

CHRISTIAN MAYER • LUKAS RIEGER



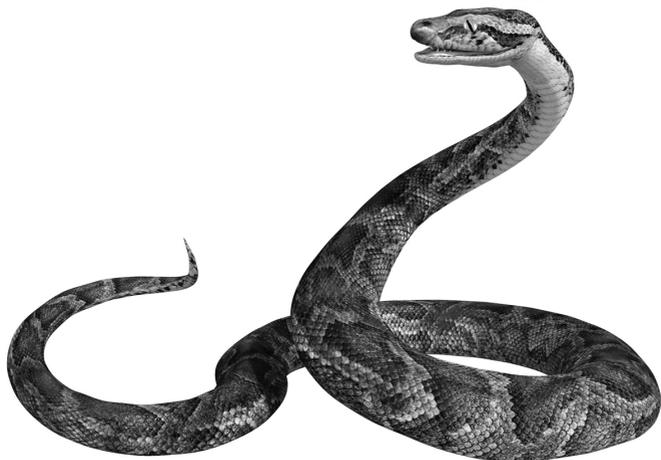
LEAVING THE RAT RACE



WITH



PYTHON



— AN INSIDER'S GUIDE TO —
FREELANCE DEVELOPING

finxter

Leaving the Rat Race with Python

An Insider's Guide to Freelance
Developing

Dr. Christian Mayer and Lukas Rieger

Dedication

This book is dedicated to the vivid Finxter community of ambitious Python coders.

© 2021 *Finxter Publishing*
ALL RIGHTS RESERVED.
<https://finxter.com/>

For more information about permission to reproduce
selections from this book, write to
admin@finxter.com.

Subscribe to our newsletter *Grow Your Python Skills
and Become a Professional Freelance Developer*
<https://blog.finxter.com/subscribe/>.

About This Book

This book originated from the massive results obtained by Finxter students and their success in becoming thriving Python business owners. Tens of thousands of avid Python students have completed the Python freelancing material on the Finxter website, such as the free webinar¹, the blog tutorial², or the full Python freelancer course³.

Many Finxters wrote in with their successes applying the principles outlined in this book.⁴

The Python freelancer book you hold in your hands (or

¹<https://blog.finxter.com/webinar-freelancer/>

²<http://bit.ly/python-freelancer-tutorial>

³<https://blog.finxter.com/become-python-freelancer-course/>

⁴<https://blog.finxter.com/what-our-users-say/>.

keep at your fingertips) is the compilation of our collective practical knowledge—working as freelancers, hiring freelancers, and teaching students how to become successful freelancers. It’s written for the complete novice in the field of Python freelancing.

However, it’s not a Python tutorial. To learn Python, get the *Coffee Break Python* book series⁵ and join the free Finxter Computer Science Email Academy⁶.

It’s sad, but for many reasons—some discussed in this book—the average reader will not reach success with their business. That’s why we add the following disclaimer: *Any earning figures stated in the book are not representative of the average reader who will earn significantly less or no money at all. Dedication and endurance are key to building a successful business.*

Finally, we’ve created a separate book page with bonus material, worksheets, and additional freelancer resources at <https://python-freelancer.com/>.

⁵<https://blog.finxter.com/coffee-break-python/>

⁶<https://blog.finxter.com/subscribe/>

Table of Contents

I Overview: \$0 to \$1000	38
II Skills: \$1000 to \$2000	68
III Strategy: \$2000 to \$4000	118
IV Systems: \$4000 to \$12000	151

Long Table of Contents

Table of Contents	vi
1 Introduction	1
1.1 Freelancing in the 21st Century	3
1.2 About This Book	6
2 Freelancer Report	7
2.1 Disruption	9
2.2 Pro vs Cons	13
2.3 What's Unique	23
2.4 Hourly Rate	25
2.5 Code From Home	27
2.6 What Does It Take	29
2.7 Intermediate Freelancer	33
2.8 Old Freelancers	34
2.9 Young Freelancers	35

I Overview: \$0 to \$1000	38
3 Low Risk High Gain	45
4 Unfair Advantage	51
5 Know Your Why	55
6 Consultants Go Big	58
7 Join Freelance Platforms	60
8 Best Freelancing Sites	64
8.1 Upwork	65
8.2 Fiverr	65
8.3 Toptal	66
8.4 Freelancer.com	66
II Skills: \$1000 to \$2000	68
9 Focus on Skill Creation	69
10 Your Success Habit	72
11 Earn Trust	75
12 Money Seeks Specialists	78
13 Read Programming Books	81

<i>LONG TABLE OF CONTENTS</i>	ix
14 Read Business Books	83
15 Seek Expert Advice	85
16 A Simple Heuristic	87
17 About Timing	93
18 Practical Training Plan	95
19 Solve Python Puzzles	98
20 Learn with Cheats	101
21 Confidence	103
22 Start Earning	106
23 Psychology Tricks	108
24 Open-Source	113
25 Non-Programming Skills	115
III Strategy: \$2000 to \$4000	118
26 Freelancer Algorithm	119
27 Network Effects	123

28 Return on Invested Effort	125
29 Don't Compete on Price	128
30 Free Float	131
31 Give	134
32 Eat Complexity	136
33 Perform From Your Strengths	139
34 Be a Specialist	141
35 Be Hyper-Responsive	143
36 Be Positive	145
37 Know Hourly Rate	147
38 Increase Hourly Rate	149
IV Systems: \$4000 to \$12000	151
39 Systems	152
40 Client List	156
41 Ad Funnel	158

LONG TABLE OF CONTENTS

xi

42 Leads	160
43 Testimonial Videos	162
44 Referral Engine	166
45 Give and Take	169
46 Befriend Colleagues	171
47 On Investing	173
48 Leave Freelancing Platforms	177
49 Maximize Control	179
50 Seek Independence	181
51 Consultant Positioning	185
52 Strategy Sessions	188
53 Tutoring	191
54 Personal Brand	194
55 Create a Blog	196
56 Final Remarks	199

— 1 —

Introduction

Do you want to learn about becoming a Python freelancer and start your thriving coding business online? Do you want to regain control of your life? And are you committed to succeeding in life and business?

You've got the right book in your hands. This practical how-to book will help you nurture, grow, and harness your new online coding business plant—even if you've got little or no experience in both the coding and business ecosystems. If you follow the instructions in this book and commit to building your successful freelance business, you'll not only make this book the most profitable investment in your life, but you'll also achieve new joy, happiness, and a sense of independence and self-reliance.

Giving advice is cheap—but taking advice can become

very expensive. Thus, you should choose your advice-givers carefully. Allow me to write a few words about myself so that you can decide if I'm the right teacher for you!

For more than four years, I've been growing my online coding business Finxter, and it still feels like Alice's—or, to make the metaphor fit, Bob's—Wonderland. I'm working from the comfort of my own home—I write, code, and help my students. I'm getting paid to do just the thing I love most: helping people become successful coders. The Finxter online business earns excess cash for myself, my family, and my business partners and coworkers, and I no longer have to worry about financial matters. You might expect that I'd have to pour every free minute into it. But the truth is that I spend only four hours focused on work every day—instead of eight hours in the office, one hour at lunch, and two hours in commuting. Yes, I spend a lot of time *thinking* about how I can help my subscribers become successful, but I don't consider this work—it's my passion. Still, I've got plenty of time to see my kids grow up, enjoy the company of my wife and friends, and take long walks. While building and growing my business, I also learn about state-of-the-art technologies and nurture my business skills daily. After years of high-pressure and stressful work as a doctoral research associate, I am grateful for this new, refreshing, and exciting life I found myself

in.

This book is about giving back and helping others obtain the same level of happiness. In essence, this book is about distilling the most valuable insights from my experiences and delivering them to you so that you can avoid wasting time and money making the same mistakes I made.

Learn from the mistakes of others. You can't live long enough to make them all yourself.—Eleanor Roosevelt

1.1 Freelancing in the 21st Century

The landscape of work has already undergone massive changes in the 21st century.

Many conflicts in the 20th century concerned the relationship between employers and employees. The former played the role of the business owner, who only cared about maximizing cash flow and business profits. The latter worked for the former—resenting the fact that their role was to generate more money than they were getting paid by the business owner.

This binary world view caused a lot of tension between the two groups, resulting in protests and strikes—millions of employees organized in unions to gain more power and

leverage against business owners. At the same time, the business owners fought back rigorously, causing suffering and pain among millions of employees.

But the world has changed. Rather than being divided into two classes with opposing world-views and incentives, the lines have blurred. Today, more and more people work independently as small business owners and as self-employed freelancers. Automation reduces job safety because machines replace human labor and intelligence. But the long-term positive effects of automation are that, increasingly, people are not anymore forced to work jobs they hate. Let me show you a few facts:

- **Fact 1:** The Bureau of Labor Statistics estimates that 10,000,000 workers in the US are self-employed—and the number grows faster than the number of employed people¹. Most of these businesses have less than ten employees, and include jobs such as Uber drivers, Airbnb renters, freelancers, and consultants.
- **Fact 2:** The trend towards massive automation is not coming to a halt anytime soon. Today, millions of people program machines to take over more and more work from traditional jobs. In

¹<https://www.bls.gov/careeroutlook/2018/article/self-employment.htm>

2015, Tesla put its first self-driving cars on the road. Baidu and Google entered the race, and Big Car followed the innovators. Soon, the logistics sector will see a comprehensive transformation towards automation. Millions of jobs are at risk. This massive transformation will reduce job stability.

- **Fact 3:** As best-selling author Yuval Harari points out in *Homo Deus*, humanity has reached a point where far more people die from overeating than hunger. The existential need to be forced to work and stay in a job to feed your family has vanished. Even if you don't want to work, it's unlikely that you or your kids die from hunger. So why bother working in a job you hate?

These facts indicate that the strategies for a successful job-based career in the 20th century may not work in the 21st century. When self-employment is on the rise, there's no such thing as a safe job anymore, and you are not forced to work in a job you hate.

This book will help you navigate today's complicated work environment towards finding your freedom and success in life—and creating more value for more people. If you position yourself on the right side of change, you can benefit rather than suffer from those trends.

1.2 About This Book

The structure of this book is simple: after providing a high-level overview of the freelancing industry, I'll give you a series of practical tips that can help you improve your own freelancing business. Not all tips may apply at all times—but by skimming through this book regularly, you'll always find good ideas for improving your business and making more money in less time. Don't read this book cover to cover but use the first idea that makes sense, put down the book—and implement it in practice right away. *Action! Action! Action!*

I've added multiple videos about topics discussed in this book. These are optional. They are not required for you to understand the book's full content. However, for a more multi-modal learning experience, feel free to follow the provided video resources. All videos are embedded on the freely accessible book companion page (no sign-up needed!) at <https://python-freelancer.com/>. Here's the first video:

Video 1: *Python Freelancing / How to Exploit This Disruptive Mega Trend (as a Coder)*



<https://python-freelancer.com>

Code From Home—Freelance Developer Industry Report

What keeps you going day after day? Your motivation is the most critical building block of your success. In this chapter, I'd like to give you some fact-based motivation on why creating your coding business online can easily be the most rewarding decision of your life.

Yet, motivation is not everything. If you want to make your business work, you must show persistence. You need to keep working on your business for many months, even years. There's no quick and easy way to create a successful, thriving, and lasting business. It takes time, discipline, and focused effort.

The truth is that creating a successful business is a straightforward endeavor if you have the right mind-

set, habits, and motivation. Using the words of legendary entrepreneur, author, and motivational speaker Jim Rohn: *What is easy to do, is also easy not to do.*

Video 2: *Freelance Developer – How to Code From Home and Earn Six Figures [Industry Report]*



<https://python-freelancer.com>

This part of the book is intended to give you all the motivation you need to sustain for many months, even years, working daily on your new online coding business. What are the advantages and disadvantages of freelancing compared to working at a job? What can you expect to happen after you decide not to follow the herd by working for a big corporation or the government? What can you expect to earn as a freelance developer? You'll find answers to those questions, and more, in the upcoming paragraphs.

Let's take a high-level perspective analyzing some major trends in society.

2.1 The Workforce Disruption of the 21st Century

Massive change is the only constant in today's world. The nature of employment in a globalized economy is one great example. Millions of self-employed business people believe that freelancing is the most suitable way of organizing, managing, and delivering one's talent to small businesses and creators in the 21st century. Freelancing platforms such as Upwork and Fiverr grow double digits year after year.

Say you're a small business owner, and you need to get some editing done for an ebook project. Would you hire a new employee for this project? Or would you just visit an online freelancing platform and hire the best editor you can get for a fair price? You may find the answer obvious, but I don't think that most people have already realized the second-order consequences. Online freelancing is not a cute idea with little practical impact, but one that has the potential to ultimately transform and dominate the orchestration and organization of the world's talent. It's accessible to billions of creators and business owners, and will become even more efficient in the future.

When I discuss the evolution of the traditional *employment-centered job market* to a project-driven *freelancing-centered*

gig market, I often end up debating the ethical implications. Yes, there will be less job security in the future. There will be a massive global competition for skill. The ones who deliver excellent work will get paid much better than their lazy, low-quality competition. You may not like this trend. But this doesn't change the fact that it is happening right now. The purpose of this book is not to provide guidance on whether society should or should not allow this trend to play out—this book is about how you can benefit from it. But, taking a stand on this, I find it a highly positive development towards a more efficient workforce where everybody can focus on the work they like and are good at and outsource everything else.

