# [[Học nghề] Thực hành, Thực hành, Thực hành](https://codegym.vn/blog/2018/09/12/hoc-nghe-thuc-hanh-thuc-hanh-thuc-hanh/)



*Những người mà chúng ta biết tới là những bậc thầy không chỉ cống hiến hết mình cho kỹ năng đặc biệt để trở nên giỏi hơn ở kỹ năng đó. Sự thực là, họ thích luyện tập – và do đó họ sẽ trở nên giỏi hơn. Sau đó, để hoàn thành cái vòng tròn ấy, họ càng giỏi hơn thì họ lại càng thích thực hiện những thao tác cơ bản lặp đi lặp lại nhiều lần.*

*-George Leonard, Mastery-*

## **Bối cảnh**

Bạn muốn làm tốt hơn những điều bạn làm và bạn muốn phát triển Kỹ Năng Cụ Thể trong các lĩnh vực mới.

## **Vấn đề**

Việc thực hiện các hoạt động lập trình hàng ngày không cho bạn cơ hội được học bằng cách mắc sai lầm. Nó giống như thể bạn luôn ở trên sân khấu.

## **Giải pháp**

Dành thời gian để luyện tập nghề của bạn mà không bị gián đoạn, trong một môi trường mà bạn có thể cảm thấy thoải mái khi mình mắc sai lầm.

Trong một thế giới lý tưởng, chúng ta sẽ sử dụng kỹ thuật “thực hành có chủ ý” như được trình bày trong nghiên cứu của K. Anders Ericsson, và một cố vấn sẽ chỉ định cho bạn một bài tập dựa trên sự hiểu biết của bạn về điểm mạnh và điểm yếu của bản thân. Khi bạn hoàn thành bài tập, người cố vấn sẽ làm việc với bạn để đánh giá hiệu suất của bạn thông qua việc sử dụng một số liệu khách quan, và sau đó làm việc với bạn để đưa ra bài tập tiếp theo. Sau đó, cố vấn của bạn sẽ sử dụng kinh nghiệm giảng dạy các học viên khác để tạo ra các bài tập mới và đầy thử thách hơn, điều này sẽ khuyến khích bạn suy ngẫm về kỹ năng của mình, tìm thói quen làm việc hiệu quả hơn và phát triển khả năng “nhìn thấy” những phần kiến ​​thức. Theo thời gian, chuỗi các bài tập này sẽ trau dồi thêm điểm mạnh của bạn và sửa chữa các điểm yếu của bạn. Đáng buồn là chúng ta không sống trong một thế giới lý tưởng, và người tập sự phải dựa vào nguồn lực của mình để đạt được hiệu quả tương tự.

Trong phần mềm, chúng ta luyện tập trên công việc của mình, và đó là lý do tại sao chúng ta phạm sai lầm trong công việc. Chúng ta cần phải tìm ra những phương pháp chia tách thực hành khỏi nghề nghiệp. Chúng ta cần những buổi luyện tập.

- Dave Thomas-

Dave Thomas thực dụng đã mượn khái niệm mã kata từ võ thuật. Một kata là một chuỗi liên hoàn gồm động tác được một võ sư tạo ra để giúp học trò tiếp thu các nguyên tắc cơ bản của võ thuật. Kata được thực hiện mà không có đối thủ, nhấn mạnh sự linh động, tốc độ và sự kiểm soát. Dave Thomas đã đăng kata lên blog của mình, khuyến khích các thợ thủ công sử dụng kata để luyện tập.

Laurent Bossavit và một nhóm các nhà phát triển phần mềm ở Paris đã mang phép ẩn dụ về võ thuật một bước xa hơn và tạo ra một võ đường của người viết mã, nơi mà mọi người có thể gặp gỡ thường xuyên để thực hiện công khai mã kata. “Bác” Bob Martin, lấy cảm hứng từ võ đường Paris, cũng đăng bài kata lên blog của mình, tán thành việc thực hành trong nghề thủ công.

Người mới bắt đầu học bằng cách làm, chứ không phải qua bài giảng. Họ thực hành, thực hành, và thực hành …. Bằng cách lặp đi lặp lại những bài tập tương tự, chúng tôi sẽ nâng cao kỹ năng của mình,chúng tôi đào tạo cơ thể và tâm trí của mình có thể đáp ứng các nguyên tắc của TDD (Mô hình phát triển với trọng tâm hướng vào việc kiểm thử) và thiết kế đơn giản.Chúng tôi kết nối, rồi kết nối, kết nối đi kết nối lại các nơ-ron của chúng tôi để đưa ra những phản xạ đúng.

