TEACH YOURSELF DESIGNING IN TEN YEARS

Note: This blogpost is inspired from Prof. Peter Norvig’s famous post “Teach Yourself Programming in Ten Years”. Credit for all the convincing arguments go to him. All errors remaining are mine.

Why is everyone in such a rush?

Walk into any bookstore, and you’ll see how to Teach Yourself Photoshop in 21 days alongside endless variations offering to teach Illustrator, Corel Draw, Dreamweaver, and so on in a few days or hours. I did the following power search at Amazon.com:

```plaintext
pubdate: after 1992 and title: days and (title: learn or title: teach yourself)
```

and got back 248 hits. The first 78 were computer books (number 79 was Learn Bengali in 30 days). I replaced "days" with "hours" and got remarkably similar results: 253 more books, with 77 computer books followed by Teach Yourself Grammar and Style in 24 Hours at number 78. Out of the top 200 total, 96% were computer books.

The conclusion is that either people are in a big rush to learn about, design or design tools or that designing is somehow fabulously easier to learn than anything else. There are no books on how to learn Beethoven, or Quantum Physics, or even Dog Grooming in a few days. Paraphrasing Felleisen et al. (from their book How to Design Programs): "Bad designing is easy. Idiots can learn it in 21 days, even if they are dummies."

Let’s analyze what a title like Sams Teach Yourself Adobe® Photoshop® 5 in 21 Days could mean:

- **Teach Yourself**: In 21 days you won’t have time to create several significant designs (e.g. mockups, wireframes, icons, fonts, digital graphics etc.) and learn from your successes and failures with them. You won’t have time to work with an experienced designer and understand what it is like to work with other designers. In short, you won’t have time to learn much. So the book can only be talking about a superficial familiarity, not a deep understanding. As Alexander Pope said, a little learning is a dangerous thing.

- **Adobe® Photoshop® 5**: In 21 days you might be able to learn some of the tools and options in Photoshop (if you already know how to use another graphics editor), but you won’t learn much about how to use the Adobe Photoshop extensively. In short, if you were, say, a GIMP designer, you could learn to create design in the style of GIMP using Photoshop tools and techniques, but you couldn’t learn what Photoshop is actually good (and bad) for. So what’s the point? Paraphrasing Alan Perlis “A graphics program that doesn’t affect the way you think about designing, is not worth knowing”. One possible point is that you have to learn a tiny bit of Photoshop (or more likely, something like Balsamiq or Just In Mind) because you need to supplement an existing tool to accomplish a specific task. But then you’re not learning how to design with this tool; you’re learning to accomplish that task using this tool.

- **in 21 Days**: Unfortunately, this is not enough, as the next section shows.

Teach Yourself Designing in Ten Years

Researchers (Bloom (1985), Bryan & Harter (1899), Hayes (1989), Simmon & Chase (1973)) have shown it takes about ten years to develop expertise in any of a wide variety of areas, including chess playing, music composition, telegraph operation, painting, piano playing, swimming, tennis, and research in neuropsychology and topology. The key is deliberative practice: not just doing it again and again, but challenging yourself with a task that is just beyond your current ability, trying it, analyzing your performance while and after doing it, and correcting any mistakes. Then repeat. And repeat again. There appear to be no real shortcuts: even Mozart, who was a musical prodigy at age 4, took 13 more years before he began to produce world-class music. In another genre, the Beatles seemed to burst
onto the scene with a string of #1 hits and an appearance on the Ed Sullivan show in 1964. But they had been playing small clubs in Liverpool and Hamburg since 1957, and while they had mass appeal early on, their first great critical success, Sgt. Peppers, was released in 1967. Malcolm Gladwell reports that a study of students at the Berlin Academy of Music compared the top, middle, and bottom third of the class and asked them how much they had practiced:

Everyone, from all three groups, started playing at roughly the same time - around the age of five. In those first few years, everyone practised roughly the same amount - about two or three hours a week. But around the age of eight real differences started to emerge. The students who would end up as the best in their class began to practise more than everyone else: six hours a week by age nine, eight by age 12, 16 a week by age 14, and up and up, until by the age of 20 they were practising well over 30 hours a week. By the age of 20, the elite performers had all totalled 10,000 hours of practice over the course of their lives. The merely good students had totalled, by contrast, 8,000 hours, and the future music teachers just over 4,000 hours.