To me, freelancing is already an integral ingredient of my existence. Here's how freelancing impacts every aspect of my professional life today:

- By earning my income as a freelancer myself, I generated cash to fund my online business *Finxter.com*.
- I hire freelancers for Finxter. The more Finxter grows, the more I rely on freelancers to create more value for my users.
- I host the most comprehensive Python freelancer course in the world at <https://blog.finxter>.

[com/become-python-freelancer-course/](https://www.udacity.com/become-python-freelancer-course/). This is my way of centralizing and sharing (but also learning from) the expertise of experienced freelancers across the globe.

My online business would have never been possible in its current form and scale without leveraging the efficiency gains of freelancing. Before freelancing became popular, large corporations practically monopolized exploiting the benefits of globalized labor. Today, every small business owner can access the global pool of talent. This way, new arbitrage opportunities open up for every small business owner who seizes them. Both business owners and freelancers benefit from this trend—as well as the people who, like me, work on both sides.

Freelancing allows people who are “losers” in the birth lottery – e.g., because of poor education or missing infrastructure – to perform from their talents and succeed beyond their wildest expectations. One of my freelancer students, Adam, quickly reached an earnings level of \$100 per hour right after finishing his math education—and consequently traveled through Europe earning top dollars and living inexpensively in a small and beautiful commune with his close friends. Another Finxter freelancer student, Shubham from India, now earns a full-time income as a content creator in the Python space—writing and learning from the comfort of his own home.

Freelancing allows you to build the lifestyle you desire—no matter where you’re coming from.

So how can *you* benefit from the global freelancing trend? One way would be to become an arbitrage trader: buy and sell freelancing services at the same time! You purchase the services you’re not good at. You sell the services you’re good at. This way, you’re continually increasing your hourly rate. A bit of napkin math will highlight the fundamental arithmetic of outsourcing.

Say you’re a fast coder: you write ten lines of code per minute. But you suck at customer service: you write 0.1 emails per minute. Now, you need to do both in your current position. To write 100 lines of code and answer ten emails, you need $10 + 100 = 110$ minutes. Most of the time, you’ll be answering emails.

Let’s assume further that your friend, Alice, has the exact opposite skill set: she writes only one line of code per minute (10x slower than you) but answers one email per minute (10x faster than you). To write 100 lines of code and answer ten emails, she’d need $100 + 10 = 110$ minutes. Most of the time, she’ll be writing code.

Both of you spend most of your time doing the work you suck at.

But what if you decide to hire each other? You hire Alice to answer your emails, and Alice hires you to do her coding. Now, you write 200 lines of code instead of

100 lines of code which takes you only 20 minutes. Alice answers 20 emails instead of 10, for which she needs 20 minutes. In total, you two finish your work in $20+20=40$ minutes instead of $110+110=220$ minutes! Together, you save $220 - 40 = 180$ minutes—3 hours per day—by outsourcing the work you suck at!

It's a stupid idea to do everything by yourself because you're incredibly inefficient in the vast majority of tasks. Fortunately, the freelancer disruption makes the world much more efficient. So, let's get some more clarity about the specific impact freelancing may have on your life.

In this book, we mainly focus on freelance coding using the Python programming language but most of the concepts will also apply to other programming languages and even outside the realms of computer science!

2.2 Freelancer: To Be Or Not To Be?

Becoming a freelancer is an exciting way of growing your business skills, participating in the new economy, learning new technologies, practicing your communication expertise, learning how to sell and market your skills, and earning more and more money on the side. Technology and globalization have expanded this opportunity.

And now it's up to you to seize it.

Video 3: *Python Freelancer - To Be Or Not To Be?*



<https://python-freelancer.com>

But what can you expect from this new path of becoming a freelancer?

First and foremost, freelancing is a path of personal growth, learning new skills, and earning money in the process. But in today's digital economy, becoming a Python freelancer is—above everything else—a lifestyle choice. It can give you fulfillment, flexibility, and endless growth opportunities. Additionally, it offers you a unique way of connecting with other people, learning about their exciting projects, and finding friends and acquaintances on the road.

While this sounds nice, becoming a Python freelancer can also be a struggle, with the potential to make your life miserable and stressful if you approach it with the wrong strategies and tactics. But no worries, this book is all about teaching you the right ones.

So is being a Python freelancer for you? Let's discuss the pros and cons of becoming a Python freelancer (see

Python Freelancing

 The Good	 The Bad	 The Ugly
Clean Python code (PEP8)	Short Deadlines	Not Getting Paid Sometimes (<10%)
Excellent earnings (>\$50/h)	Unfocused Clients	Deal with Horrible Legacy Code
Learn business and marketing skills	Clients expect full-stack knowledge	
Learn communication skills		
Learn PRACTICAL coding skills		

Figure 2.1: An overview of advantages and disadvantages when working as a *Python* freelancer.

Figure 2.1). The list is based not only on my personal experience as a Python freelancer—working for diverse projects in science, data analytics, and even law enforcement—but I have also assembled the experience of some of the top experts in the field.

The Good Things. There are many advantages to being a Python freelancer. Here are the most important of them:

- **Flexibility:** You are flexible in time and space. I live in a large German city (Stuttgart) where real estate prices are increasing. However, since I am

working full-time in the Python industry, being self-employed and 100% digital, I have the freedom to move to the countryside. Outside large cities, housing is exceptionally cheap and living expenses are genuinely affordable. I am earning good money matched only by a few employees in my hometown, and I don't have to compete for housing to live close to my employers. It's a huge advantage that can make your life wonderfully peaceful and efficient. Taken to the extreme, you can move to countries with minimal living expenses: earn dollars but pay rupees. As a Python freelancer, you are 100% flexible, and this flexibility opens you up to new possibilities for your life and work. There's a moral argument to this—no matter whether you live in the mountains of Tibet or in Manhattan, your opportunities to participate in the global economy have skyrocketed. Besides, maybe you like to travel back and forth between those locations—now you can too!

- **Independence:** Do you hate working for your boss? Being a Python freelancer injects a dose of true independence into your life. While you are not free from external influences (after all, you are still working for clients), you can theoretically get rid of any single client while not sacrificing your business. Firing your bad clients is often a

smart move because they demand more of your time, drain your energy, pay you badly (if at all), and generally don't value your work. In contrast, good clients will treat you with respect, pay well and on time, come back, refer you to other clients, and make working with them a pleasant and productive experience. As an employee, you don't have the freedom to fire your boss before finding a better one. This is a unique advantage of being a Python freelancer compared to being a Python employee.

- **Tax advantages:** As a freelancer, you are the owner of your business. I'm not an accountant — and tax laws are different in different countries. But in Germany and many other developed nations, your small Python freelancing business usually comes with a lot of tax advantages. You can deduct many things from the taxes you pay. For example, your computer, smartphone, car, living expenses, working environment, or even eating outside with clients or business partner. At the end of the year, many freelancers enjoy tax benefits worth tens of thousands of dollars.
- **Business expertise:** As a Python freelancer, you gain a tremendous amount of experience in the business world. You learn to offer and sell your

skills on the marketplace, how to acquire clients and keep them happy, how to solve problems, and how to keep your books clean, and invest and manage your money. Being a Python freelancer gives you a lot of valuable business experience. And even if you plan to start a more scalable business system, being a Python freelancer is a great first step towards your goal.

- **Paid learning:** While you have to pay to learn at University, being a Python freelancer flips this situation upside-down. You are getting paid for your education. As a bonus, the things you are learning are as practical as can be. Instead of coding toy projects in University, you are coding (more or less) exciting projects that have an impact on the real world.
- **Save time in commute:** Commuting is one of the major time killers of modern life. Every morning, people rush to their jobs, offices, factories, schools, or Universities. Every evening, they rush back home. They leave 1-2 hours of their valuable time on the streets every day, or 200 days a year. In ten years, they'll have wasted 2000-4000 hours—enough to become a master in a new topic of your choice, or write more than ten full books with a publisher and sell them on the marketplace.

Commute time to work is one of the greatest inefficiencies in our society. And you, as a Python freelancer, can eliminate it from your life. This will make your life easier, and you have an unfair advantage compared to other employees. You can spend the time on learning, recreation, or building more side businesses. You don't even need a car (I don't have one), which will save you hundreds of thousands of dollars over your lifetime: e.g., the average German employee spends 300,000 € on cars. Besides, save the environment!

- **Family time:** During the last 12 months of being self-employed with Python, I watched my 1-year old son walking his first steps and speaking his first words. Many parents who work at big companies as employees may miss out on their sons and daughters growing up. In my environment, most fathers and mothers do not have time to spend with their kids during their working days. But I have the time, and I'm very grateful for this.

Are you convinced that becoming a Python freelancer is the way to go for you? You are not alone. Thousands of Finxter students have checked out the Python Freelance Developer program¹—and I compiled this list of

¹<https://blog.finxter.com/become-python-freelancer-course/>

advantages from their personal feedback.

Testimonial Video 1: *How to Start Earning \$5000 per Month on Upwork as a Python Freelancer?*



<https://python-freelancer.com>

The Bad Things. But it's not all sunshine and rainbow being a freelancer. There are a few severe disadvantages which you have to consider before starting your own freelancing business. Let's dive right into them!

- **No stability:** It's not a trivial matter to reach a stable income as a freelancer. If you only feel safe when knowing exactly how much income you will bring home every month, you'll be terrified as a freelancer. And if you live from paycheck to paycheck and you haven't yet developed the valuable habit of saving money every month, being a freelancer can be very dangerous because a few bad months will push you out of business. You need to buffer the lack of stability by implementing a rigorous savings plan. There is no way around that. Especially in the beginning!

- **Bad clients:** Yes, they exist. If you commit to becoming a freelancer, you will get some bad clients for sure. They expect a lot, are never satisfied, give you a bad rating, and don't even pay you. You might as well already accept this fact and write 10% of your income off as insurance for freeing yourself from those bad clients. I'm not kidding — set aside a fraction of your income so that you have the ability to fire a bad client immediately. You'll save yourself a lot of time, energy, and ultimately money (and time is money in the freelancing business).
- **Procrastination:** Are you a procrastinator? If so, it may be difficult for you to start a freelancing business because it requires that you stay disciplined. No boss will kick your ass if you don't perform. All the initiative is on you. Of course, if you have established a thriving freelancing business, new clients will line up to do business with you. In this case, it may be easier to overcome procrastination. But, especially in the early days when you have to make a name for yourself, you must show the discipline that this job profile requires. Make a clear plan for how you will acquire clients. For example, if you are a freelancer on Upwork, make it a habit to apply for ten projects every day. Yes, you've read this right. Commit

first, figure out the specifics later. You can always hire freelancers of your own to help you if you find yourself with more projects than you can handle. Or you can even withdraw your services. But doing this will ensure that you never run out of clients, which will practically guarantee your success as a freelancer in the long run.

- **Solitude:** If you are working as an employee at a company, you always have company, quite literally. You will meet your buddies at the coffee corner, attend seminars and conferences, and present your work to your group. You'll generally get a lot of external input regarding upcoming trends and technology. As a freelancer, you cannot count on these advantages. You have to intentionally structure your day well, read books, attend conferences, and meet new people. Otherwise, you will quickly fall out of shape with both your coding and communication skills because you regularly work on your own. The ambitious way out is to continually grow your freelancing business by hiring more and more employees.

2.3 What's unique in Python freelancing compared to general IT or coding freelancing?

Python is a unique language in many ways. The gold standard provides strict rules (PEP standards) on how to write “Pythonic code”. Coding in Python has the significant advantage that you will usually work on clean and standardized code projects that are easily understandable. This is in stark contrast to languages such as C, where it's hard to find common ground from time to time.

Python is an old language that, naturally, comes with a lot of legacy code. For example, many projects use Python 2 or old frameworks that don't exist anymore. Managing the various dependencies of a Python project can be a curse. Make sure to ask more experienced developers and the original creators of a certain codebase for support - this will make your life much easier. And they usually don't mind talking about their code.

The Python ecosystem is incredibly active and vivid — you'll find tons of resources about every question you may have. The documentation is excellent. Many languages such as COBOL (yes, still alive and well), Go,

Haskell, and C# are documented poorly compared to Python. Python's excellent documentation will help you a lot when trying to figure out the nasty bugs in your (or your clients') code.

Python's entry barrier is relatively low, which is partly a result of the comprehensive documentation, and partly a result of the easy-to-understand language design. Python is clean and concise — there's no doubt about that.

Although the number of code projects in Python is snowballing, so is the international competition. Many coders are attracted to Python because of its excellent documentation and suitability for machine learning and data science. Thus, the significant advantage of writing Python code that is fun can also be your biggest challenge. Competition can be fierce. However, this is usually only a problem if you are just starting and have not yet made a name for yourself. If you are doing good work and focus on one sought-after area (e.g., machine learning), you will have a good chance that plenty of clients will compete for your valued time!

If you plan to start your career in machine learning or data science, Python is the 800-pound gorilla in the room. The library support is stunning — more and more people are migrating from Matlab or R to Python due to its generality and the rise of new machine learning frameworks such as TensorFlow.

Knowing about those, let's dive into the more worldly benefits of becoming a freelance developer.

2.4 What's the Hourly Rate of a Python Freelancer?

Today, many Python freelance developers earn six figures. How much can you expect to earn as a Python freelancer? The short answer is: the average Python developer in the US makes between \$51 and \$61 per hour.² In other words, if you work eight hours per day for 230 days per year earning \$55 per hour, you'll reach six figure income level.

