LẬP TRÌNH HÀM VỚI JAVA 8?

Đã có rất nhiều thảo luận về cách "Java 8 được đưa vào lập trình Java" - nhưng bên trong câu nói đó ý nghĩa là gì?

Trong bài viết này, tôi sẽ nói rằng nó là một ngôn ngữ, hoặc một phong cách lập trình, có được hàm. Bằng cách nhìn vào sự phát triển của Java - đặc biệt là hệ thống kiểu của nó, chúng ta có thể thấy các tính năng mới của Java 8, đặc biệt là các biểu thức lambda, thay đổi cảnh quan, và cung cấp một số lợi ích quan trọng của phong cách hàm - trước khi giải quyết các câu hỏi: "Làm thế nào hàm Java là 8? ".

# Một ngôn ngữ lập trình hàm là gì?

## Giới thiệu

Một ngôn ngữ lập trình hàm là một trong những giao dịch với mã trong cùng một cách như là dữ liệu. Điều này có nghĩa rằng một hàm nên có một giá trị hạng nhất và có thể được gán cho các biến, thông qua hàm…

## Lập trình hàm được sử dụng khi nào?

Nhiều ngôn ngữ hàm thậm chí còn đi xa hơn, và được xem như tính toán và các thuật toán cơ bản hơn so với các dữ liệu mà họ hoạt động trên đó. Một số các ngôn ngữ muốn gỡ chương trình nhà nước từ hàm (trong một cách mà có vẻ mâu thuẫn với mong muốn của các ngôn ngữ hướng đối tượng thường muốn mang lại cho họ xích lại gần nhau).

Lập trình hàm hữu dụng nhất khi chương trình được viết bằng về hàm và luôn luôn trả về kết quả tương tự cho một đầu vào cho trước (không phân biệt của bất kỳ sự xuất hiện của vấn đề trong các chương trình đang chạy) và không gây ra bất kỳ tác dụng khác hoặc thay đổi bất kỳ chương trình. Hàm tuân thủ này đôi khi được gọi là hàm "tinh khiết", và chúng thực thi theo cùng một cách mà các hàm toán học làm.

Ưu điểm lớn nhất là hàm tinh khiết là chúng được dễ dàng thực hiện hơn vì lý do hoạt động của nó không phụ thuộc vào yếu tố bên ngoài. Hàm có thể dễ dàng kết hợp với nhau - và điều này có thể được nhìn thấy trong phong cách công việc phát triển như REPL (Read, Execute, In, Loop) phong cách chung cho các phương ngữ Lisp và các ngôn ngữ khác với di sản hàm mạnh mẽ.

# Lập trình hàm trong ngôn ngữ NON\_FP

Cho dù một ngôn ngữ là hàm hay không không phải là một điều kiện nhị phân - thay vào đó, ngôn ngữ tồn tại trên một quang phổ. Lựa chọn cuối cùng là ngôn ngữ cơ bản thực thi các hàm lập trình, thường bằng cách cấm các cấu trúc dữ liệu có thể thay đổi. Clojure là một ví dụ của một ngôn ngữ không cho phép dữ liệu có thể thay đổi trong ý thức chấp nhận.

Tuy nhiên, có những ngôn ngữ khác, trong đó người ta thường viết các chương trình trong một phong cách hàm, mặc dù ngôn ngữ không thực thi này. Một ví dụ là Scala, mà là một sự pha trộn của ngôn ngữ hướng đối tượng và hàm. Nó cho phép các hàm như các giá trị, chẳng hạn như trong khi giữ lại lớp học và cú pháp đối tượng đó là rất gần với Java.

# Hệ thống kiểu ban đầu của Java

Hệ thống kiểu ban đầu của Java tính đến hiện tại đã hơn 15 năm. Nó là đơn giản và rõ ràng - loại là kiểu tham chiếu hoặc các loại nguyên thủy. Các loại tài liệu tham khảo là các lớp, giao diện hoặc mảng.Từ những ngày đầu tiên, hệ thống loại Java đã nhất quyết một điểm rất quan trọng - mỗi loại phải có một cái tên mà nó có thể được gọi bằng. Điều này được gọi là "gõ nominative" - ​​và Java là một ngôn ngữ mạnh mẽ nominatively gõ.

# Java 5 - loại hệ thống

Việc phát hành của Java 5 mang ba tính năng chính mới cho các loại hệ thống - đếm, chú thích và các loại chung chung.

1. Kiểu liệt kê (đếm).
2. Chú thích liên quan đến giao diện.
3. Cung cấp các loại tham số.

Trong số các tính năng được giới thiệu bởi Java 5, đếm và chú thích cung cấp các hình thức mới của các loại tài liệu tham khảo mà cần điều trị đặc biệt bởi trình biên dịch, và đó là một cách hiệu quả tách rời khỏi hệ thống phân loại hiện có.

# Các tính năng được giới thiệu trong Java 6 & 7

Java 6 thực chất là hoạt động phát hành và thư viện nâng cao. Sự thay đổi duy nhất để loại hệ thống là một mở rộng vai trò của các chú thích, và phát hành của một khả năng để xử lý chú thích. Điều này hầu hết không ảnh hưởng đến các nhà phát triển Java, và không thực sự cung cấp cho các hệ thống loại cắm trong Java 6. Java 7 không thay đổi loại vật chất hệ thống. Những tính năng mới duy nhất, tất cả đều rất nhỏ, là:

* Cải thiện nhỏ trong suy luận kiểu trong trình biên dịch javac.
* Chữ ký đa hình văn - sử dụng như một chi tiết thực hiện cho tính năng gọi là phương pháp xử lý - được lần lượt sử dụng để thực hiện các biểu thức lambda trong Java 8.
* Multi-catch cung cấp một số dấu vết nhỏ của "các kiểu dữ liệu đại số" - nhưng đây là hoàn toàn là nội bộ của javac và không phải của bất kỳ đều hiệu quả thực sự cho các lập trình viên sử dụng.

# Java 8 loại hệ thống

Trong suốt lịch sử của mình, Java đã được cơ bản được xác định bởi hệ thống kiểu của nó. Nó là trung tâm ngôn ngữ và đã duy trì được tuân thủ nghiêm ngặt gõ nominative. Từ một quan điểm thực tế, các loại hệ thống Java không thay đổi nhiều giữa Java 5 và Java 7. Ngay từ cái nhìn đầu tiên, chúng ta có thể mong đợi Java 8 để thay đổi điều đó.

Tác giả: **Ben Evans**