So it may be that 10,000 hours, not 10 years, is the magic number. (Henri Cartier-Bresson (1908-2004) said “Your first 10,000 photographs are your worst, “but he shot more than one an hour.”) Samuel Johnson (1709-1784) thought it took even longer: “Excellence in any department can be attained only by the labor of a lifetime; it is not to be purchased at a lesser price.” And Chaucer (1340-1400) complained “the lyf so short, the craft so long to lerne.” Hippocrates (c. 400BC) is known for the excerpt “ars longa, vita brevis”, which is part of the longer quotation “Ars longa, vita brevis, occasio praeceps, experimentum periculosum, iudicium difficile”, which in English renders as “Life is short, [the] craft long, opportunity fleeting, experiment treacherous, judgment difficult.” Although in Latin, ars can mean either art or craft, in the original Greek the word “techne” can only mean “skill”, not “art”.

So You Want to be a Designer

Here’s my recipe for designing success:

- **Get interested** in programming, and do some because it is fun. Make sure that it keeps being enough fun so that you will be willing to put in your ten years/10,000 hours.

- **Design**. The best kind of learning is learning by doing. To put it more technically, “the maximal level of performance for individuals in a given domain is not attained automatically as a function of extended experience, but the level of performance can be increased even by highly experienced individuals as a result of deliberate efforts to improve.” (p. 366) and “the most effective learning requires a well-defined task with an appropriate difficulty level for the particular individual, informative feedback, and opportunities for repetition and corrections of errors.” (p. 20-21) The book Cognition in Practice: Mind, Mathematics, and Culture in Everyday Life is an interesting reference for this viewpoint.

- **Talk with** other designers; review other’s designs (on sites such as Dribble, Behance etc.) This is more important than any book or training course.

- If you want, put in four years at a college (or more at a Design school). This will give you access to some jobs that require credentials, and it will give you a deeper understanding of the field, but if you don’t enjoy school, you can (with some dedication) get similar experience on your own or on the job. In any case, book learning alone won’t be enough. Paraphrasing Eric Raymond (author of The New Hacker’s Dictionary): “Design education cannot make anybody an expert designer any more than studying brushes and pigment can make somebody an expert painter”. One of the best design perfectionist dropped out of college, he was behind loads of great designs, had his own company and took his company to heights no one could ever imagine.

- **Work on projects** with other designers. Be the best designer on some projects; be the worst on some others. When you’re the best, you get to test your abilities to lead a project, and to inspire others with your vision. When you’re the worst, you learn what the masters do, and you learn what they don’t like to do (because they make you do it for them).

- **Work on projects after** other designers. Understand and analyse design created by someone else. See what it takes to understand and fix it when the original designers are not around. Think about how to create designs to make it easier for those who will maintain them after you.

- Learn at least a half dozen design tools and techniques. Include one tool that allows editing of raster images (like Photoshop or GIMP), one that allows editing of vector images (like Illustrator or Inkscape), one that allows creation of wireframes (like Balsamiq), one that allows creations of functional mockups (like JustInMind or HTML/CSS) and one that allows creation of low fidelity mockups for rapid iterations (like hand drawn wireframes).

With all that in mind, it’s questionable how far you can get just by book learning. Quoting Peter Norvig:
"Before my first child was born, I read all the How To books, and still felt like a clueless novice. 30 Months later, when my second child was due, did I go back to the books for a refresher? No. Instead, I relied on my personal experience, which turned out to be far more useful and reassuring to me than the thousands of pages written by experts."