The variation in earning potential is high, but so is the quality of the Python freelancers in the wild. Take the average salary as a starting point and add +/- 50% to account for your expertise. If you work on the side, for example 8 hours each Saturday, you might earn \$400 extra per week – or \$1600 per month before taxes. Initially, your hourly rate may be a bit lower because you need to find freelancing clients first — an activity that may cost 20% or so of your total time.

What's your market value as a Python developer? Based

²Sources: <https://blog.finxter.com/whats-the-hourly-rate-of-a-python-freelancer/>

on my estimation, it's roughly 1.07 million USD.

I base this calculation on a standard and straightforward way to value companies. In a way, you're a one-person company when you're selling your coding skills to the marketplace. This holds whether you're an employee or a freelancer. When estimating the value of a company, analysts often use multiples of the company's yearly earnings. Let's take this approach to come up with a rough estimate of how much your Python skills are worth.

Let's say we calculate the value of a company by multiplying its earnings with 10, i.e., an investor would have earned their initial investment back after ten years. Now, let's apply this 10x multiplier to your earnings when working as a Python freelancer earning about \$55 per hour.

Your yearly earnings may be \$107,800 if you work for 40 hours per week, 49 weeks per year, earning \$55 per hour. With yearly earnings of \$107,800 and a multiplier of ten, the value of your *personal freelancing company* would be \$107,800 times 10 equals **\$1.078 Million**. And this is when assuming a relatively conservative earnings multiplier and average earnings. So even for an average programmer, Python can be a million-dollar skill!

And the value of a top-5% coder can easily be 10x that of the average coder.

So if you want to thrive with your own coding business, you need to think strategically. Not sharpening your programming skills will cost you hundreds of thousands of dollars. The more you'll learn, the more you'll earn. So you simply cannot invest too much time, energy, or even money in your education.

Billionaire investor Warren Buffett is a strong proponent of education, value, and integrity:

“Ultimately, there’s one investment that supersedes all others: Invest in yourself. Nobody can take away what you’ve got in yourself, and everybody has potential they haven’t used yet.”—**Warren Buffett**

2.5 Avoid This Productivity Killer

Before you learn about the concrete skills of a Python freelancer in the next section, let me quickly dive deeper into one freelancing advantage: working from home saves you a lot of *commute time*.

Commute time is a massive productivity killer that drains your energy silently but surely. Even if you used commute time productively by listening to audiobooks or reading business books, it’s still a waste of your time compared to what you *could* be achieving instead. Roughly

speaking, commute time is *consumption time* that takes away from your *production time*.

When I became self-employed, my work productivity skyrocketed. At the same time, work became more relaxed and less stressful. When I analyzed my days to find out the reason for this, it struck me: *no commute time*.

Suddenly, I had a lot more time and more energy to create content. Skipping commute time simply gave me more resources to work with. The same happened to my Finxter partner Lukas when he reduced his commute time from two hours to half an hour.

Working from home means that you don't have these enormous drains of energy every day—even more so if you're involved in a lot of office politics.

Many research studies show that having a long commute time makes you less happy:

*“It is found that longer commute times are associated with lower job and leisure time satisfaction, increased strain and poorer mental health.”*³

It's one of the top ten influential factors for your happiness — even more important than making a lot of

³*How commuting affects subjective wellbeing*, Clark et al. 2019

money. Working from home is one of the best advantages of being a Python freelancer.

You'll save 1-2 hours per day commute time. If you invest this commute time into your dream project every day, you'll be wildly successful within a few years. You could write 2-3 books per year, finish ten small web projects per year, or learn and master an entirely new skill such as business or marketing.

2.6 What Does it Take to Be a Freelancer?

Surprisingly, many people fear taking their first step towards freelancing. They are hesitant because they believe that they don't have enough knowledge, skill, or expertise. But this limiting belief harms their ability to progress toward their dream life.

The only thing it takes for certain to become a freelancer is to be human (and this may not even be a requirement in the decades to come). Everything else, you already have in more or less rudimentary form:

- **Communication skills.** You need to ask and respond to questions, figure out what your clients want, be responsive, positive, enthusiastic, and helpful.

- **Technical skills.** There's always an underlying set of technical skills for which clients hire you. They may want you to develop their next website, write their copy and ads, create valuable content, or solve other problems. Before being able to deliver the solution, you first need to have the technical skills required to develop the solution.
- **The ability and ambition to learn.** Chances are you won't already know enough to solve the client's problems. So you need to learn. There's no way around it. If you are willing to learn, you can solve any problem — it's just a matter of taking the time. And as you learn more in your area of expertise, all freelancing gigs that follow will become a little bit easier.
- **Time.** All of us have the same number of hours in a day. You already have enough time to become a freelancer. You just need to focus your effort — and maybe skip a Netflix episode this evening.

You see, there's nothing special about what you need to have in order to become a freelancer. You already have everything you need to get started. Now it's just a matter of persistence.

André, one of my Finxter freelancing students, asked me the following question: *“How much do I have to learn to*

become a Python freelancer?” My answer is straightforward: Nothing! Start right away, no matter your current skill level, and learn as you create value for clients.

But I know that for many new Python coders, it’s tough to start right away. Why? Because they don’t yet have the confidence to start taking on projects.

And the reason why is because they never quite finished a Python project — and they are full of doubt and have low self-esteem. They fear not being able to finish their freelancer project and earn the criticism of their client.

If you have to overcome this fear first, I recommend working on some old or archived freelancer projects. There are thousands of such projects in Python, data science, web development, machine learning, or Blockchain. To make it easier for you, I have compiled a list of 10 suitable Python projects for beginners to intermediates and published a blog article about them⁴, which you can start doing today to improve your skill level and gain some confidence. Real freelancers have earned real money solving these projects — so they are as practical as they can be.

I recommend that you invest 70% of your learning time finishing these projects. First, select a project. Second, finish the project - no matter your current skill level.

⁴<https://blog.finxter.com/how-real-freelancers-earn-money-in-2019-10-practical-python-projects/>

If you are a complete beginner, it may take weeks to finish a project that may have earned the freelancer \$20. However, by finishing the project anyway, you'll improve your skill level. Why not finish your first real project next, earning your first \$20 yourself? Even if it's hard to earn your first \$20, the next \$20 will be much easier, and the next \$20 will be easier again until the money starts flowing in greater and greater quantities. Each time you earn money with your Python skills, it builds your confidence as a professional freelancer—it's a vote from the marketplace that you are valuable. This vote of confidence is priceless.

For example, say the first project takes a couple of weeks to complete. Due to your learnings from the first project, the next one will take just one week rather than a few. And the next project will take you only three days, then one day, then one hour, then half an hour. And this is how your hourly rate increases exponentially until you reach some convergence. As this happens, you must specialize even further. Select the skills that interest you and focus on those skills first. Always play your strengths.

2.7 Can I Start Freelancing as an Intermediate-Level Python Programmer?

For sure! You should have started much earlier. Have a look at the income distribution of Python freelancers (see Figure 2.2).

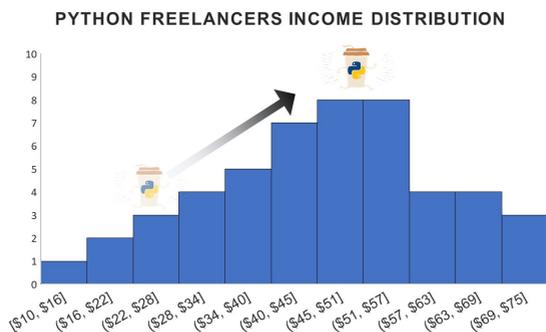


Figure 2.2: Income distribution of Python freelancers.

The hourly rate as an online Python freelancer resembles a Gaussian distribution around the average value of \$51 per hour. So if you are an average Python freelancer, you'll likely earn \$51 per hour in the US.

I have gained a lot of experience at the freelancing platform Upwork.com. Many beginner-level Python coders

earn great money finishing smaller code projects. If you are an intermediate-level Python coder interested in freelancing, you should start making money as soon as possible.

The significant benefit is that not only are you getting paid to learn and improve your Python skills further, you're also learning the right skill sets that will make you successful online: communication, marketing, and coding (the essential, practical stuff).

Only practice can push you to the next level. And working as a Python freelancer online will give you a lot of training for sure!

2.8 Are You too Old to Become a Python Freelancer?

The short answer is no. You are not too old.

The older you are, the better your communication skills tend to be. Having excellent communication skills is the main factor for success in Python freelancing.

Just to make this crystal clear: there are plenty of successful freelancers with limited technical skills that earn more than highly-skilled employees. They are successful because they are responsive, positive, upbeat, and committed to making their clients' lives easier. That's

what matters most as a freelancer.

As you see, there's no age barrier here – just double down on your advantages rather than focus too much on your disadvantages.

Video 4: *Are You Too Old to Become a Freelance Developer?*



<https://python-freelancer.com>

2.9 Are You Too Young to Become a Python Freelancer?

Again, the answer is no. You are not too young.

Was Mark Zuckerberg too young when he started Facebook at age twenty? Was Warren Buffet too young when buying his first stocks at age eleven? Was Magnus Carlsen, one of the world's top-rated chess players, too young to start playing chess at age five?

Young age may have disadvantages. However, it has advantages too. And you should double down on those advantages by relentlessly pursuing maximum value for

your clients—and focus on learning and mastering emerging technologies you’re excited about! Old people often remain stuck in their mode of thinking—at least the ones that don’t keep learning throughout their lives.

If you are young, you’ll learn quickly. By focusing your learning on highly practical tasks such as solving problems for clients by using Python code, you’ll create a well-rounded personality and skill set. If you do this, you’ll have an excellent chance to build yourself a thriving business within a few years.

Video 5: *Are You Too Young to Become a Freelance Developer?*



<https://python-freelancer.com>

If you want to become a Python freelance developer and create your coding business online, check out our free webinar. Just click the link, register, and watch the webinar immediately. It’s a replay so you won’t have to wait even a minute to watch it. The webinar is a PowerPoint presentation that will give you a detailed overview of the Python freelancing space.

Complete Webinar Guide

How to Build Your High-Income Skill Python



[https://blog.finxter.com/
webinar-freelancer/](https://blog.finxter.com/webinar-freelancer/)

Are you motivated to transform your life, have more time for your friends and family, and work from the comfort of your own home while earning great money? Let's dive into the central part of this book: the expert secrets to 10x your business success.

Part I

Overview: \$0 to \$1000

Can you become a business person that's also an expert coder? Knowing only a few things about you, I can confidently answer this question: yes!

Let me quickly share a personal story with you. I vividly remember my first semester studying computer science more than ten years ago.

I knew almost nothing about coding. To be precise, I had never written a single line of code. The only insight I had about computer science was a piece of knowledge obtained from the only person in my environment who worked in the IT sector - my uncle, a professional software developer. He had quickly shown me the programming language *Visual Basic* and told me something that you may find helpful, too:

“Learning a programming language is like learning a natural (spoken) language — it becomes easier the more languages you’ve already mastered. Two things underlie every single language: concepts and syntax. Concepts are hard; syntax is easy. But if you understand the concepts, you can directly build upon the concepts from an old language when studying a new language. Therefore, you should focus on the concepts!”

Equipped with such high-level advice, I tried my luck as an upcoming computer scientist. In hindsight, I didn't even know what I was doing—I only chose computer science because I loved puzzles and riddles. And it didn't

hurt that the bar to get into the program was very low. Of course, people told me about the great job opportunities that came with becoming a computer scientist. But I didn't care about them at that point in my life. I had no clue what I was going to do with these new computer science skills I was working hard to acquire.

As a kid, I dreamed of becoming self-employed as a writer and starting my own business. For the people around me, this was weird because not a single person in my environment succeeded in their own business. Most of my family members were employees and skeptical about self-employment—all had seen my grandfather fall into bankruptcy by trying to create a highly-leveraged business (*be careful taking on too much debt*). His case often served as a negative example to showcase the dangers of creating one's own business.

What about you: Have you ever dreamed of becoming self-employed? Statistically, more than half of the Americans want to be their own boss and create their own business. But many of them fear the risks associated with being self-employed. Like my grandfather, you may fear being forced to file for bankruptcy and losing everything.

Debt increases the pressure to serve creditors, and it'll cost you peace of mind. Some people fear taking the big leap forward, getting rid of their day job, and risking

not being able to support themselves and their families. So they stick with their dreaded jobs—paralyzed and unsatisfied with their current situation.

If people try to force you into an “either-or” decision, why not aim for both? Why not stick with the day job and gradually build your business on the side? And, even more critically, grow your business experience? This strategy is the one I followed, and it worked.

You too can spend your Saturday mornings working on your side business and earn double income. You too can improve your technical skills and business expertise at the same time. You too can make yourself more robust, valuable, and knowledgeable. And, when the time comes, you can grow your side business into a full-time income.

But are you good enough? We’ve already discussed this in the previous chapter: You are! The fear of not being good enough is a limiting belief we all face—and you must get rid of it. Only by risking not being good enough do you have the chance to actually become good enough. If you’ve any ambition, you must get rid of this fear immediately.

