**An introduction to java**

**Trong chương này**

* + 1.1 Java làm nền tảng lập trình, trang 1
  + 1.2 Từ ngữ Java ‘White Paper’, trang 2
  + 1.4 Java Applet và Internet, trang 8
  + 1.3 Lịch sử ngắn gọn về Java, trang 10
  + 1.5 Các quan niệm sai lầm phổ biến về Java, trang 13

Bản phát hành đầu tiên của Java vào năm 1996 đã tạo ra một sự phấn khích đáng kinh ngạc,

không chỉ trên báo chí máy tính, mà trên các phương tiện truyền thông chính thống như New York

Times, Washington Post và BusinessWeek. Java có sự khác biệt là

ngôn ngữ lập trình đầu tiên và duy nhất có câu chuyện dài 10 phút trên National

Đài phát thanh công cộng. Một quỹ đầu tư mạo hiểm 100.000.000 đô la được thành lập chỉ dành cho các sản phẩm

sử dụng một ngôn ngữ máy tính cụ thể. Tôi hy vọng bạn sẽ thích lịch sử ngắn gọn của Java

mà bạn sẽ tìm thấy trong chương này.

* 1. java programming platform

Trong ấn bản đầu tiên của cuốn sách này, đồng tác giả của tôi, Gary Cornell và tôi đã viết cái này

Về java:

“Là một ngôn ngữ máy tính, sự cường điệu của Java là quá mức: Java chắc chắn là một ngôn ngữ lập trình tốt. Không có nghi ngờ gì rằng nó là một trong những ngôn ngữ tốt hơn

1 có sẵn cho các lập trình viên nghiêm túc. Chúng tôi nghĩ rằng nó có thể là một

ngôn ngữ lập trình, nhưng có lẽ đã quá muộn cho điều đó. Một khi một ngôn ngữ là

ngoài hiện trường, thực tế xấu xí về khả năng tương thích với các bộ mã hiện có trong. "

Biên tập viên của chúng tôi đã nhận được rất nhiều sự ngạc nhiên cho đoạn văn này từ một người rất cao ở Sun

Microsystems, công ty ban đầu phát triển Java. Ngôn ngữ Java

có rất nhiều tính năng hay mà chúng ta sẽ xem xét chi tiết ở phần sau của chương này. Nó có

tỷ lệ mụn cóc của nó và một số bổ sung mới hơn cho ngôn ngữ không bằng

thanh lịch như các tính năng ban đầu vì thực tế xấu xí của khả năng tương thích.

Nhưng, như chúng ta đã nói trong lần xuất bản đầu tiên, Java không bao giờ chỉ là một ngôn ngữ. Ở đó

có rất nhiều ngôn ngữ lập trình, nhưng ít ngôn ngữ trong số đó tạo ra nhiều

bắn tung tóe. Java là một nền tảng toàn diện, với một thư viện khổng lồ, chứa rất nhiều

mã và môi trường thực thi cung cấp các dịch vụ như bảo mật,

khả năng di động trên các hệ điều hành và thu gom rác tự động.

Là một lập trình viên, bạn sẽ muốn một ngôn ngữ có cú pháp dễ chịu và ngữ nghĩa dễ hiểu (tức là không phải C ++). Java phù hợp với dự luật, cũng như hàng tá điểm tốt khác

ngôn ngữ. Một số ngôn ngữ cung cấp cho bạn tính di động, thu gom rác và những thứ tương tự,

nhưng họ không có nhiều thư viện, buộc bạn phải tự triển khai nếu muốn

đồ họa ưa thích hoặc mạng hoặc truy cập cơ sở dữ liệu. Chà, Java có mọi thứ — a

ngôn ngữ tốt, môi trường thực thi chất lượng cao và thư viện rộng lớn.

Sự kết hợp đó là điều khiến Java trở thành một mệnh đề không thể cưỡng lại đối với rất nhiều

lập trình viên.