-Robert Martin-

Rõ ràng mã kata chỉ là một trong những cách để Thực Hành, Thực Hành, Thực Hành (Trò Chơi Lắp Ghép là việc khác). Mấu chốt của mẫu này là bỏ thời gian ra để phát triển phần mềm trong môi trường không căng thẳng và thoải mái: không có ngày phát hành, không có vấn đề về sản xuất, và không có những gián đoạn. Như Dave Thomas nói về thực hành, “Thư giãn là điều cần được chấp nhận, bởi vì nếu bạn không thư giãn bạn sẽ không học hỏi từ việc luyện tập.”

Các vòng phản hồi ngắn cần được kết hợp vào các buổi thực hành. Mặc dù thực hành về mặt lý thuyết là tốt, nhưng nếu bạn không nhận được thông tin phản hồi định kỳ thì có khả năng bạn đang phát triển những thói quen xấu. Đây là vẻ đẹp của dojo (trường dạy võ tự vệ) của các lập trình viên: thực hành công khai trong bối cảnh của một cộng đồng đoàn kết chặt chẽ của các thợ thủ công. Nhu cầu cần có thông tin phản hồi liên tục giảm dần đi khi bạn phát triển như một thợ thủ công, và dần dần được thay thế bằng nhiệm vụ của bạn để đảm nhiệm vai trò của một người học việc cao cấp, mô phỏng những thói quen tốt bằng cách thực hành với các nhà phát triển ít kinh nghiệm hơn.

Một trong những lý do mà các bậc thầy được George Leonard mô tả là thích thực hành, đó là họ luôn làm một điều khác biệt mỗi khi thực hiện một bài tập. Mục đích ở đây không phải là để trau dồi trí nhớ của bạn, mà để khám phá những sắc thái khác nhau ngay cả trong những hoạt động có tay nghề đơn giản nhất. Bà của bạn có thể đã nói với bạn rằng thực hành làm nên sự hoàn hảo. Bà đã sai rồi. Trong thực tế, thực hành làm nên sự vĩnh viễn. Vì vậy, hãy cẩn thận những gì bạn thực hành, và liên tục đánh giá nó để đảm bảo bạn vẫn chưa bị mụ mẫm. Lựa chọn điều đúng đắn để thực hành mỗi ngày là một kỹ năng quan trọng gần ngang với hành động lặp đi lặp lại việc thực hành. Một cách tốt để đảm bảo bạn có các bài tập thú vị để sử dụng trong các buổi thực hành của bạn là rà soát lại các cuốn sách cũ như *Programming Pearls*, *More Programming Pearls*, hay *Etudes for Programmers*. Chúng đã được viết đủ lâu từ trước kia, rằng họ không còn lựa chọn nào khác ngoài việc tập trung vào các nguyên tắc cơ bản của khoa học máy tính, chứ không phải vào khuôn khổ hợp thời mới nhất. Các tác giả hiểu rằng việc nhận được các nguyên tắc cơ bản từ sự phức tạp của các thuật toán và các cấu trúc dữ liệu đã được ngấm sâu vào, là ít gây hại và hiếm khi không còn hữu ích. Các chủ đề này cũng cung cấp một nguồn vô tận các vấn đề nhỏ thú vị để giữ cho các buổi thực hành của bạn thú vị, tươi mới và đầy tính giáo dục.

## **Hành động**

Tìm một bài tập trong một trong những cuốn sách đã đề cập trước đó hoặc tạo ra một bài tập của riêng bạn. Hãy chắc chắn rằng nó chỉ khó hơn một chút so với điều bạn biết bạn có thể dễ dàng giải quyết. Bạn nên gặp khó khăn khi cố gắng để giải quyết vấn đề đó lần đầu tiên. Giải quyết bài tập này từ đầu mỗi tuần một lần trong bốn tuần tới và quan sát cách các giải pháp của bạn tiến triển. Điều này cho bạn biết gì về điểm mạnh và điểm yếu của mình khi là một lập trình viên? Hãy nắm lấy kiến ​​thức đó và cố gắng tìm hoặc đưa ra một bài tập mới mà sẽ có một tác động đáng kể đến khả năng của bạn. Lặp lại điều này.

[[1]](https://codegym.vn/blog/2018/09/12/hoc-nghe-thuc-hanh-thuc-hanh-thuc-hanh/#_ftnref1) *Dave Thomas về Mã Kata:* [*http://codekata.pragprog.com/2007/01/code\_kata\_backg.html*](http://codekata.pragprog.com/2007/01/code_kata_backg.html)

[[2]](https://codegym.vn/blog/2018/09/12/hoc-nghe-thuc-hanh-thuc-hanh-thuc-hanh/#_ftnref2) *Robert Martin với Lập trình Dojo:* [*http://butunclebob.com/ArticleS.UncleBob.TheProgrammingDojo*](http://butunclebob.com/ArticleS.UncleBob.TheProgrammingDojo)