I started with zero knowledge in computer science, and, frankly, little innate talent. Most people in my environment doubted that I could become successful in business. Fast forward ten years, and I got my PhD in com-

puter science, and created a thriving coding business online based on \$100+ per hour work in the freelancing industry. I love my work—it’s fun, exciting, and I learn a lot in areas I’m interested in such as Blockchain development and machine learning. I work less and earn more, and I’ve got plenty of time to read *Lord of the Rings* with my kid. I’m studying computer science every day, just for fun, on my quest for continuous improvement, and can afford everything I need to live a great and satisfying life.

Video 6: *One Line of Code Every Day—A Powerful Habit*



<https://python-freelancer.com>

I tell you this because I truly believe that becoming self-employed on the side was one of the best decisions I ever made. Yes, I had to overcome a lot of objections, doubts, and fears. And I had to work hard in the beginning, as I built my side business—hard work is expected! But by doing it, I not only created a thriving side business, but I used my growing experience in my “real job” to become more valuable as an employee too.

In the following chapter, I'd like to share some of my most significant insights with you – to save you from working on the wrong things for weeks, if not months. Your time truly is your most valuable asset, especially if you decide to become self-employed. Every unit of time passes and never comes back. Money is plentiful, but time is limited. Don't let all of those opportunities pass. The best time to start your own business was ten years ago, in which case you'd now have a profitable freelancing business. The second best time is now!

So why not start your own (side) business as a Python programmer? Side-hustling Python programmers have zero debt risk, zero liquidity risk, and zero downside risk. You only need to give up some of the stupid consumption time, we all indulge in, watching Netflix or playing World of Warcraft. Yet, there's tremendous upside potential towards financial safety and stability.

Read the upcoming expert secrets. If you encounter a secret that is likely to improve your freelancing business, implement it immediately. Adopt a “do it now” mentality with a strong bias towards action! You get maximum value out of this book if you revisit these expert tips from time to time. Each time you read over the list, you'll find another expert tip that will help you grow your business. You don't have to follow each expert secret to the letter, and you don't have to read the whole thing. Just use it as a source of inspiration for

how to increase the value of your business today. If the expert secrets can accomplish just that, you'll already have a positive return on your invested time and money.

Video 7: *Finxter Python Freelancer Course – Walkthrough Towards Six Figures*



<https://python-freelancer.com>

Expert Secret: Low Risk High Gain

Many Americans dream of being their own boss by becoming self-employed. But most never take the leap because of their limiting beliefs. They fear creating something new in their life, being dependent on other people like clients or co-founders. They fear telling their bosses, coworkers, or families about their new business ventures. They fear not being good enough to charge money. These fears are self-fulfilling prophecies because they become true as they hold on to them. The lack of exploration and experience will prevent them from becoming good enough in the first place. The only way out is to try the things they fear and improve as they go. If you're one of those people, you must get rid of this fear as soon as possible if you have any ambition in life.

Video 8: *Low Risk High Gain? How to Create a Coding Side Business*



<https://python-freelancer.com>

Let's assume that you are currently working as an employee in your primary job, or you're a computer science student thinking of building your own business in the mid- or long term.

If you want to start your own business at any point in your life, you should start it now. It doesn't matter what else you are doing—even if you work most of the time in your primary job. I don't recommend that you get rid of your main job and go all in right away.

Instead, a much better way of becoming self-employed is to work part-time on your business. Maybe you decide to work for 5-10 hours every Saturday. Over time, you'll be able to create a very sustainable, robust, and long-term side business.

When building your business on the side, you're more robust because even if there are market changes and you lose your job, you've got something to fall back on. You can always go back to the business and do more of what already works — now full-time. Or, if the business fails,

you still have a job that can pay for your basic needs.

But expect that neither the business nor the job fails. They will coexist, and there will even be synergies where one feeds the other. For example, when working as a doctoral researcher at University, I would also write blog articles about my research topics - which was great for my main job and great for my side business as a Python freelancer.

As a result, you'll earn more, save more, and learn more. Financial stability is a massive advantage for you as a small business owner because you can think long-term. The job brings in a steady stream of income while you can build a robust, stable, long-term asset (the business). For example, you can write a book, give courses, steadily grow your online community, and grow your client base and testimonials.

Video 9: *How to Earn \$4000/M Passive Income as a Coder?*



<https://python-freelancer.com>

You can also write blog articles. Many people wrongly believe that this is not a profitable use of their time. But if you write a blog article now, it will take months

to rank and then bring in a consistent flow of traffic, month after month. You can then start selling stuff on your website — or monetize your traffic with ads, which is truly passive income. Blogging is a long-term strategy.

You can even go for businesses with high-entry barriers because you have an unfair advantage: you are not dependent on your business. So you can try out things and experiment until something works. For example, you could start long-term code projects just for fun that don't have to show a positive return on investment for a long time. Most new market participants need to make money quickly — that's their heaviest liability: they cannot take the long-term perspective.

But you can afford to do so because you work on your business on the side. You can grow your business by reinvesting the cash it earns because you don't depend on it.

That is how I created the Finxter.com platform for learning Python. I could think long-term. It took me years to build. I wrote everything from scratch in Python — it was tedious, but I had money coming in working as a researcher in computer science. There was a decent entry barrier because new market participants couldn't invest a lot of time creating new platforms — they were running out of money. If they had to pay a team of developers, they'd often run out of money faster.

In summary, if you are thinking about creating your own business, don't wait - start now. Think radically long-term.

An excellent way of creating a side business is to become a Python freelancer. You can learn the business side of things, become better, build your reputation and knowledge, and earn good money proportionally to the time invested. You can then build upon your new skills to attract better and better clients - all while growing your skills.

Video 10: *Get More Clients as a Freelance Developer with This One Simple Trick*



<https://python-freelancer.com>

Action steps: Answer the following questions in written form.

- What are your life goals?
- How much monthly income will you need to fund your dream life?

- How much will you earn from your job, your (side-) freelancing venture, your passive income streams, and your investments?
- Create a simple spreadsheet answering the previous questions for each of the next ten years!

— 4 —

Expert Secret: Gain an Unfair Advantage

Do you know how much money you currently earn? Publicly available US government statistics indicate that an average employee works 1811 hours per year. As an employee, it is difficult to earn more than \$90,000. In

Description	Salary (year)	Salary (month)	Salary (hour)
Professor	\$98,423	\$8,201	\$54
School Teacher	\$67,007	\$5,583	\$37
1-line Office Supervisor	\$56,170	\$4,681	\$31
Median Wage	\$44,564	\$3,713	\$24
Office Assistant	\$31,524	\$2,627	\$17

Figure 4.1: Table of salaries.

fact, the median wage of all workers in the US is \$24. For example, if you are a student, you are earning -\$4 per hour, school teachers earn \$37 per hour (see Figure 4.1). If you push yourself very hard and become an extremely skilled employee, you might become a university professor with a yearly salary of \$98,423 or \$54 per hour. First, know your hourly wage. Second, improve it.

So how can you increase your value to the marketplace? First, by creating a new high-income skill: *Python development*. Second, by learning how and why to switch the road from being a full-time employee to being at least part-time self-employed.

In Figure 4.2, you can see the income distribution of Python freelancers. The median wage of a Python freelancer is \$51 per hour! In other words, an ordinary, self-employed Python freelancer easily reaches the university professor's income level. Think about this: can you become a university professor? It's totally up to you to answer this question. But you can undoubtedly become an average-skilled Python freelancer, can't you?

The key takeaway is that intermediate-level Python freelancers today earn six figures easily. Statistically, they earn a healthy \$100,000 yearly gross income.

If you work on the side, let's make it 8 hours each Saturday, you will earn \$400 extra per week - or \$1600 per

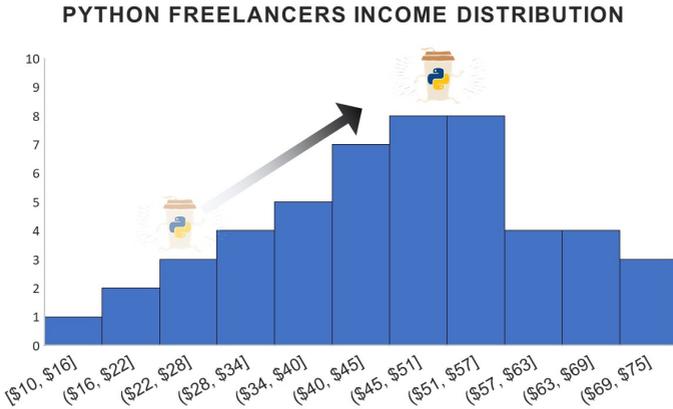


Figure 4.2: The figure shows the income distribution of Python freelancers (relative frequency) based on various data sources. An important observation is that your skills are always valuable—even if you’re relatively untrained. The best strategy is to start now and improve your hourly rate over time as you gain more and more practical experience.

month (before taxes).

Video 11: *How to Go Full-Time (\$3000/m) as a Python Freelancer [Working Only Part-Time Hours]*



<https://python-freelancer.com>

Action steps:

- When and how do you plan to become self-employed?
- Can you become self-employed on the side?
- Write down how many hours you can invest per week.
- Write down your goal hourly rate.

Feel free to download worksheet 1 as a PDF to complete the action steps: <https://python-freelancer.com/>

Expert Secret: Know Your Why

Video 12: *Code From Home! How to Be Happier and More Successful*



<https://python-freelancer.com>

How would your life look if you only needed to work part-time as a Python freelancer doing projects that you like?

I have already stressed the first point: Imagine you work from home and see your kids growing up, having the flexibility to spend more quality time with your spouse. But there is also an equally important point if you need to take care of your family. And that is: you can increase

your value to the marketplace. And there is virtually no upper limit of your hourly rate. If you are an employee, you will definitely have an upper limit - you have seen that a professor earns \$53 per hour. But many freelancers earn \$100-\$200 per hour. It all depends on how expensive you can make yourself for the marketplace.

For some of my students, being a Python freelancer is also a lifestyle choice. For example, one of my students is successfully employed in the US and earns good money there. But his dream is to go back to India to his family and work as a Python freelancer. Doing this, he earns dollars and pays rupees for his living expenses. Why not enjoy the benefits of globalization?

It's also good to diversify your income streams. You could spend one day per month to earn \$400-\$500 per month as an additional income for you or your family to lead a better life or even save it for later.

Finally, being a Python freelancer is also a lot of fun. You have to stretch your abilities regarding Python but also your soft skills such as communication ability and language skills. If you are not a native speaker, freelancing is a great way to improve your skills while you are getting paid to do good work for other people.

Can you already see yourself working as a Python freelancer? Let's gain some practical experience before you overthink it.

Action steps:

- Block at least 8h per month to work on your business.
- Write ten emails to business owners and offer them your services for \$500 per month. Make it valuable to them! You may create a social media account for them, write blog articles, do some administrative things. Or simply ask them what they would need from you to pay you \$500 per month.

Feel free to download worksheet 2 as a PDF to complete the action steps: <https://python-freelancer.com/>

Expert Secret: Consultants Go Big

There are three main ways of becoming a Python freelancer. The first is being a consultant working for a big company. The second is to be a freelancer working on a platform such as Upwork or Fiverr doing mostly smaller tasks. The third option is to create your own platform (for example, creating your website and driving traffic to it). The last one, I call the *hybrid approach* because you have some elements of both previous options. Now, we will dive a bit deeper into each of these options.

Video 13: *How to Get Traffic for Your Online Coding Business FAST? It's Not Advertising!*



<https://python-freelancer.com>

The first way of becoming self-employed is to work for one or a few big clients as a consultant. Working as a consultant has some advantages. You work in a business-to-business setting, which allows you to tap into enormous earning potentials. There is a lot of money in business-to-business – especially if you focus on high-ticket sales.

However, many people working as consultants heavily rely on one or two big clients. They are not diversified at all. And if you work for a single big company, you will have minimal freedom in terms of your projects and working conditions, and minimal job security. Many Python consultants report that the pressure is hard, and it feels like working as an employee.

If you like the corporate environment, however, you should gravitate towards this option because it's the easiest way to get massive results fast!

Action steps:

- Would you rather work for one big corporate client or ten small business owners?

Expert Secret: Join Freelancing Platforms

The second way is to sell your services as a Python freelancer on existing freelancing platforms such as Upwork and Fiverr. These platforms are very convenient. You could start today, create your freelancing account and begin with your first gig in the evening. Then you solve the jobs that take maybe a week or even only a day. This way is very flexible, and you can learn fast and without too much pain or commitment.

You can also have a small feedback cycle by acquiring a client, doing the work, finishing the job, and getting reviewed. Over time, you will become an expert in the soft skills and communication part, and you will learn about many different areas where your Python skills can

help people out.

Video 14: *How Long Does It Take To Become A Python Developer*



<https://python-freelancer.com>

It's the perfect option to get your foot in the door and to converge, job-by-job, toward your final specializations, in case you want to specialize in increasing your earning potential on the freelancing market.

Finally, there is no startup overhead. Marketing is simple. If you have an attractive profile and good ratings, you'll get recommended by the freelancing platforms and you'll soon get more gigs than you can handle.

Of course, nothing is perfect. If you work as a freelancer, you don't own the platforms. You don't own the clients. Later, we will see that you can also acquire clients from the platforms and create your own client database to mitigate this last point. Moreover, these platforms get a significant cut of 25% for each job. That's quite something.

Overall, working as a freelancer on these platforms is all about getting testimonials, skills, and experience.

Video 15: *What Are the Best Freelancing Sites for Coders?*



<https://python-freelancer.com>

If you're already successful, you don't need more clients. There will be many more clients than you can serve in a limited time. Demand is much higher than supply, which will enable you to raise prices—your ultimate goal.