* 1. The java “white paper” buzzwords ( từ ngữ sách trắng trong java)

Các tác giả của Java đã viết một sách trắng có ảnh hưởng giải thích thiết kế của họ

mục tiêu và thành tích. Họ cũng xuất bản một tổng quan ngắn hơn

được tổ chức theo 11 từ thông dụng sau:

1. Đơn giản

2. Hướng đối tượng

3. Phân tán

4. Mạnh mẽ

5. Bảo mật

6. Kiến trúc-Trung lập

7. Di động

8. Thông dịch

9. Hiệu suất cao

10. Đa luồng

11. Động

Trong phần này, bạn sẽ tìm thấy một bản tóm tắt, với các đoạn trích từ sách trắng, về

những gì các nhà thiết kế Java nói về mỗi từ thông dụng, cùng với phần bình luận

dựa trên kinh nghiệm của tôi với phiên bản Java hiện tại

* 1. A shrot history of Java ( lịch sử ngắn về java)

Phần này cung cấp một lịch sử ngắn về sự phát triển của Java. Nó dựa trên nhiều nguồn đã xuất bản khác nhau (quan trọng nhất là cuộc phỏng vấn với những người sáng tạo của Java vào tháng 7

Tạp chí trực tuyến của SunWorld phát hành năm 1995).

Java quay trở lại năm 1991, khi một nhóm kỹ sư Sun, dẫn đầu là Patrick Naughton

và James Gosling (một Sun Fellow và một thuật sĩ máy tính toàn diện), muốn

để thiết kế một ngôn ngữ máy tính nhỏ có thể được sử dụng cho các thiết bị tiêu dùng

như hộp chuyển mạch truyền hình cáp. Vì các thiết bị này không có nhiều điện năng hoặc

bộ nhớ, ngôn ngữ phải nhỏ và tạo ra mã rất chặt chẽ. Ngoài ra, như

các nhà sản xuất khác nhau có thể chọn các đơn vị xử lý trung tâm (CPU) khác nhau, nó

điều quan trọng là ngôn ngữ không bị ràng buộc với bất kỳ kiến ​​trúc đơn lẻ nào. Dự án

có tên mã là “Green”.

Các yêu cầu đối với mã nhỏ, chặt chẽ và trung lập với nền tảng đã khiến nhóm thiết kế

một ngôn ngữ di động tạo ra mã trung gian cho một máy ảo.

Người Mặt trời đến từ nền UNIX, vì vậy họ dựa trên ngôn ngữ của họ

C ++ chứ không phải Lisp, Smalltalk hoặc Pascal. Nhưng, như Gosling nói trong cuộc phỏng vấn,

"Nói chung, ngôn ngữ là một công cụ, không phải là mục đích cuối cùng." Gosling quyết định gọi

ngôn ngữ “Oak” (có lẽ vì anh ấy thích dáng vẻ của một cây sồi

ngay bên ngoài cửa sổ của anh ấy tại Sun). Những người ở Sun sau đó nhận ra rằng Oak

là tên của một ngôn ngữ máy tính hiện có, vì vậy họ đã đổi tên thành

Java. Đây hóa ra là một sự lựa chọn đầy cảm hứng.

Năm 1992, dự án Xanh đã cung cấp sản phẩm đầu tiên của mình, được gọi là “\* 7”. Đó là một điều khiển từ xa cực kỳ thông minh. Thật không may, không ai quan tâm đến việc sản xuất

điều này ở Sun, và những người Xanh phải tìm những cách khác để tiếp thị công nghệ của họ. Tuy nhiên, không một công ty điện tử tiêu dùng nào quan tâm. Sau đó nhóm đấu thầu một dự án thiết kế hộp truyền hình cáp có thể

đối phó với các dịch vụ cáp mới nổi như video theo yêu cầu. Họ đã không nhận được

hợp đồng. (Thật thú vị, công ty đã được dẫn dắt bởi cùng một Jim Clark

người đã thành lập Netscape — một công ty đã làm rất nhiều để tạo nên thành công cho Java.)

Dự án Xanh (với tên mới là “First Person, Inc.”) đã dành toàn bộ năm 1993 và

nửa năm 1994 tìm kiếm mọi người mua công nghệ của nó. Không ai được tìm thấy. (Patrick

Naughton, một trong những người sáng lập của nhóm và là người cuối cùng đã làm

most of the marketing, claims to have accumulated 300,000 air miles in trying to

bán công nghệ.) First Person đã bị giải thể vào năm 1994.

