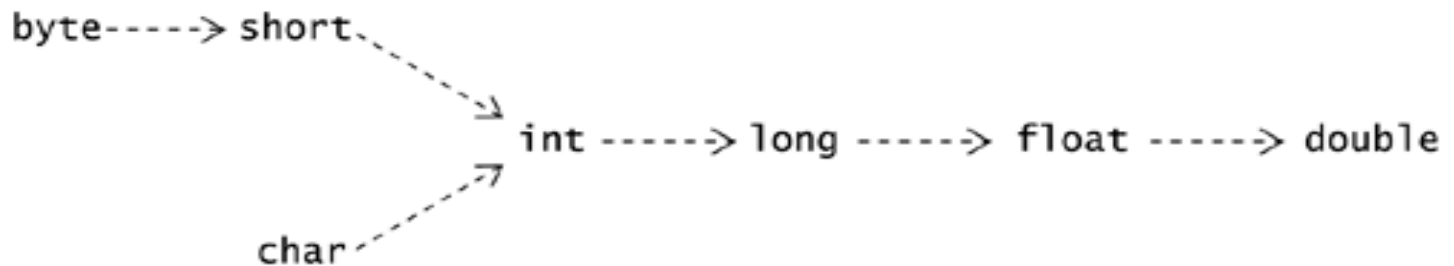


Kiểu dữ liệu cơ sở (tt)

Kiểu	Kích thước (bits)	Giá trị	Giá trị mặc định
boolean	[<i>Note:</i> The representation of a boolean is specific to the Java Virtual Machine on each computer platform.]	true và false	false
char	16	'\u0000' to '\uFFFF' (0 to 65535)	null
byte	8	-128 to +127 (-2^7 to $2^7 - 1$)	0
short	16	-32,768 to +32,767 (-2^{15} to $2^{15} - 1$)	0
int	32	-2,147,483,648 to +2,147,483,647 (-2^{31} to $2^{31} - 1$)	0
long	64	-9,223,372,036,854,775,808 to +9,223,372,036,854,775,807 (-2^{63} to $2^{63} - 1$)	0l
float	32	1.40129846432481707e-45 to 3.4028234663852886E+38	0.0f
double	64	4.94065645841246544e-324 to 1.7976931348623157E+308	0.0d

Kiểu dữ liệu cơ sở (tt)

- **Chuyển đổi kiểu dữ liệu:** khi có sự không tương thích về kiểu dữ liệu (gán, tính toán biểu thức, truyền đối số gọi phương thức)
 - ✓ Chuyển kiểu hẹp (lớn → nhỏ): ***cần ép kiểu***
<tên biến 2> = (kiểu dữ liệu) <tên biến 1>;
 - ✓ Chuyển kiểu rộng (nhỏ → lớn): ***tự động chuyển***



Kiểu dữ liệu cơ sở (tt)

- **Lưu ý**

1. Không thể chuyển đổi giữa kiểu boolean với int và ngược lại.

2. **Nếu** 1 toán hạng kiểu **double** thì

“Toán hạng kia chuyển thành **double**”

Nếu 1 toán hạng kiểu **float** thì

“Toán hạng kia chuyển thành **float**”

Nếu 1 toán hạng kiểu **long** thì

“Toán hạng kia chuyển thành **long**”

Ngược lại “Tất cả chuyển thành **int** để tính toán”

Kiểu dữ liệu cơ sở (tt)

- **Ví dụ minh họa**

1. *byte* $x = 5$;

2. *byte* $y = 10$;

3. *byte* $z = x + y$;

// Dòng lệnh thứ 3 báo lỗi chuyển kiểu cần sửa lại

// byte z = (byte) (x + y);

Kiểu dữ liệu tham chiếu

- **Kiểu mảng**

- ✓ Mảng là tập hợp các phần tử có cùng tên và cùng kiểu dữ liệu.
- ✓ Mỗi phần tử được truy xuất thông qua chỉ số

- **Khai báo mảng**

<kiểu dữ liệu>[] <tên mảng>; // mảng 1 chiều

<kiểu dữ liệu> <tên mảng>[]; // mảng 1 chiều

<kiểu dữ liệu>[][] <tên mảng>; // mảng 2 chiều

<kiểu dữ liệu> <tên mảng>[][]; // mảng 2 chiều

Kiểu dữ liệu tham chiếu (tt)

- Khởi tạo

```
int arrInt[] = {1, 2, 3};
```

```
char arrChar[] = {'a', 'b', 'c'};
```

```
String arrString[] = {"ABC", "EFG", "GHI"};
```

- Cấp phát & truy cập mảng

```
int arrInt = new int[100];
```

int arrInt[100]; // Khai báo này trong Java sẽ bị báo lỗi.