However, you need to gain momentum. Clients will not flock to you in the beginning. It's hard to land your first client. It's also hard (but a little bit less so) to land your second client. For each new client, it tends to become easier—if you provide more value than you take in payment (if you only internalize this one rule, you'll earn tens, hundreds, or even millions of dollars more in your lifetime).

To gain momentum and experience, you can join freelancing platforms in the beginning. As a rule of thumb: working on freelancing platforms for 10-20 clients (1-2 months) is a good use of your time—even if the pay sucks. After that, you should gradually switch to your own client acquisition strategy. The reason is that freelancing platforms take a significant cut of 20% or more

of your pre-tax earnings.

Action steps:

- Do you want to start on a freelancer platform?
- Is this your long-term goal or only one step up the ladder of your career?
- Create a freelancing account on all major platforms: Fiverr, Freelancer.com, and Upwork.
- Find the platform that fits you best and focus on mastering that platform. If you're unsure about which fits best, continue reading the next Expert Secret Chapter 8.

Feel free to download worksheet 2 as a PDF to complete the action steps: <https://python-freelancer.com/>

Expert Secret: Best Freelancing Sites

There are many different ways of starting your Python freelancing adventures. Many freelancing platforms compete for your time, attention, and a share of your value creation. These platforms are a great way to start your freelancing career as a Python coder, gain some experience in business and coding, and get some testimonials to kick off your freelancing business. But keep in mind that they are only the first step, and in the mid-term you should strive to become independent of those platforms if you want to avoid global competition for each project in the future.

These are the best places to start your Python freelancing career and get clients fast (ordered by my recommendation):

8.1 Upwork

Upwork places a significant focus on quality. This is great for clients because it ensures that their work will get delivered—without compromising quality. For freelancers who are just starting, Upwork poses a significant barrier of entry. Oftentimes, new profiles will get rejected by the Upwork team. They want to ensure that only clients who take their freelancing jobs seriously will start on their platform. However, the relatively high entry barrier also protects established freelancers on the Upwork platform from too much competition. There is no price dumping because of low-quality offers, which ultimately benefits all market participants.

8.2 Fiverr

Fiverr initially started as a platform where you could buy and sell small gigs worth five bucks. However, since then, it has grown to a full-fledged freelancing platform where people earn six-figure incomes. Many jobs earn hundreds of dollars per hour, and many freelancers make a killing, especially in attractive industries such as programming, machine learning, and data science.

8.3 Toptal

Toptal has a strong market proposition: it's the platform with the top 3% of freelancers. Hence, it connects high-quality freelancers with high-quality clients. It's tough to become a freelancer at Toptal: 97% of the applicants will not enter the platform. However, if you manage to join Toptal, you can greatly benefit from the best-in-class hourly rates. You can easily earn \$100 per hour and beyond. Also, the high entry barrier ensures that the freelancer stays the valuable resource—he or she doesn't become a commodity like on other freelancer platforms. If you are an upcoming freelancer, you should aim to join Toptal one day.

8.4 Freelancer.com

Freelancer.com is the go-to resource for beginners with a low barrier of entry and opportunities for everyone. For years, this site has been my recommended starting point to gain experience and finish your first projects, because Freelancer.com can help you obtain your first testimonials—while getting paid for learning and polishing your skills. However, in recent years, I found that Fiverr offers many of the same advantages while being more active, larger, faster-growing, more fun, and easier to use. On the other hand, Freelancer.com is a

great site with countless freelancing projects. A great resource is the archived freelancing projects, which help you get some real-world projects for training purposes.

Action steps:

- Before you read on, decide on one platform and stick to it for at least one year. Commitment is king! My recommendation is Upwork – and, no, I’m not affiliated with Upwork.

Part II

Skills: \$1000 to \$2000

Expert Secret: Focus on Skill Creation

Are you broke? Especially in the US, but also in Europe, many people are broke. You may ask: What is the definition of being broke, anyway? Here it is: You don't have any leftover money to account for individual circumstances. It's that simple.

The average debt of a college student is \$27,225. Many political commentaries consider this a significant problem in the US. But is it really such a big problem? The famous consultant Dan Lok, who calls himself the *world's highest-paid consultant*, has a somehow different view on debt. Let me show you one of his statements:

“You don't have a debt problem, you have an income problem. You don't have an income problem. You have a SKILL problem!”

Because if you are skilled, you can always sell your service at a higher rate. Suppose there are two employees: Bob and Alice. Bob has \$10,000 in assets and a yearly income of \$31,524. Bob is debt-free. So many people would consider Bob's financial state as convenient, when in fact he is broke—a negative life event such as an illness in his family forcing him to pause work for a year will wipe him out in an instant. And even if we remain optimistic, it is really hard to see how he can better his financial position significantly on his relatively small yearly income. Alice, on the other hand, has an inconvenient \$100,000 in debt. But Alice can sell her skills at a rate of \$131,000 per year. What she doesn't have in assets, she overcompensates in skills. Alice at least has hope to transform her skilled labor into new assets to create wealth for herself and her family.

In two years, Alice can easily outsave Bob by tens of thousands of dollars—even if she starts with a lot of debt. You can also see this scenario in Figure 9.1. The higher the skills, the more you can expect to earn.

Action steps:

- Make a list of all the skills you have (at least 20 skills).
- Assign a number to each skill (0 - beginner, 1 - intermediate, 2 - expert, 3 - top 1%)



Figure 9.1: Starting point of Bob and Alice. Who has the higher net worth? Who will become richer?

- How can you give more value applying your skills with scores 2 and 3.

Feel free to download worksheet 3 as a PDF to complete the action steps: <https://python-freelancer.com/>

Expert Secret: Your Individual Success Habit

The success formula is simple. You start working as a Python freelancer now, no matter your current skill level. Then, you keep increasing your value to the marketplace by taking jobs and learning as you deliver value to clients until you have reached at least an average Python freelance level. At this point, you will charge \$51 per hour.

Would you consider a daily income of \$100 as a full-time income? According to US statistics, earning \$3000 per month is above the median salary. Thus, to make a full-time income, you may need to work for only two hours on your core freelancing activities, given an average income of \$51 per hour as a freelance coder. The

rest of your time, you are free to spend with your family, rest, learn, or find even better freelancing jobs. That's it. The strategy is simple but effective, nonetheless. It provides you with a clear and manageable path to your new freelancing lifestyle.

Action steps:

- Choose one habit that will transform your business, and complete it *every day*. An idea would be to write a valuable and friendly email to a lead or client.
- Set up a habit tracker and complete it every day. I use a smartphone app—any simple device where you can check boxes daily should suffice for instant gratification. The real reward will come in terms of growth, appreciation, higher income, and success.

Feel free to download worksheet 4 as a PDF to complete the action steps: <https://python-freelancer.com/>

How can you increase your value to the marketplace so you can quickly work at the average Python freelancing level? What's the magic key that will allow you to open the door to your dream clients? First, use research insights of psychology to build trust. Second, become

a specialist rather than a generalist. Third, leverage network effects.

Let's dive into each of those three keys next.

Expert Secret: Earn Trust

Clients will pay you more and give you better jobs if you are trustworthy and appealing to them. Consequently, you should focus on earning trust and make yourself more attractive to clients in every way you can. I did a small experiment and searched for the keyword “Python” on the freelancing platform Upwork. All recommended freelancers have a very high job satisfaction rate. Because Upwork recommends them, the demand for their services is much higher which leads to higher hourly rates. This leads us to the simple observation: *Trust is vital for your success in business.*

So how do you earn trust? First and foremost, you collect positive ratings. The more you have, the more money you’ll earn. If you buy property in the real estate sector, it’s all about location. In the freelancing

industry, it's all about ratings. You need to engineer your ratings by using the tips discussed in this book. If you have great ratings, you will always find jobs, no matter how good your external achievements are. You don't even need an academic degree; you can find the best jobs if you have good ratings. With good ratings, you will always find good, well-paid, and attractive jobs. Ratings are king.

How to get good ratings? There are five basic ways.

First, communicate well. Be responsive, very positive, and generous with your offers.

Second, acquire a lot of (Python) skills.

Third, over-deliver. If your task is to give a client 100 Python puzzles, and you send them 110 Python puzzles, you can almost be sure to get the 5-star rating on the platform. You not only delivered what they asked you to deliver, but you delivered more than you were getting paid for. *Always do more than you're getting paid for and clients will keep paying you to get more.* How simple!

Fourth, tap into the power of reciprocity. If you give something away, the receiving person will feel the obligation to give back to you. That's why they have free food in supermarkets. I have hired many freelancers for *Finxter.com*, and some of them were really smart to get the gigs: When applying for a project, they would give

me something for free. For example, one project was “develop 100 Python puzzles,” and a freelancer applying for it gave me a couple of sample puzzles for free. I not only saw that he has the skills to do the work, but I also felt the strong urge to give back by hiring him. Even though I wanted to hire another freelancer who was an even better fit, I couldn’t help but hire the giver too because I didn’t want to miss out on giving back to him. Here’s a powerful mindset: *Give first! The receiving will take care of itself.*

Fifth, certificates go a long way in building trust with your clients. Today, many online courses offer course completion certificates. For example, you can download a personal course certificate for each completed course on the Finxter Computer Science Academy <https://academy.finxter.com/>.

Action steps:

- Create a new document titled *Business Systems*.
- Add a bullet list with things to do before completing each gig.
- Add bullet points of things to consider to increase the odds that clients will love you.

Expert Secret: Money Flows to Specialists

There are more approaches that will impact your success on these platforms. One is the specificity of your skill set—the more specific, the better and more trustworthy you will be perceived.

For example, many clients run their own Python Django application. Say, a client looks for an expert on Django authentication. If you'd sell your services by telling them, "*I can code any program you need Python*", chances are that they will not trust you—who knows everything in Python after all? But if you'd position yourself as the go-to expert for *Python Django Authentication*, the client will go for you because they need just that. For you, it's not only easier to find clients, it's also eas-

ier to master only one specific niche than to master a broad and general topic consisting of multiple specific niches. Communicating your circle of confidence—the things you are skilled and knowledgeable about—in an honest and authentic way signals expertise and professionalism to your clients.

What are the skills that the marketplace seeks? There are some foundations that any excellent Python freelancer must master. These are basic and complex data types, lambda functions, list comprehension, the complexity of data structure access, basic algorithms, keywords, and so on. Knowing about the foundations already qualifies you to do Python freelance jobs.

However, if you want to increase your earning potential, you need to specialize in more advanced knowledge areas. Examples are machine learning, data analysis, web scraping, or web development with Django or Flask. Each of these areas consists of subtopics like scikit-learn, regression analysis, and NumPy. In each of these specializations, you become more focused on this specific area, which automatically increases your value to the client. But an important observation is also that every specialization builds upon a solid foundation. So don't be lazy and skip the foundations!

Action steps:

- What's your specialization? Choose one—even if you're not yet knowledgeable about it. You will be soon!

Expert Secret: Read More Programming Books

Experts read books. Warren Buffett is said to have read 1000 pages of annual reports every day. Buffett and his partner Charlie Munger became billionaires by spending 80% of their day reading. Buffett was once asked how to become smarter, to which he responded:

“Read 500 pages like this every week. That’s how knowledge builds up, like compound interest.”

Knowledge compounds—a fundamental truth that all experts know. If you want to get highly paid in any space—say, as a freelance developer—you’ll need to become a highly sought-after person. And how do you achieve that? By reading a lot of programming books.

Programming books provide a maximum return on in-

vested time and dollars. Of course, you can find all the information on the web for free. But it'll take you a lot of time searching—and what you find tends to be low-quality information.

The opportunity costs of consuming low-quality information are huge (even if they never appear on your bank statement)! Instead of spending 20 hours reading software documentation, you could have read one book in 10 hours (or even two books in 20 hours). Not buying a book because it costs \$20 is like standing in line for two hours to get a free coffee: stupid.

Good books have the highest quality of information for a low, almost negligible price. If books are still too expensive for you, have a look at my list of 101+ free programming books.¹

Action steps:

- Spend at least 30 minutes every day reading a programming book. Use your habit tracker or simply add a small “pb” into your calendar every day where you manage to finish your 30-minute programming book reading session. Just read any book you can lay your hands on.

¹<https://blog.finxter.com/free-python-books/>

Expert Secret: Read More Business Books

The habit of reading more programming books will make you a coding expert. But as a freelance developer, you're as much a business person as you are a coder. You need to study business from the best. Fortunately, many successful business people have written great business books. Each of those books will make you a better business person.

Think about it this way: if every business book you read increases your yearly income by 2%, how many business books would you read? You'd read hundreds of business books—doubling your annual income every 35 books: *Knowledge Compounds*.

Action steps:

- Every day, spend at least 30 minutes reading a business book. Yes, with your 30-minute programming book reading that's 60 minutes reading every day.
- Add a small "bb" into your calendar every day where you manage to finish your 30-minute business book reading session.

Expert Secret: Seek Expert Advice

Here's a nice trick to boost your performance: seek out someone who's already been there and done that. How much faster would you learn to code if you had an expert coder who guides you through every step? How much faster would you learn to create a thriving business if Warren Buffett gave you valuable advice as you went along? This can save you years, even decades. We stand on the shoulders of giants. If you're feeling too small and insignificant, it's because you're NOT standing on the shoulders of giants.

