Most of the implementation guides so far in Learning on Steroids have focused on conceptual skills. Things like being able to form mental images, remembering facts and formulas, understanding big ideas.

But conceptual learning is only one part (and in some cases, a small part) of learning. The other half is mastering skills.

Skill is the Missing Half

Take a class on computer programming. Sure, there are probably a lot of big concepts to learn: object orientation, binary search trees or mapping. There are also a lot of factual details, such as Python defining blocks by indentation and not braces, or that Java is strongly typed.

But you could know all of these details and still not be able to quickly spit out an efficient computer program. Conceptual knowledge gives you the base to work from, but only practicing writing programs and building the skill can make you an excellent coder.

Some subjects are highly skill-based. The fact that you can take an entire MBA and still be lousy at running a company, is evidence of that. Skills are greatly helped by conceptual knowledge, but ideas can't replace execution.

Other subjects in school are more conceptual and less skill-based. Psychology or history classes might be difficult to "practice" on. But even these areas have skill components: from developing research methods to learning to write convincing theses.

Rapid Mastery?

There's no substitute for practice. There's no way you can become a concert-level violin player in two weeks, for example. However, like conceptual learning, you can accelerate your ability to learn skills (or perhaps, more accurately, make sure you aren't wasting your time when practicing).

Deliberate practice is a powerful technique for ensuring you get the most out of your time spent practicing.

What is Deliberate Practice?

Deliberate practice is based on the research of K. Anders Ericsson. There are several core tenets to deliberate practice, but for this guide, I'll focus on two:

1. Training on the edge of your current abilities
2. Getting specific, timely feedback.

Not too Easy, Not too Hard

The first point is fairly obvious. If the difficulty level of your practice is too low, you won't improve much. If you've been programming for awhile, writing a simple "hello world" program won't teach you anything, unless you somehow get creative with the implementation.

However, if the difficulty level is too high, there isn't much chance of success. Unless you are able to learn (at least in part) what actually works, then the practice is meaningless. If, two days ago, you just wrote your first "hello word" program, trying to write the data structure for a 2-3 tree is probably a bit out of your league.

Writing that second program won't teach you much if you can't even get it to compile.

The first principle of deliberate practice I'll mention today is: practice should be structured to be on the edge of your current abilities. Too easy, no learning. Too hard, no chance for feedback and improvement.

Training in the Dark

The second point is sensible, but also an element many people forget when practicing a new skill. Feedback is incredibly important for the obvious reason that it allows you to adjust. If nobody ever tells you that you've made a mistake, how can you possibly correct it?

Without feedback, you're doing the equivalent of training in the dark. You may be making improvements, you may not be, you really can't see the difference.

Continuing with the programming example, this would be like writing all your computer programs on paper. It may be a good starting point for understanding concepts, but there is no feedback. No compilation, no execution and no errors. It's easier, but precisely because nobody is telling you when you slip up.

Training through Time-Delayed Glasses

I doubt any of the programmers here would consider writing all their programs on paper (unless through mathematical proofs or other means, they were able to correct them). Most people work through some kind of feedback system.

But an equally important, and often missed, point is that the feedback needs to be relatively timely in order to work. Doing a final exam and getting a grade back a month later doesn't tell you much.

Delayed practice is like training through a time-delay video. You can see what you're doing, except only what you did several days ago. In terms of being able to figure out what works and what doesn't, that isn't fast enough.

Take writing for example. If you want to improve your essay writing ability, then waiting for a professor to review a paper, make corrections and hand it back will help more than if you didn't get any feedback, but it isn't in the same window of practice.

A better approach would be to have someone read your drafts as you write them, so they can point out mistakes or give their general opinion while you're in the process of writing. Having a blog, similarly, has the power to improve your writing rapidly, since you get immediate feedback on what people liked and what they didn't like.

Implementing Deliberate Practice

Deliberate practice is not a complex tactic to understand. Nothing I've said so far has probably surprised you. The real difficulty is aligning your current practice so that it better embodies the principles of the right difficulty with the right feedback.

Aligning the Difficulty Level

When practicing a skill, there are two situations which you might find yourself in:

1. The skill is currently too easy.
2. The skill is currently too hard.

With the first, you want to take steps to deliberately increase the difficulty. This is not an obvious move, as many people fail to continue mastering a skill because they get stuck at a comfort zone–their skills merely sitting in the moderate challenge range.