Chỉ số mảng *n* phần tử: từ **0** đến ***n*-1**

Kiểu dữ liệu tham chiếu (tt)

- Kiểu đối tượng
 - ▶ Khai báo đối tượng
<Kiểu đối tượng> <biến ĐT>;
 - ▶ Khởi tạo đối tượng
*<Kiểu đối tượng> <biến ĐT> = new
<Kiểu đối tượng>;*
 - ▶ Truy xuất thành phần đối tượng
<biến ĐT>.<thuộc tính>
<biến ĐT>.<phương thức>

Toán tử, biểu thức

► Toán tử số học

Toán tử	Ý nghĩa
+	Cộng
-	Trừ
*	Nhân
/	Chia nguyên
%	Chia dư
++	Tăng 1
--	Giảm 1

Toán tử, biểu thức (tt)

► Phép toán trên bit

Toán tử	Ý nghĩa
&	AND
	OR
^	XOR
<<	Dịch trái
>>	Dịch phải
~	Bù bit

Toán tử, biểu thức (tt)

► Toán tử quan hệ và logic

Toán tử	Ý nghĩa
<code>==</code>	So sánh bằng
<code>!=</code>	So sánh khác
<code>></code>	So sánh lớn hơn
<code><</code>	So sánh nhỏ hơn
<code>>=</code>	So sánh lớn hơn hay bằng
<code><=</code>	So sánh nhỏ hơn hay bằng
<code> </code>	OR (biểu thức logic)
<code>&&</code>	AND (biểu thức logic)
<code>!</code>	NOT (biểu thức logic)

Toán tử, biểu thức (tt)

► Toán tử gán

Toán tử	Ví dụ	Ý nghĩa
=	$a = b$	gán $a = b$
$+=$	$a += 5$	$a = a + 5$
$-=$	$b -= 10$	$b = b - 10$
$*=$	$c *= 3$	$c = c * 3$
$/=$	$d /= 2$	$d = d / 2$
$\% =$	$e \% = 4$	$e = e \% 4$

Toán tử, biểu thức (tt)

► Toán tử điều kiện

Cú pháp: <điều kiện> ? <biểu thức 1> : <biểu thức 2>

► Ví dụ:

► *int x = 10;*

► *int y = 20;*

► *int Z = (x < y) ? 30 : 40;*

► *// Kết quả z = 30 do biểu thức (x < y) là đúng.*

Cấu trúc điều khiển

- **Cấu trúc *if ... else***

- ▶ **Dạng 1:** *if* (<điều_kiện>) {
 <khối_lệnh>;
 }

- ▶ **Dạng 2:** *if* (<điều_kiện>) {
 <khối_lệnh1>;
 }
 else {
 <khối_lệnh2>;
 }

Cấu trúc điều khiển (tt)

- Cấu trúc *switch ... case*

```
► switch (<biến>
{  case <giá trị_1>:
        <khối_lệnh_1>;
        break;

    ....
    case <giá trị_n>:
        <khối_lệnh_n>;
        break;
    default:
        <khối_lệnh_default>;
}
```


Cấu trúc điều khiển (tt)

- Cấu trúc lặp

- Dạng 1:

```
while (<điều_kiện_lặp>) {  
    <khối_lệnh>;  
}
```

- Dạng 2:

```
do {  
    <khối_lệnh>;  
} while (điều_kiện);
```

- Dạng 3:

```
for (khởi_tạo_biến_đếm;đk_lặp;tăng_biến) {  
    <khối_lệnh>;  
}
```

Cấu trúc điều khiển (tt)

- **Cấu trúc lệnh nhảy jump:** dùng kết hợp nhãn (label) với từ khóa *break* và *continue* để thay thế cho lệnh *goto* (trong C).

► Ví dụ:

```
label:
for (...) {
    for (...) {
        if (<biểu thức điều kiện>)
            break label;
        else
            continue label;
    }
}
```

Lớp bao kiểu dữ liệu cơ sở

Data type	Wrapper Class (java.lang.*)	Ghi chú
boolean	Boolean	<ul style="list-style-type: none">- Gói (package): chứa nhóm nhiều class.- Ngoài các Wrapper Class, gói java.lang còn cung cấp các lớp nền tảng cho việc thiết kế ngôn ngữ java như: String, Math, ...
byte	Byte	
short	Short	
char	Character	
int	Integer	
long	Long	
float	Float	
double	Double	

Q/A