Spending time, energy, and even money on expert advice is the success secret of the most successful people on earth: Warren Buffett, Bill Gates, Larry Page – they all spend millions of dollars and thousands of hours seeking expert counsel.

Action steps:

- Seek expert advice from freelance developers who've already mastered what you want to achieve.
- Consider joining my in-depth Python freelancer course.¹

¹<https://blog.finxter.com/become-python-freelancer-course/>

Expert Secret: A Simple Heuristic

Should you become your own boss, being self-employed? In this chapter, I'll give you a simple heuristic you can use as a decision framework.

Video 16: *Becoming a Self-Employed Programmer? A Simple Heuristic*



<https://python-freelancer.com>

Are you committed to becoming an above-average programmer? Think about salaries in a company. They often follow the Pareto principle: 20% of the people earn 80% of the salaries. 20% of employees create 80% of the

output in companies. The concrete parameters do not have to be 80/20; they could be 70/30, or even 95/5. In the programming space, it's often more extreme (like 95/5). For example, Bill Gates famously said:

“A great lathe operator commands several times the wage of an average lathe operator, but a great writer of software code is worth 10,000 times the price of an average software writer.”

There are many reasons for this. First, an expert programmer can solve a problem in a few seconds that would take an average coder days. Second, an expert programmer adds fewer bugs to the code base which reduces negative first-order consequences such as longer development cycles for the whole team, as well as negative second-order consequences such as damages produced by faulty code in the real world—think of the negative effects on a car company's brand due to a damaged self-driving car. Third, an expert programmer increases the social standards for all other coders, for example, by educating them and, thereby, raising the bar for the whole company.

In the programming space, some people are much more productive than other people. Above-average programmers create and, often, command more value than average programmers. Thus, if you are at least above average, you should consider becoming a full-time self-

employed coder to make sure you get paid proportionally to the value you create.

Note that this recommendation addresses only the first part of the question: should you become a full-time coder? So if you are committed to joining the top 20% of coders, you should become a full-time coder because then you will be able to become much more profitable than you would while working for a company.

If you don't become self-employed as a top 20% coder, the lion's share of your profits will be collected by the company you work for. If you are 10,000 times more productive than the average coder, you won't earn 10,000 times more as an employee. But if you are self-employed, you can! Just ask billionaire coders like Vitalik Buterin, Bill Gates, and Steve Wozniak.

But being a self-employed coder is not all about your coding productivity. Business success is first and foremost about marketing, client acquisition, effective communication with your clients (this alone can be more important than your coding skills), and your positioning in the marketplace.

And these crucial skills can only be learned if you are working as a self-employed coder.

Video 17: *Massive Action: A Foolproof Way to Find Clients as a Freelance Programmer*



<https://python-freelancer.com>

If you are an employee working for a big company, I bet you don't feel very confident taking the risk of becoming self-employed. And you'll never acquire these critical skills.

The only way to acquire these skills is to become self-employed, increase your hourly rate over time, and work on your business skills and coding skills at the same time. Read coding books, business books, books about selling, copywriting, and so on. Combining these skills will bring you much more substantial profits in the long term than working as an employee at a company.

Therefore, before you become full-time self-employed, just earn money for your company and do some Python freelancing on the side. And if you feel that you are skilled enough to match your salary income as a Python freelancer, you can safely decide to leave your job. You'll be very confident in the marketplace because you'll know that you have already provided value for your clients.

If what you earn is proportional to your invested time and above, say, \$100 per hour, go full-time. Don't lose any more time in this case because you already belong

to the top of self-employed coders.

In summary, this is the best advice that I can give: start for one or two years freelancing on the side and then take the big leap with confidence and a nice cushion.

If you are committed to becoming a top 20% coder and top 20% business person, then there is no doubt that you can and should become self-employed as a freelancer. But how can you become a top coder? By reading business and programming books. Do this every day for an hour, and you'll be virtually guaranteed to join the top 20% of coders and business people.

Why? Because the average person will read maybe 3-4 books a year (mostly fiction). So if you are reading non-fiction books such as programming books and business books, then there's no question that you'll become an above-average freelancer. In addition, you also have practical experience working part-time as a Python freelancer.

For some people, attending online courses is more efficient than reading books because of the content's multimodality and interactivity. Mastermind groups are also a great way of continually improving your skill level if you are the person who needs other people to push you.

Action steps:

- Commit now to lifelong learning. Use a habit

tracker to make sure you implement your commitment and read in a programming book and in a business book for an hour a day.

— 17 —

Expert Secret: About Timing

When will you start earning money? Just start now and figure out how to solve the problems as you go.

Video 18: *Freelance Programmer – When To Start Taking Projects?*



<https://python-freelancer.com>

You will be paid for your learning time. If you are just beginning with your Python career, you will earn less, but you will still earn something, get a lot of experience, and gain practical insight into what to learn and where your knowledge gaps are.

If you don't feel confident yet, you can master the Python basics first. You can already specialize in a Python topic. And to gain even more confidence, you can learn with toy projects. A little-known Python freelancing tip is to learn with archived freelancing projects. You can already gain practical experience and learn the type of projects for which clients have paid freelancers. Still—it's even better just to start doing real Python projects and then putting in all the effort to earn your five-star ratings. Don't worry about failing: you will fail from time to time. If you do, be honest with the client, accept your failure, and try again with a new client. But whatever you do, stay in the game and keep learning!

Action steps:

- Read one book teaching the Python basics.
- Get yourself a freelance gig (don't overthink it). Go!

Expert Secret: Practical Training Plan

My recommendation is that you use a personalized training plan which focuses on the practical skills you need.

First, decide how much time you can allocate to learning. Second, divide your learning time into two blocks using a 70%/30% allocation. Third, use 70% of your *learning* time to implement a practical code project with real-world impact. The rest of the time (30%), invest in solving Python puzzles (see next Expert Secret), work through Python courses, or read Python books.

For example, if you can spend 100 minutes per day learning to code, spend 70 of those minutes working on a practical project! What is a practical code project? A practical Python project can be archived freelancing Python projects that challenge you to go to higher levels. Or it could be one of your dream projects—this is

even better because it keeps you highly motivated and engaged. The key is not to stop working on these until you have successfully finished them and have created a viable product.

You can see that this is a highly practical approach. The reason is that only practical motivation can push you to mastery in any subject. You need to understand your knowledge gap and see what you don't know before stuffing things into your brain. Any expert will tell you that practice-first is how to learn Python fast.

Video 19: *You Are Stuck in Coding! ... And Why Nobody Tells You*



<https://python-freelancer.com>

Action steps:

- If you haven't already, watch the free Finxter freelancer webinar at <https://blog.finxter.com/webinar-freelancer/> because freelancing provides you with a never-ending stream of practical code projects.

- Create a training plan to track your learning time and use it to track your learning and practice time every day.

Expert Secret: Solve Python Puzzles

For the theoretical part, I recommend solving Python puzzles as one of your primary methods of improvement. Python puzzles are a powerful tool for becoming more proficient in reading and understanding Python source code (see Figure 19.1).

What's a Python puzzle? *A Python puzzle is an educational snippet of Python source code that teaches a single computer science concept by activating the learner's curiosity and involving them in the learning process.*

The idea is that you solve code puzzles that start out simple but become more and more complex as you progress in your learning journey. In essence, you play Python interpreter and compute the output of a code snippet in your head. Then you check whether you were right with your guess using feedback and explanations to adapt



```
meal_1 = "Meat"
meal_2 = "Flaxseeds"
meal_3 = "Marshmallows"

healthyFoods = ["Kale",
                "Apple",
                "Strawberry",
                "Banana",
                "Flaxseeds"]

def isHealthy(food):
    return food in healthyFoods

m_1 = isHealthy(meal_1)
m_2 = isHealthy(meal_2)
m_3 = isHealthy(meal_3)

print((not m_1 or m_2) and
      (meal_2 is "Flaxseeds") and
      isHealthy("Kale"))
```

Figure 19.1: Example of code puzzle. Find the solution at <https://app.finxter.com/learn/computer/science/408>.

and improve your coding skills over time. To make this idea a reality, I developed the online coding academy *Finxter.com*. The Python puzzles range from easy to complex—each puzzle will push your theoretical and

practical coding skills one step further. The puzzle-based learning method is very effective, proven by tens of thousands of online students. It incorporates elements of immediate feedback, conditioning, repetition, trial and error, and active learning techniques for maximum learning efficiency.

Here is an example of a code puzzle:

```
# Create the list
friends = ["Alice", "Bob", "Ann"]

# Remove the final element
friends.pop()

# Remove and return the first element
print(friends.pop(0))
```

What's the output of this code snippet?

Action steps:

- Check your correct solution to this puzzle here:
- <https://app.finxter.com/learn/computer/science/563>
- Solve 10 more puzzles on the app, it's addictive!

Expert Secret: Learn with Cheats

In your theoretical learning time, feel free to invest 10-20 hours in refreshing your basic Python skills before anything else. This is not much of a time commitment—after all, you are learning a high-income skill. You can learn a lot in 20 hours if you do it right. The key is to learn probabilistically by mastering important sub-skills first.

So what's the best way to learn probabilistically? Simple: use the 80/20 principle and learn 80% of the necessary Python skills to get started in 20% of the time via Python cheat sheets. I have summarized the five best Python cheat sheets at <https://blog.finxter.com/python-cheat-sheets/>. Download the cheat sheets and spend your first 20 hours learning them thoroughly. Or even better: print them and post them to your office

wall.

Video 20: *How to Learn Python Faster?*



<https://python-freelancer.com>

Action steps:

- Download the 11 Python cheat sheets.
- Consult them daily until you know them by heart.

Expert Secret: Confidence

How can you gain confidence that will increase your value to the marketplace? If you're like most other Finxter students, you don't want to offer your services before you don't feel 100% confident about your skills. Unfortunately, this moment never arrives. I have seen hundreds of advanced coders who are still not confident in selling their services. They cannot overcome their self-woven system of limiting beliefs and mental barriers.

Can I tell you a harsh truth? You're unlikely to join the top 1% of the Python coders. Unfortunate, but a hard statistical fact. Most people can join the top 10% of the coders — many of them earn much more than \$50 per hour — if they're ambitious and keep learning Python daily following the ideas outlined in the previous chapters. But joining the top 1% of coders is much

tougher than that.

Never mind! Your services will still be valuable to clients who either have less programming skills (there are plenty of them) or little time (a big part of the rest). Most clients are happy to outsource the complex coding work to focus on their key result areas.

Regardless of your skill level, the variety of Python projects is huge. There are simple projects for \$10 that an experienced coder can solve in 5 minutes. And there are complex projects that take months and promise you large payments of \$100 to \$1000 after completing each milestone.

You can be reassured that you will find projects within your skill level.

Action steps:

- Take your time to browse all the archived Python freelance projects.¹
- Select 3 projects that you think you can solve in the price range (\$10-\$50). Better yet: browse real projects and apply to them even though you don't feel ready yet!

¹See the book companion page resources provided at <https://python-freelancer.com/>

- Write down your dream niche within the programming space — for example, Blockchain development, Python automation, web scraping, machine learning, data visualization — keeping the projects in mind that you just selected: data science, web scraping, application development, or scripting.

Expert Secret: Learn While You Earn

Most developers know that they should never do premature code optimizations. But they do premature *skill optimization*—that is, learning without immediate practical application—all the time. Don't do that. Instead, you should laser-focus your time to learn the essential skills you need to finish the current code project and delight your users and clients with the highest priority.

Start with real projects as soon as possible. Don't wait too long—even if you are a beginner programmer. Set aside 10 minutes per day to watch out for interesting freelancing projects. If you do so, you will learn the practical Python problems that clients will need solved. You will learn about the patterns of Python problems “in the wild.” This knowledge will guide you in your efforts to become more valuable to the marketplace. In

contrast to millions of other aspiring coders, you will develop a practical Python skill set tailored to your interest level. Try to establish your brand as soon as possible. Learn with real clients. Focus on mastering the highly practical skills by working on real projects.

Action steps:

- Browse current freelancing Python projects daily on websites such as Upwork or Fiverr.
- Find projects you like within your difficulty level. Take the first project you think you can solve.
- Invest all your time and effort into cracking this project. Make sure to track the time you need and calculate your hourly income.
- Learn on the fly, as you solve real projects. It's a sure way to improve and build a relationship with your clients—and make money in the process.

Expert Secret: Psychology Tricks

My company Finxter relies on many freelancers to improve the overall student experience. Naturally, I have collected a lot of experience using freelancer services as a client. I soon realized that a similar pattern emerged every time I posted a new project description. A few hours after posting the description, several freelancers applied. The competition was fierce. But within minutes, I had subconsciously chosen an inner circle of high-potential candidates. In the vast majority of cases, I didn't select a freelancer who could not immediately pass the entry barriers posed by my subconscious mind.

You want to be in that inner circle of potential freelancers. To get there, you must appeal to the subconscious mind. The following factors will give you a psychological advantage when competing for a freelance

job.

Use the power of reciprocity. Reciprocity is the social impulse to reply to a positive stimulus with a positive reaction, thereby rewarding other people’s positive actions.

This principle is powerful. It works on me—and it will work on other clients. For example, I published a project to check the Python code of my recent Python programming book. The project description stated that I intended to edit the code to make it more Pythonic. Several freelancers applied immediately for the job. Most of them went into “competition-mode”, bragging about their credentials. They tried to convince me that they were the perfect fit for this project. I selected a few candidates but was not 100% sure about any of them.