Trong khi tất cả những điều này đang diễn ra tại Sun, World Wide Web một phần của Internet

đang phát triển lớn hơn và lớn hơn. Chìa khóa của World Wide Web là

trình duyệt dịch các trang siêu văn bản ra màn hình. Vào năm 1994, hầu hết mọi người đang sử dụng

Mosaic, một trình duyệt web phi thương mại ra đời từ siêu máy tính

* 1. Java applets and the intrenet

Ý tưởng ở đây rất đơn giản: Người dùng sẽ tải xuống các mã byte Java từ Internet

và chạy chúng trên máy của riêng họ. Các chương trình Java hoạt động trên các trang web là

được gọi là applet. Để sử dụng một applet, bạn chỉ cần một trình duyệt web hỗ trợ Java,

sẽ thực thi các mã bytecodes cho bạn. Bạn không cần cài đặt bất kỳ phần mềm nào. Bạn lấy

phiên bản mới nhất của chương trình bất cứ khi nào bạn truy cập trang web chứa

applet. Quan trọng nhất, nhờ sự bảo mật của máy ảo, bạn không bao giờ

cần phải lo lắng về các cuộc tấn công từ mã thù địch.

Chèn một applet vào một trang web hoạt động giống như nhúng một hình ảnh. Các

applet trở thành một phần của trang và văn bản chảy xung quanh không gian được sử dụng cho

applet. Vấn đề là, hình ảnh này sống động. Nó phản ứng với các lệnh của người dùng, các thay đổi

sự xuất hiện của nó và trao đổi dữ liệu giữa máy tính trình bày applet

và máy tính phục vụ nó.

8 Chương 1 Giới thiệu về Java Hình 1.1 cho thấy một ví dụ điển hình về một trang web động thực hiện các phép tính phức tạp. Ứng dụng Jmol hiển thị các cấu trúc phân tử. Bằng cách sử dụng

chuột, bạn có thể xoay và phóng to từng phân tử để hiểu rõ hơn cấu trúc của nó.

Loại thao tác trực tiếp này không thể thực hiện được với các trang web tĩnh, nhưng

applet làm cho nó có thể. (Bạn có thể tìm thấy applet này tại http://jmol.sourceforge.net.)

Khi các applet lần đầu tiên xuất hiện, chúng đã tạo ra một lượng lớn sự thích thú. Nhiều

mọi người tin rằng sự lôi cuốn của các applet là nguyên nhân dẫn đến sự phổ biến đáng kinh ngạc của Java. Tuy nhiên, sự phấn khích ban đầu sớm chuyển thành thất vọng. Đa dạng

các phiên bản của trình duyệt Netscape và Internet Explorer chạy các phiên bản khác nhau

của Java, một số trong số đó đã lỗi thời nghiêm trọng. Tình huống đáng tiếc này khiến việc phát triển các applet tận dụng những lợi thế hiện tại ngày càng khó khăn.

Phiên bản Java. Thay vào đó, công nghệ Flash của Adobe trở nên phổ biến để đạt được

hiệu ứng động trong trình duyệt. Sau đó, khi Java bị bảo mật nghiêm trọng

các vấn đề, trình duyệt và trình cắm thêm của trình duyệt Java ngày càng trở nên hạn chế.

Ngày nay, đòi hỏi kỹ năng và sự cống hiến để các applet hoạt động trong trình duyệt của bạn.

Ví dụ: nếu bạn truy cập trang web Jmol, bạn có thể sẽ gặp một thông báo

khuyến khích bạn định cấu hình trình duyệt của mình để cho phép các applet chạy.

* 1. Common Misconceptions about Java

Chương này kết thúc với một danh sách bình luận về một số quan niệm sai lầm phổ biến

Về java.

Java là một phần mở rộng của HTML.

Java là một ngôn ngữ lập trình; HTML là một cách để mô tả cấu trúc của một

trang web. Chúng không có điểm chung nào ngoại trừ việc có các phần mở rộng HTML

để đặt các ứng dụng Java trên một trang web.

Tôi sử dụng XML, vì vậy tôi không cần Java.

Java là một ngôn ngữ lập trình; XML là một cách để mô tả dữ liệu. Bạn có thể xử lý

Dữ liệu XML với bất kỳ ngôn ngữ lập trình nào, nhưng API Java có chứa

hỗ trợ xử lý XML. Ngoài ra, nhiều công cụ XML quan trọng là

được thực hiện trong Java. Xem Tập II để biết thêm thông tin.

Java là một ngôn ngữ lập trình dễ học.

