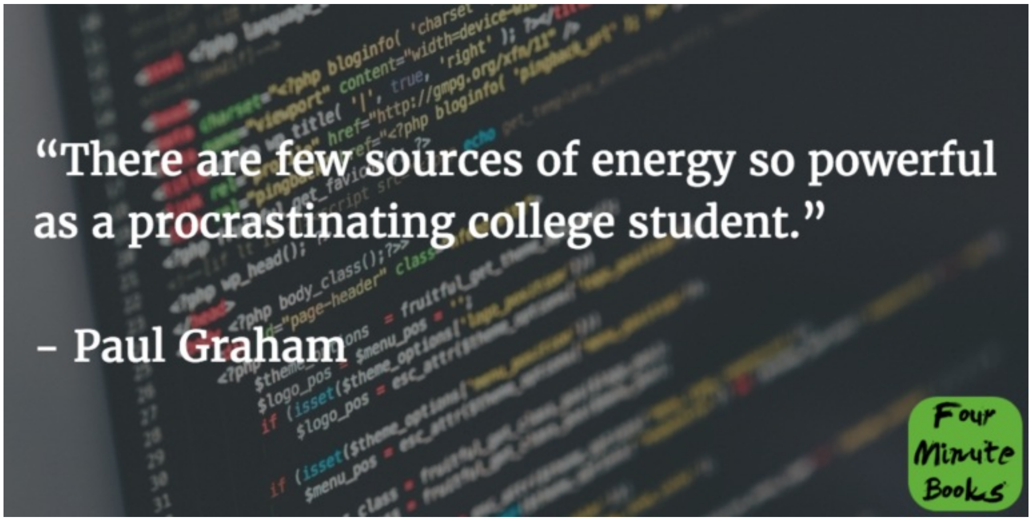
Hackers And Painters Summary

fourminutebooks.com/hackers-and-painters-summary

1-Sentence-Summary: *Hackers And Painters is a collection of essays by Y Combinator founder Paul Graham about what makes a good computer programmer and how you can code the future if you are one, making a fortune in the process.*

Read in: 4 minutes

Favorite quote from the author:



"There are few sources of energy so powerful as a procrastinating college student."

– Paul Graham

Paul Graham was one of the early adopters of the internet, having launched a company called Viaweb in 1995, which was the first business to offer an online application as a service – they allowed their users to set up their own, very simple online stores. Three years later, the company sold to Yahoo! for about $50 million.

Today Paul is mostly known for founding Y Combinator, one of the most successful venture capital companies in the world, with investments in over 400 startups to date, some of them household names like Dropbox, AirBnB and Stripe. Paul has also started writing essays about programming, startups, entrepreneurship and other topics, some of which were bundled up in this book and released in 2004.

His story is *the* prototype of how tech entrepreneurs can become a success and he's making great use of it by spreading what he's learned and inspiring others to do the same.

Here are 3 lessons from *Hackers And Painters*:

1. Both morals and fashion trends are temporary, which is why nerds don't care about either of them.
2. Hackers are more like painters than mathematicians.
3. User feedback is the ultimate test of your programming skills, so get it as fast as possible.

Ready to take your hacker mentality to the next level? Here we code!

Lesson 1: Nerds are neither interested in fashion, nor morals, because both are seasonal.

I never thought to connect these two, so this is interesting.

If you look at pictures of yourself from the 90s (if those exist) you'll likely go "Oh my god, what the hell was I thinking when wearing this?!" That's because fashion trends change fast and every decade has its own unique, characteristic style. Plus, there are huge differences in cultural fashion, for example Japanese fashion differs vastly from trends in the US or European clothing.

What else is highly location-dependent, seasonal and fluctuates a lot? Morals. A faithful husband or wife might be tempted to cheat during a trip to Vegas, church services always see a surge in faithful attendants around Christmas and the short skirt that was irresponsible and amoral to wear 15 years ago has become a household item at Forever 21.

Because both morals and fashion swing from one extreme to the next as much as the weather, nerds don't care about either of them.

This is because nerds, according to Graham, are smart people, who don't bother with conforming to social conventions. They know fashion comes and goes, so they don't even waste their energy trying. As for morals, they do sure have their own set of them, but they don't change them every quarter to do what the prom king and queen think is right. Instead, they hold on to their values.

Nerds therefore have an advantage in the real world, because after school and college, neither fashion nor morals will get you very far.

Lesson 2: Hackers are much more similar to painters, rather than mathematicians.

Most people imagine hackers as very calculated people, meticulous planners, who are very analytical. At least for *good* hackers, the opposite is true.

The very definition of a hacker gets this wrong quite often. For example, here's what Google says:

> A person who uses computers to gain unauthorized access to data.

Now look at the definition in the Urban Dictionary, a dictionary for colloquial usage of words, which is a collaboration of millions of users:

> A person skilled with the use of computers that uses his talents to gain knowledge.

You see, hacking isn't about doing illegal things at all. It can be. But it doesn't have to. In order to gain knowledge through programming and get even smarter, coders have to try and create their own concepts, not just implement what others tell them to – just like any great painter comes up with his own artwork and doesn't just try to re-create famous paintings, like the Mona Lisa.

For example, in college Paul Graham was taught to write his code on paper, perfect it, and only then transfer it to a computer. But Paul found taking a painter's approach worked a lot better: if he started coding on the machine and then fixed problems as they happened, the result was much better.

Similarly, a hacker's work can only be valued subjectively, because different people will have different needs, and how well they like a program depends on how good it is at taking care of those needs.

Hackers are artists, just like painters.

Lesson 3: The ultimate test of your coding skills is user feedback, so try to get it fast!

Worse is better. Jane Austen knew that already. The author of Pride and Prejudice would get "user feedback" from family members by reading her novels aloud to them before finalizing them, and asking them what they thought of the characters in her books.

Since the value of your work as a programmer is determined by what people think of your projects, **the fastest way to get better is to get your work in front of others**.

So don't waste a lot of time fleshing out nice-to-have features and cool gimmicks. Just build a raw, stripped-down prototype with the core functionality and ship it. Every piece of feedback will help you make the next version better.

A great chair has to be primarily one thing: comfortable. You could build a chair that's super ugly, but incredibly comfortable, and it'd sell instantly. Making it pretty afterwards is easy, as long as it does what a chair is supposed to. Software is exactly the same.

Hackers And Painters Review

I imagine reading *Hackers And Painters* when it came out 12 years ago must've felt like a revelation. Of course programmer has become a much more conventional career by now, but I think learning to adopt the mentality of what makes a *good* programmer is just as valuable today. It's timeless. And if you're a non-coder, this will help you understand programmers and hackers a lot better!

Read full summary on Blinkist >>

Free Preview >>

Learn more about the author >>

What else can you learn from the blinks?

- Why the FBI was stunned when they first identified hackers' true motives
- How there's nothing wrong with funding a startup to get rich
- Why there are so many different programming languages
- What makes a good programming language
- The advantage businesses can get from using uncommon coding languages
- What you can learn about good hacking from coding your own spam filter

Who would I recommend the Hackers And Painters summary to?

The 15 year old computer nerd who's bullied in school for wearing plain clothes and being socially awkward, the 33 year old who leads a team of programmers at his company, and anyone who thinks hacking is primarily about crime.