Towards the end of the application phase, a new freelancer registered interest with an unusual application. Instead of talking about his credentials, he focused on the project itself. He dived right into the project and submitted annotated and corrected Python code snippets—improving those I provided as sample files. He gave them to me for free. Of course, I knew that he purposely used the reciprocity rule to get the job. Yet, I was immediately hooked and felt a strong obligation to reward him for his work—and I gave him the job.

This is the power of the reciprocity rule.

Don't hide your titles and credentials. They still work. When you apply for a job, and you have the title “Prof.,” “Ph.D.,” “BSc,” or “MSc” in a relevant area, you will gain immediate credibility. In most cases, it will set you apart from the other freelancers without strong credentials or titles.

Note that credentials are not limited to the academic world. You should also highlight your practical achievements, such as your websites, shiny projects, or certificates. Be creative.

Invest time in your profile picture. You wouldn't believe the powerful impact of your profile picture on your chances of getting the job. Many coders don't focus too much on appearance. Don't do this. Smile, dress professionally, use a natural image background.

Don't compete on price. Forget about it. Competing on price is a race towards zero. You can not win. There is always a cheaper freelancer, and some of them WILL apply for the same projects. It's true—some clients look for the cheapest freelancer who barely finishes the task. But most clients will choose high quality and predictability over price. What would you do if you were a business owner who works 60 hour weeks to push his website? You love your baby and don't want a cheap freelancer to mess around with it. A freelancer that offers a service at a very cheap rate is also per-

ceived to produce cheap quality. After all, you wouldn't be cheap if you were good.

So what is the value of an hour of your work? Multiply this number with 1.5 for two reasons: (1) You tend to underestimate your value to the marketplace, and (2) it's a good practice to push yourself towards higher income (that's what you want, isn't it?). Now you have your number. Don't work for an hourly rate below that number! And keep pushing it - the sky is the limit!

Video 21: *Programmer Procrastination: Best Hack From Behavioral Psychology*



<https://python-freelancer.com>

Action steps:

- Collect Python certificates. For example, use our web app Finxter.com to certify your Python skill level. You can also find certificates on Udemy, Udacity, and RealPython.
- Take an awesome profile picture.
- Give something to each potential client. For example, invest some time creating a prototype so-

lution. This will greatly improve your acceptance rate and ultimately save a lot of time!

Expert Secret: Contribute to Open-Source

Clients want to see that you're the right fit for them. They want to minimize transaction risk: if they book a freelance developer, they spend time and money. Above everything else, they fear losing their invested time and money. This is a psychological effect that has proven to be very robust in countless scientific studies, called loss aversion. Most people would rather avoid losing than get the chance to win.

A powerful way of mitigating loss aversion is to contribute to open-source projects and showcase those projects to your clients. If clients browse your projects, they'll feel that you are a trustworthy coder who gives a lot to the community. Naturally, they'll feel more attracted to

you and see you in a positive light.

Video 22: *Don't Think in Terms of Resumes as a Coder – Do This Instead!*



<https://python-freelancer.com>

Moreover, working on open-source projects connects you to like-minded coders. Many will refer freelance developing gigs your way—make sure to talk to and help a lot of people in the open-source community. From time to time, drop that you're a freelance developer and watch how more business will come your way.

Last but not least, you'll boost your coding skills. Working on practical code projects is the number one strategy to improve your skills. Chapter 18 outlines a practical-first strategy for learning how to code (70% practice projects, 30% theory).

Action steps:

- Which open-source projects would you love to work on?

Expert Secret: Non-Programming Skills

Do you work in the software development industry? Big news: programming expertise is an important skill set that will help you accelerate your career. For example, one of the most satisfying and highest-paying jobs is *DevOps specialist*. But DevOps specialists need to have a high level of coding expertise, gained through working on practical projects in different developer teams.

However, programming is not the most important skill needed to become wildly successful in the field, especially if you are working as a freelance developer online, creating your coding side-business, or looking for other ways to increase your income. These two skills are more important than programming expertise to earn six figures online:

Sales and Marketing. Without the shadow of a doubt,

creating good sales and marketing systems is the most important skill you can have as a self-employed freelance developer online. You can even have *only* this skill and become wildly successful these days. Many online marketers do just that. They create their community of like-minded people, offer more value, and are crushing it.

As a programmer, you need to know how to market and position yourself.

- How would you like to be perceived?
- Where should you focus?
- How can you create a landing page that converts?
- How can you start on freelancing platforms such as Upwork or Fiverr?

Communication. Again, a no-brainer. But many programmers think it's only about the quality of code they submit. No! It's about being in constant communication with your client. Ask them if you don't understand the specifications. Figure out exactly what they want. You may need to dive deep into this. For example, a client may tell you they need a beautiful GUI for their back-end, but they need to improve their back-end functionality.

You need to keep communicating, be responsive, positive, upbeat. This will ensure that you get good ratings and recommendations on freelancing platforms—and that your pipeline of clients stays full.

Video 23: *How to Get Clients as a Python Freelancer? A Guide For Noobs*



<https://python-freelancer.com>

If you want to improve your communication skills, read the excellent book from Dale Carnegie: *How To Win Friends And Influence People*. Also, sharpen your marketing skills. You can find free marketing courses at Hubspotacademy.com.

Action steps:

- What is one thing you can do today to improve your sales and marketing systems?
- How can you become a better coder today?
- Send one of your clients a valuable email (or, even better, call them) now.

Part III

Strategy: \$2000 to \$4000

Expert Secret: Freelancer Algorithm

Becoming a freelance coder is both easy and hard. It's easy because everybody can do it within a day or so. And it's hard because most people fail because they don't know the right way of approaching this.

Maybe you are interested in coding, or maybe you are already very good at coding.

But see, you've already approached this endeavor from the wrong perspective. As a freelance coder, you are first a business person and second a coder. It's not the other way round. Coding is not your number one skill as a freelancer. Many coders who start on platforms such as Upwork or Fiverr believe that they need to sharpen their coding skills before they can become successful using these platforms. Nothing could be further from the truth!

A much better strategy is to think like a business person: *what problems do people have and how can I solve them?* With this in mind, you can even approach people and ask them whether they would be interested in solving their problems. This skill is called *selling*—and it’s the lifeblood of any business.

So you start solving small problems, grow your skills as you solve them, and then attack larger and larger problems for your clients. As you solve their problems, they will be happy and give you a positive rating. You’ll find it much easier to get more and more clients. At some point, not too far in the future, you’ll have much more on your plate than you can solve alone. Clients are contacting you and asking you to solve their problems. And they’ll refer you to other clients with similar problems. Heck, even the freelancing platform will propose you as a freelancer to solve their clients’ problems since they want to satisfy them. This is a good sign, and it means that you have been quite successful in solving clients’ problems.

As you’re a coder, I’ve compiled the great freelancing strategy in pseudocode for you:

```
goal_hourly_rate = input("What's your goal
↪ hourly rate?")
current_hourly_rate = 0
while current_hourly_rate < goal_hourly_rate:
```

```
problems = make_list_of_problems()
clients = make_list_of_clients()
clients[0].contact("Can I solve your
↪ problem: " + problems[0] + "?")
solve_problem(problems[0])
```

This is a simple algorithm, but it works. Become aware of the problems in your industry—guess if you don't know them exactly. Then, contact potential clients outside of the freelancing platform. You can use the freelancing platform to find the contact info of the clients. Ask them whether they would be willing to hire you if you solve one problem. If not, ask them what problems they have. Over time, you'll get pretty good experience in determining the problems of clients.

Then you attack the problem. No worries if it takes you forever to solve it. That's just not relevant. You are on a path of continuous improvement, and every follow-up project will become easier because of your experience, while finding follow-up clients will become easier too. It's hard at the beginning and very easy later—if you stay in the game long enough.

Video 24: *Why Most Coders Fail Riding the Curve of Continuous Improvement*



`https://python-freelancer.com`

Action steps:

- Print the freelancer success algorithm.
- Run through the first loop body.
- If you haven't already, watch our free video about the state-of-the-art in Python freelancing: `https://blog.finxter.com/webinar-freelancer/`.

Expert Secret: Leverage Network Effects

The internet follows a universal law: “The winner takes it all.” The rich get richer, and popular people get even more popular. If you are already winning on these platforms, you will win even more. People tend to reinforce the decisions of their peers. If all of them gave you five stars, most clients would simply default to giving you five stars as well. The network effect is a well-researched phenomenon in all kinds of networks like social networks, the web, and freelancer-rating networks.

There are two basic tactics that you can use to leverage this information to earn more money and increase your value to the marketplace.

First, focus on your first jobs—see them as investments

into your future. Even if you did them for free (I'm not advocating this), they will be profitable by attracting better jobs and clients.

Video 25: *Why You Should Offer Your Services For Free as a New Freelance Developer?*



<https://python-freelancer.com>

Second, you should prefer many small jobs over a few large jobs, so that you'll gain credibility faster. Many people subconsciously have the simple heuristic: more jobs + more ratings = better freelancer.

Action steps:

- Get a small \$20 gig on a freelancing platform.
- Overdeliver.

Expert Secret: Return on Invested Effort

May I share with you one of my business secrets?

You can transform your business by calculating the Return On Investment (ROI) of all activities you're doing. For instance, here's how this would look like for creating Facebook ads:

Investment: Writing an ad may take you one hour. You may price the value of one hour of your time at \$100. So writing an ad costs you \$100.

Return: Say your average ad brings in 1000 visitors per month. About 1% will buy your product with a profit of \$10 (after deducting the costs per click). So the return is $1000 \text{ visitors} * 1\% * \$10 = \$100$ per month.

So, writing an ad results in a 100% return on investment month after month. Based on your estimations, you decide that spending time writing ads is a great idea for your business.

I started my career as a Python freelancer, and the experience I got and the skills I learned were priceless. Before I purchase anything, like an educational program, I calculate the expected return on investment.

Now, I want you to do one thing: calculate the expected ROI of taking the freelancer course¹ by assuming it increases your hourly rate by only 10% in the first year. It will increase your hourly rate much more, but this is a good conservative starting point.

You'll probably realize that investing in yourself is the most profitable use of your time and money, especially if you're still young (below 30). That is if the course can keep its promise!²

Action steps:

- Start your capital allocation habit today—your most valuable capital is your time!

¹<https://blog.finxter.com/become-python-freelancer-course/>

²The good thing is that there's a strong 60 (!) day money-back guarantee. If the course doesn't deliver, you'll get your money back.

- Calculate the ROI based on your rough estimates of an activity every day.
- This may sound fuzzy (and it is), but it's the highest value activity I have ever started in my business. It will transform your business, too.

Expert Secret: Don't Compete on Price

Today, there's global competition in the coding market. If you create your coding business online, you compete for business with coders all around the world. And there are great coders in India, Pakistan, and other countries who have a significant advantage in pricing flexibility—it's much easier to reduce prices if you are living in a non-expensive area. If you're living in Silicon Valley, you need to charge premium prices just to be able to afford basic housing.

Video 26: *Where Should You Live as a Freelance Developer?*



<https://python-freelancer.com>

Before you conclude that you don't have a chance to pursue this freelancer thing, stay with me. If the world were that easy, America would already be out of business. But America's freelance developers thrive.

Now, how can you possibly compete in the global economy as a coder in a developed nation?

The answer is simple and straightforward: don't compete for price. You need to provide superior perceived value. That's it. If you compete for perceived value, you can easily earn six figures in the US.

Video 27: *Don't Compete For Price as a Self-Employed Developer!*



<https://python-freelancer.com>

How can you improve your perceived value? Establish predictability through testimonials and positive ratings. Communicate well, be responsive, be positive. Become

a marketing and sales pro (e.g., create your own community).

Action steps:

- Ask every client for a testimonial.
- Place the previous step to your gig checklist.

Expert Secret: Free Float Your Learning

But can it be a good idea to offer your services for free, considering that Python freelancers earn \$51 per hour on average? Yes, it can! Note that this doesn't conflict with the previous expert secret.

You can increase your Python skill level, but you don't have to polish your salesman skills from day one. You don't compete on price; you compete on learning speed. Price is only a tool to accelerate your learning speed.

It's a great way to increase your confidence as a Python freelancer. And confidence is a crucial skill to working successfully on freelancing platforms.

Building this confidence is one of the most important things you can do to become a successful freelancer be-

cause your clients will sense whether you feel confident enough to deliver a lot of value. Without this, it'll be hard to find jobs. But without jobs, you cannot build confidence. Many students remain stuck in this negative spiral and lose confidence over time.

A great way to break out is to offer your services for free. In any case, you won't get rich with your first few gigs—the bulk of the money is earned later. The first gigs are for the experience, the testimonials, and the success rate statistics on platforms such as Upwork.

You cannot offer your services for free on the Upwork platform (it's against their terms). So what can you do?

An alternative is to contact people who rely on freelancers to grow their businesses. One great example is a programming blog. Blog owners always need new articles, and many of them hire freelance writers (who are also skilled in programming) to create articles for them. Even if you can't land a paid job, you can contact any blog owner and offer it for free. Just tell them that you are looking for testimonials and experience.

Action steps:

- Before getting your first gig, offer your services for free in exchange for a testimonial.
- An even better idea is to offer clients the following deal: finish the gig for free. Seeing the result, they

can pay you what they believe is fair compensation. This leverages several psychological effects that are well-known to be beneficial for your income.