Không có ngôn ngữ lập trình nào mạnh mẽ như Java là dễ dàng. Bạn luôn phải phân biệt giữa mức độ dễ dàng để viết các chương trình đồ chơi và mức độ khó thực hiện

công việc nghiêm túc. Ngoài ra, hãy xem xét rằng chỉ có bảy chương trong cuốn sách này thảo luận về Java

ngôn ngữ. Các chương còn lại của cả hai tập đều hướng dẫn cách đặt ngôn ngữ

để làm việc bằng cách sử dụng các thư viện Java. Các thư viện Java chứa hàng nghìn lớp

và các giao diện và hàng chục nghìn chức năng. May mắn thay, bạn không cần phải

biết mọi thứ trong số họ, nhưng bạn cần phải biết nhiều điều đáng ngạc nhiên để sử dụng Java

cho bất cứ điều gì thực tế.

Java sẽ trở thành một ngôn ngữ lập trình chung cho tất cả các nền tảng.

Điều này là có thể trên lý thuyết. Nhưng trên thực tế, có những miền mà các ngôn ngữ khác

đang cố thủ. Objective C và người kế nhiệm của nó, Swift, sẽ không bị thay thế

trên thiết bị iOS. Bất cứ điều gì xảy ra trong trình duyệt đều được kiểm soát bởi JavaScript.

Các chương trình Windows được viết bằng C ++ hoặc C #. Java có lợi thế ở phía máy chủ

lập trình và trong các ứng dụng khách đa nền tảng.

Java chỉ là một ngôn ngữ lập trình khác.

Java là một ngôn ngữ lập trình tốt đẹp; hầu hết các lập trình viên thích nó hơn C, C ++ hoặc

C #. Nhưng đã có hàng trăm ngôn ngữ lập trình tuyệt vời mà không bao giờ

đã trở nên phổ biến rộng rãi, trong khi các ngôn ngữ có sai sót rõ ràng, chẳng hạn như

C ++ và Visual Basic, đã rất thành công.

Tại sao? Sự thành công của một ngôn ngữ lập trình được xác định nhiều hơn bởi

tiện ích của hệ thống hỗ trợ xung quanh nó hơn là bởi sự sang trọng của cú pháp của nó. Chúng tôi

có các thư viện hữu ích, tiện lợi và tiêu chuẩn cho các tính năng mà bạn cần

triển khai thực hiện? Có nhà cung cấp công cụ nào xây dựng chương trình và gỡ lỗi tuyệt vời không môi trường? Ngôn ngữ và bộ công cụ có tích hợp với phần còn lại của

cơ sở hạ tầng điện toán? Java thành công vì các thư viện của nó cho phép bạn dễ dàng

làm những việc như mạng, ứng dụng web và đồng thời. Thực tế là

Java giảm lỗi con trỏ là một phần thưởng, vì vậy các lập trình viên dường như

hiệu quả với Java — nhưng những yếu tố này không phải là nguồn gốc thành công của nó.

Java là độc quyền và do đó cần tránh sử dụng nó.

Khi Java lần đầu tiên được tạo ra, Sun đã cấp giấy phép miễn phí cho các nhà phân phối và người dùng cuối.

Mặc dù Sun có quyền kiểm soát cuối cùng đối với Java, nhưng họ liên quan đến nhiều công ty khác

trong việc phát triển các bản sửa đổi ngôn ngữ và thiết kế các thư viện mới. Nguồn

mã cho máy ảo và các thư viện luôn có sẵn miễn phí,

nhưng chỉ để kiểm tra, không phải để sửa đổi và phân phối lại. Java đã “đóng cửa

nguồn, nhưng chơi đẹp. ”

Tình hình này đã thay đổi đáng kể vào năm 2007, khi Sun công bố rằng tương lai

các phiên bản của Java sẽ có sẵn theo Giấy phép Công cộng (GPL),

cùng một giấy phép nguồn mở được sử dụng bởi Linux. Oracle đã cam kết giữ

Mã nguồn mở Java. Chỉ có một con ruồi trong thuốc mỡ — bằng sáng chế. Mọi người là

được cấp bằng sáng chế để sử dụng và sửa đổi Java, tuân theo GPL, nhưng chỉ trên

nền tảng máy tính để bàn và máy chủ. Nếu bạn muốn sử dụng Java trong các hệ thống nhúng, bạn

cần một giấy phép khác và có thể sẽ phải trả tiền bản quyền. Tuy nhiên,

các bằng sáng chế này sẽ hết hạn trong vòng một thập kỷ tới và tại thời điểm đó Java sẽ

hoàn toàn miễn phí.

Java được thông dịch, vì vậy nó quá chậm đối với các ứng dụng nghiêm trọng.