Paraphrasing Fred Brooks (from his essay *No Silver Bullet*), a three-part plan for finding great designers:

- Systematically identify top designers as early as possible.
- Assign a career mentor to be responsible for the development of the prospect and carefully keep a career file.
- Provide opportunities for growing designers to interact and stimulate each other.

This assumes that some people already have the qualities necessary for being a great designer; the job is to properly coax them along. Alan Perlis put it more succinctly: "Everyone can be taught to sculpt: Michelangelo would have had to be taught how not to. So it is with the great designers". Perlis is saying that the greats have some internal quality that transcends their training. But where does the quality come from? Is it innate? Or do they develop it through diligence? As Auguste Gusteau (the fictional chef in Ratatouille) puts it, "anyone can cook, but only the fearless can be great." I think of it more as willingness to devote a large portion of one's life to deliberative practice. But maybe fearless is a way to summarize that. Or, as Gusteau's critic, Anton Ego, says: "Not everyone can become a great artist, but a great artist can come from anywhere."

So go ahead and buy that designing/photoshop/graphics book; you'll probably get some use out of it.

But you won't change your life, or your real overall expertise as a designer in 24 hours, days, or even weeks. How about working hard to continually improve over 24 months? Well, now you're starting to get somewhere…

References

Norvig, Peter, *Teach Yourself Programming in Ten Years*

Appendix: Choice of Tools

Several people have asked which graphics/HCI designer tools they should learn first. There is no one answer, but consider these points:

- **Use your friends.** When asked "what vector graphics editor should I use, Illustrator, CorelDraw, or Inkscape?", my answer is usually: "use whatever your friends use." The advantage you get from learning from your friends will offset any intrinsic difference between vector graphics editors. Also consider your future friends: the community of designers that you will be a part of if you continue. Does your chosen design tool have a large growing community or a small dying one? Are there books, web sites, and online forums to get answers from? Do you like the people in those forums?
- **Keep it simple.** Tools such as Photoshop and Illustrator are designed for professional graphics design and editing by large teams of experienced designers who are concerned about the rapidly creating graphics. As a result, these programs have complicated features designed for these circumstances. You're concerned with learning to design. You probably don't need that complication. You want a tool that was designed to be easy to learn and remember by a single new designer.
- **Play.** Which way would you rather learn to play the piano: the normal, interactive way, in which you hear each note as soon as you hit a key, or "batch" mode, in which you only hear the notes after you finish a whole song? Clearly, interactive mode makes learning easier for the piano, and also for designing. Insist on a tool with an interactive mode and use it.

Given these criteria, my recommendations for a first HCI design would be Balsamiq or hand drawn wireframes. But your circumstances may vary, and there are other good choices. If your age is a single-digit, you might prefer MS-Paint or it's variants for simple graphics editing (older learners might also enjoy these). The important thing is that you choose and get started.

Appendix: Books and Other Resources

Several people have asked what books and web pages they should learn from. I repeat that “book learning alone won’t be enough” but I
can recommend the following:

- **HCI/UI/UX design courses:** Human Computer Interaction (Coursera), Intro to Design of Everyday Things (Udacity), Design: Creation of Artifacts in Society (Coursera), Human Computer Interaction (by Alan Dix).
- **HCI/UI/UX books:** Don’t Make Me Think by Steve Krug, Design of Everyday Things by Don Norman, Human Computer Interaction by Alan Dix.

**Notes**

T. Capey points out that the *Complete Problem Solver* page on Amazon now has the “Teach Yourself Bengali in 21 days” and “Teach Yourself Grammar and Style” books under the “Customers who shopped for this item also shopped for these items” section. I guess that a large portion of the people who look at that book are coming from this page. Thanks to Ross Cohen for help with Hippocrates.

Peter Norvig’s famous post “Teach Yourself Programming in Ten Years”. Credit for all the convincing arguments go to... Teach Yourself: In 21 days you won’t have time to create several significant designs (e.g. mockups, wireframes, icons, fonts, digital graphics etc.) and learn from your successes and failures with them. You won’t have time to work with an experienced designer and understand what it is like to work with other designers.