For the second, you need to find a way to decompose the difficult challenge into simpler steps. Again, this is not always obvious. Many people either shut down in the face of a bigger challenge, and give up altogether, or they run headlong into it and fail to learn the intermediate lessons often necessary for mastery.

How to Upgrade the Challenge Level

There are two ways I've found useful to upgrade the challenge level: move to a different practice area or strip away the easy parts.

Switching Practice Areas

Take a skill like learning a foreign language. One way to increase the challenge level is simply to switch where you're training. If you're pretty good on the beginner audio tapes, get recordings from actual programs. If you can read the magazine, try a book. If you can understand the one-on-one conversation, try talking in a group.

The "move" to a different practice area is sometimes metaphorical. If you're working on your math skills, you might move to more difficult proofs and problem sets. So if the basic algorithms are okay, try shifting to the more difficult formulas.

Stripping Away the Easy Parts

Let's look at programming again. In any reasonable application, you probably have a lot of really easy stuff: variable assignments, function definitions, or setting up the databases. Maybe for you, the edge of your abilities is on more complex algorithms, data structures or mathematical equations.

However, if you approach practice blindly, you'll probably spend 90% of your time on the basics with only 10% of your time on the really difficult stuff that pushes you. So for every hour you spend practicing, you're only really improving for six minutes of that time.

The solution here is to craft projects, problems or practice sessions which deliberately eliminate the easy stuff. Structure

your programming problem in such a way that it deliberately forces you to learn the new data structure or algorithm.

For a language, this could mean learning to write poetry or reading philosophy. Something that may not be your natural form of practice, but makes the challenge level more dense.

Knowing When to Increase the Challenge

My feeling is that our natural tendency is to make practice too easy, not too hard. We like best challenges that are moderately hard, where our success rates are 80-90%. However, the best way to improve is probably closer to when success rates are lower and you are a bit stressed with the challenge.

My rule-of-thumb is that if you aren't sure whether your current practice is too easy, it is too easy, and increase the difficulty. If you hit an impasse and have no idea what to do, you've raised the challenge level too high and can take a step back.

How to Downgrade the Challenge Level

What happens if you do hit that impasse, and can't go any further. Here your problem is the opposite, the challenge level is too hard and you're spinning your wheels instead of getting the feedback you need to improve.

The first way to downgrade the challenge level is to just switch the area of training. So if the rapid fire foreign-language talk show is overwhelming you, why not try the radio program with a single announcer. Same idea as before, just in the opposite direction.

Another way is to cut away elements of the difficult problem and focus on one at a time. So if a difficult word problem forces you to do gradients and matrix multiplication, split the problems up and focus just on the gradients first.

Getting Timely Feedback

Along with right difficulty, timely feedback is crucial. However, your default practice zone may not give great feedback and it may not give you back information fast enough.

One way to fix this is to simply change the environment of your practice to one that offers feedback more quickly. Flashcards for arbitrary information work well often because they let you know immediately whether you got the answer right or wrong.

For subjective skills, like essay writing or musical performance, you can tap into the power of the internet. There are few other ways to get dozens of impressions based on your work than by showcasing it online for free. Getting academic coaches and professors to look at your work can be another way, but then you need to work on their schedule or sometimes pay hefty fees.

Implementing Deliberate Practice

Deliberate practice isn't so much a skill you can master with a 30-Day Trial as it is a way of mastering skills. Your job with implementation shouldn't be to become an expert in deliberate practice, but to align one particular skill-based component of your learning with the right difficulty and timely feedback for 30 days.

The first step is to decide which skill you want to create a deliberate practice schedule for.

The next step is to design a practice schedule at the right difficulty level and with timely feedback.

Just as an example, I recently decided to do this with my blog writing. I picked one dimension of difficulty (in this case, writing longer, more researched articles) and then created a new writing schedule around it. I changed my workflow to allow me more time to work on the research and drafting of the articles I'm trying to stretch myself with.

If you were trying to improve your ability with a subset of mathematical problems, you may want to set up a certain amount of time each day to read some of the techniques and then work through the proofs or problem applications.

Getting Started with Deliberate Practice

If your current environment doesn't foster immediate feedback, find a way to work that in. You should be able to know rapidly whether your efforts in this new domain are successful or not. Some environments already foster immediate feedback, such as my blog writing, others require tweaks, such as getting solutions to problem sets you want to work on.

Pick a skill, set up the deliberate practice schedule and then work on it for thirty days.

Good luck with this tactic, and I'll see you on the other side.