Trong những ngày đầu của Java, ngôn ngữ này đã được thông dịch. Ngày nay, máy ảo Java sử dụng một trình biên dịch đúng lúc. Các "điểm nóng" trong mã của bạn sẽ chạy

trong Java cũng nhanh như trong C ++ và trong một số trường hợp, thậm chí còn nhanh hơn.

Mọi người từng phàn nàn rằng các ứng dụng Java trên máy tính để bàn chạy chậm. Tuy nhiên, ngày hôm nay

máy tính nhanh hơn nhiều so với khi những lời phàn nàn này bắt đầu. Chậm

Chương trình Java ngày nay vẫn sẽ chạy khá tốt hơn một chút so với C ++ nhanh như vũ bão

chương trình đã thực hiện một vài năm trước đây.

Tất cả các chương trình Java đều chạy bên trong một trang web.

Tất cả các ứng dụng Java đều chạy bên trong trình duyệt web. Đó là định nghĩa của một applet — a

Chương trình Java chạy bên trong trình duyệt. Nhưng hầu hết các chương trình Java là độc lập

các ứng dụng chạy bên ngoài trình duyệt web. Trên thực tế, nhiều chương trình Java chạy

trên máy chủ web và tạo mã cho các trang web.

Các chương trình Java là một rủi ro bảo mật lớn.

Trong những ngày đầu của Java, có một số báo cáo được công bố rộng rãi về các lỗi trong

hệ thống bảo mật Java. Các nhà nghiên cứu đã xem nó như một thách thức để tìm ra những chú gà con

trong bộ giáp Java và thách thức sức mạnh và sự tinh vi của bảo mật applet mô hình. Các lỗi kỹ thuật mà họ phát hiện được đều đã được khắc phục nhanh chóng.

Sau đó, có nhiều vụ khai thác nghiêm trọng hơn, mà Sun, và sau này là Oracle, đã phản hồi

quá chậm. Các nhà sản xuất trình duyệt đã phản ứng, và có lẽ đã phản ứng quá mức, bằng cách tắt Java theo mặc định. Để giữ điều này trong quan điểm, hãy xem xét hàng triệu

các cuộc tấn công của vi-rút trong các tệp thực thi Windows và macro Word gây ra

đau buồn nhưng ít chỉ trích đáng ngạc nhiên về những điểm yếu của nền tảng bị tấn công.

Một số quản trị viên hệ thống thậm chí đã vô hiệu hóa Java trong các trình duyệt của công ty,

trong khi tiếp tục cho phép người dùng của họ tải xuống các tệp thực thi và Word

các tài liệu gây ra rủi ro lớn hơn nhiều. Ngay cả 20 năm sau khi được tạo ra, Java vẫn

an toàn hơn nhiều so với bất kỳ nền tảng thực thi thông thường nào khác.

JavaScript là một phiên bản đơn giản hơn của Java.

JavaScript, một ngôn ngữ kịch bản có thể được sử dụng bên trong các trang web, đã được phát minh

bởi Netscape và ban đầu được gọi là LiveScript. JavaScript có cú pháp gợi nhớ đến Java và tên của các ngôn ngữ nghe giống nhau, nhưng nếu không thì

không liên quan. Một tập hợp con của JavaScript được chuẩn hóa là ECMA-262. JavaScript nhiều hơn

tích hợp chặt chẽ với các trình duyệt hơn so với các ứng dụng Java. Đặc biệt, một JavaScript

chương trình có thể sửa đổi tài liệu đang được hiển thị, trong khi một applet

chỉ có thể kiểm soát sự xuất hiện của một khu vực giới hạn.

Với Java, tôi có thể thay thế máy tính để bàn của mình bằng một “thiết bị Internet” rẻ tiền.

Khi Java lần đầu tiên được phát hành, một số người đặt cược lớn rằng điều này sẽ xảy ra.

Các công ty đã sản xuất các nguyên mẫu của máy tính mạng hỗ trợ Java, nhưng người dùng

đã không sẵn sàng từ bỏ một máy tính để bàn mạnh mẽ và tiện lợi cho một máy hạn chế không có bộ nhớ cục bộ. Ngày nay, tất nhiên, thế giới đã thay đổi, và

đối với phần lớn người dùng cuối, nền tảng quan trọng là điện thoại di động hoặc

máy tính bảng. Phần lớn các thiết bị này được điều khiển bởi nền tảng Android,

là một dẫn xuất của Java. Học lập trình Java sẽ giúp bạn

Lập trình Android cũng vậy.