— 31 —

Expert Secret: Give More Than You Take

As a self-employed freelance developer, you're as much a business owner as you are a coder. You need to master both skills— business and coding—to thrive in your industry.

“Giving more value than you take in payment” is the first rule of business. While it sounds simple, most struggling business owners violate it. And this is the reason they struggle in the first place. They forget that on the other end of the transaction is a human being with a pain point, a problem, or a desire. And the only reason you're here is to remove the pain, solve the problem, or satisfy the desire.

If you pay two bucks for a coffee, it's because you value

the warm, tasty coffee in a comfortable environment, where you can relax and read a book, much more than you value two bucks. If you'd value your money and the coffee equally, you wouldn't give up your two bucks (behavioral psychologists call this "loss aversion").

See the pattern? The sole reason your business exists is to solve the problems of your customers. You simply cannot survive if you don't know which problem you solve.

Formula: If your client asks you to do X, do X...and a little more.

With this mindset, you're going to crush it in business. The power of reciprocity will kick in, and your client will always want to hire you for the next thing. It's far easier to convince an existing client to buy again than to convince a new lead to buy the first time. Every great business leverages this powerful law: give more than you take.

Action steps:

- Before you submit work to your client, ask yourself: have you given all you have? Can you give some more?
- Add this to your checklist to ensure that you'll do this after each gig.

Expert Secret: Eat Complexity

Fortunately, you're in the coding business. And coding is a mess. It's difficult, complicated, error-prone, and only a small minority of people have the patience and time to learn it. There's a massive barrier of entry. You need to spend a lot of time staring at your computer and searching StackOverflow to learn how to solve bugs in the code.

Video 28: *How to Create a Barrier of Entry
Running a Software Business*



<https://python-freelancer.com>

What does this tell you? It tells you that your client needs you more than you need him. This is a great position to be in. If you can solve and alleviate the complexity in your client's life, you're doing him an essential service.

Either you or your client has to handle the complexity. If you're paid to set up a server system, it's because your client doesn't want to burden the complexity. He hires you to free himself from the complexity. Great business owners such as Jeff Bezos, the founder of Amazon, have mastered this rule of eating their customers' complexity. Jeff Bezos pushed towards one-click payments because it makes purchasing easier. Jeff Bezos created the Kindle to simplify the whole book consumption process. You don't have to order a book and wait forever. With one click, you can now read almost any book in the world.

Great freelance developers empathize with their clients or just ask them to learn about their complexities. Then they take on their clients' complexities.

These are the two most important rules for freelance developer success. They don't sound fancy, and they're not easy to do—they are not business hacks to make a quick buck. But they will make you more money than everything else you've learned.

Action steps:

- Make a big list of things that are complicated for your customers.
- How can you eat your clients' complexity?

Expert Secret: Perform From Your Strengths

This rule was formulated by Peter Drucker, the famous management consultant. Time is limited, and you can either improve your weaknesses or your strengths. Say you're a great writer, but you don't enjoy being in front of a camera. On a given day, you can write your new book or create your new video course. What do you think will create more value for society?

Many people are great in front of the camera, but they cannot write. Why on earth should everyone do what he is naturally bad at? More value can be created if everyone performs from their strengths. As a natural writer, write. As a natural coder, code. As a natural singer, sing. This way, society becomes much more efficient,

and value creation explodes.

For you, this means that you need to know your strengths. What are you naturally good at? Don't read on until you've answered this question.

Make a list!

- What have you been great at in school?
- Which YouTube videos do you watch a lot?
- What do you keep talking about?
- What do you love to do?
- What do you spend money on? Check your income statements.

These questions can give you a hint at where your strengths lie. Use this information to your advantage! Let someone else compensate for your weaknesses. That's how a society should operate anyway.

Action steps:

- Make a list of your strengths by answering the previous questions.
- How can you move towards your strengths in your professional life?

Expert Secret: Be a Specialist

Would you pay more to a brain surgeon than your generalist doctor to perform critical brain surgery? Of course, you would! Specialists can demand higher rates because their skills are highly developed in one specific area. Due to their specialized knowledge, there's little competition, and they often have a dominant market position.

You have 24 hours each day. You can either invest, say, 10 hours every day in data science. Or, you can invest 1 hour in data science, 1 hour in blogging, 1 hour in writing a book, 1 hour in learning guitar, 1 hour in programming, 1 hour in singing, 1 hour in preparing coffee, 1 hour in studying cryptocurrencies, 1 hour in writing a compiler, and 1 hour in politics.

If you do the former, you become a specialist. If you do the latter, you become a generalist.

As a freelance developer, you shouldn't specialize in a specific technology but on a particular problem. For example, don't focus on Python's TensorFlow library for deep learning, focus on machine learning concepts. Don't focus on NumPy, focus on data science. Don't focus on Django, focus on web development. While the technologies may change over time, your specialized knowledge in a particular problem area will remain valuable.

Video 29: *Focus on One Thing. But What's the Thing?*



<https://python-freelancer.com>

Action steps:

- Which problems will you solve for clients?
- Which technologies should you learn first?
- Write a letter template to your dream client—how can you help them solve their problems?

Expert Secret: Be Hyper-Responsive

Communication is a crucial skill for freelance developers. If you don't know the exact problems and struggles of your clients, you cannot help them. A lack of communication will regularly lead to hours and hours of wasted time. Let's say your client wants a website hosted. Then, you host the website. But it turns out that the hosting provider you selected is too expensive for the client. All the work is lost.

You need to communicate with the client at every stage of the process. Give them a daily update of what you're currently doing and what you have accomplished. Ask them many questions. This will help you figure out the concrete gap where you can give the most value. And it'll ensure that the client sees how much time and effort you've invested.

Of course, this doesn't mean that you should spam the client with minor questions—you have to find a good balance. But if you're in doubt, it's far better to err on the side of too much communication. And, of course, a freelance developer who is hyper-responsive and answering emails within hours (not days) will make a very good impression, even if the code he delivers is not perfect. The freelancer-client relationship lives and dies on great communication.

Action steps:

- If you haven't already, get the great book *How To Win Friends And Influence People* by Dale Carnegie to learn how to communicate with people. Otherwise, take a break—you earned it.

Expert Secret: Be Positive and Upbeat

Have a can-do attitude with a bias towards action. This is almost always superior to an attitude where you see constraints and limitations everywhere. If the client asks you if you can solve one of his problems, be optimistic and positive, and figure out how to do it. If you can't do it, figure out how to solve a slightly easier problem—and repeat until you're able to solve the original problem.

Clients love freelance developers who are positive and upbeat. If you're their daily positive contact, you have their subconscious mind on your site because it'll always push your client to hire you again. Without the client even being aware of this, he will be attracted to you, which is a powerful client retention tool. Plus, it's a lot more fun that way.

Action steps:

- Use the Grammarly plugin in your browser. It also gives you “clarity” and “engagement” scores to start writing more engaging messages to your clients! A spellchecker will immediately improve your average writing quality.

Expert Secret: Know Your Hourly Rate

Everything that gets tracked gets improved. As a business owner, the most important metric is how much you earn. This is as true for freelance developers as it is for Fortune 500 CEOs.

Therefore, you should continuously track and increase your hourly rate. This challenges you to keep increasing your value to the marketplace, communicate your value better, and learn more by reading books or industry periodicals.

How do you calculate your hourly rate? There's nothing simpler—just start tracking your time. Divide your average monthly income by 30 to get your daily income. Divide it by your average number of hours worked to get your hourly rate. This is your base rate. If you charge clients, you should at least double this rate—even bet-

ter, more than that. The reason is that not all working hours will be tracked: you also have to work on your taxes, learn and improve, do marketing, etc.

The average hourly rate of a Python freelance developer is about \$51. This is what the average developer earns, and you should at least target this rate. However, it's relatively easy to become better than average—especially if you use the tricks outlined in this tutorial.

Action steps:

- Answer one simple question now: What's your hourly rate?
- Copy my hourly rate template here: <https://bit.ly/hourly-rate-template>

Expert Secret: Increase Your Hourly Rate

The next step after knowing your hourly rate is to increase it. Earning more money is simple: charge more. If you don't increase your hourly rate, you won't earn more money. But if you have higher hourly rates, clients will value your services more. Clients are humans, and, as a result, they tend to reinforce their decisions. If they pay more money, they will focus much more on the value you bring to the table. The Halo effect in behavioral psychology is defined as the tendency to generalize a positive impression to unrelated areas. Charging more money may impress a client, causing them to look for these positive impressions in your work. Deliver quality code quickly—it's mostly a matter of focused effort on your part.

Charging more money will increase your confidence. If clients pay you more money, you feel valued and motivated to keep learning and keep improving. With more money, you can also reinvest more money in your business to enhance your appearance and marketing. With more money, you can solve most business problems quickly.

Action steps:

- By how much (percent) will you increase your hourly rate every month?
- How much time will you spend each day reading books in your field to justify charging more money?

Part IV

Systems: \$4000 to
\$12000

Expert Secret: Use Systems and Templates

All thriving businesses use systems to create consistency, predictability, and scale. Without systems, you're the slave, and your business is the master. With systems, you're the master, and your business system is your slave. Controlling your systems means that you can create a predictable stream of leads, customers, and cash flow. If you're the one business owner in your industry who masters systems, you'll crush your competition.

As a freelance developer, you can create systems in many different areas. Here are some ideas where systems can help you do more with less:

Write a template email response for similar emails you receive from your clients. If clients ask you for

your credentials, prepare a killer email response once, copy it into a Google Docs file, and reuse it multiple times. Can you see the power of email templates? You only do the work once but leverage your work again and again and again. Over time, you'll polish your email templates so that they become better and better. This way, you'll reduce the average time spent on each email, but you'll also increase the average quality of your responses. You'll work less and convert more leads to clients.

Write a template to apply for new freelancing gigs. However, it's smart to modify the first sentence of the template to fit the individual freelancing gig. I know it sounds extremely simple, but working on your lead generation template is a highly effective and leveraged use of your time. Increasing the conversion rate of your gig application template from 10% to 20% instantly doubles your sales! I've written a blog article to help you write effective gig application templates.

Create ads to sell your service. Creating and improving ads for your freelancing service is time well spent. Why? Because every ad is a mini system that sells your skills on the marketplace—even while you sleep.

Create content such as YouTube videos or blog posts in your field. Each blog post and each video lives in the web decoupled from your time. Content sys-

tems exist independently in cyberspace—and they sell your services 24/7. A nice benefit of content systems is that they instantly position you as an educator rather than a marketer. Imagine you're a business owner, and you want to hire a freelance developer to write an Android application. Who would you employ as a freelance developer —Alice, who writes blog articles about developing Android apps, or Bob, who does nothing of the sort?

Create people systems. A people system can be an employee who helps you with your freelance developer business. Here are some examples: your accountant, your lawyer, subcontractors, or affiliate marketers who sell your services for a 10% fee. Every person who helps you in any way is a stakeholder of your small freelancing business. The more stakeholders there are in your business, the more firepower it has.

Create referral systems as outlined before. Of course, there are many other soft and hard systems. As a rule of thumb, every action you perform on a repeat basis is a good starting point to create a mini system. Work as much on your freelance developer business as you work in it. Don't just write code for clients—spend the same amount of time improving your marketing systems!

Action steps:

- Track your systems in a new document in the form:
trigger -> checklist of what to do on that trigger.
- Template: https://docs.google.com/document/d/1r2r_oJcxm4ZoNmCsndv96EdJ_KuvNMaQd1Ai4cb_-cU/edit?usp=sharing

Expert Secret: Create a Client List

Depending on what study you read, it's between 5 and 25 times more expensive to acquire a new client than to convince an existing or previous client to buy again. Every great business has high client retention rates.

The first step towards having high client retention is to deliver more in value than you take in payment—the golden rule of business success. The second step is to maintain a client or customer list. Nurture your list. Send them regular emails with an email automation tool such as Mailchimp. Send them valuable resources, tools, information products, or greetings. Be good to your customers, and they'll be good to you. You should center your whole business around the simple idea of generating a list of leads, converting those leads into customers, and converting one-time customers into re-

peat customers. In each step, you move them further down the funnel.

Action steps:

- Create a free account for email marketing software.
- Start building your list of clients.

Expert Secret: Create an Ad Funnel

Funnels are potent mechanisms to deliver value for clients and create a predictable flow of business. In the end, the company with an optimized value delivery funnel will beat the company that performs random acts of marketing.

What's a funnel? A funnel is a powerful machine that transforms random people into leads, then into customers, and finally into repeat customers. Make no mistake: working on your funnel is one of the most effective things you can do for your business. It's one of those \$10,000 per hour activities that can make or break your business.

This tip is about advertising. If you read the Google annual reports, you'll see that the costs per click keep decreasing year over year. One reason is the progress

made in artificial intelligence that helps deliver more relevant ads. Advertising tends to become more profitable as a result. Businesses that don't use advertisements to acquire leads miss out significantly. If you've got a high-converting funnel, you can easily make back your advertisement costs—in many cases, you'll double, triple, even quadruple your ad spend.

Use ads to create a predictable source of leads for your business.

Action steps:

- Sketch your ad funnel: what's your dream client, and what are the steps they take in your funnel?
- Create a Google, Facebook, LinkedIn ad account, and start serving ads. Use a small daily amount (e.g., \$5) and never stop paying this. This forces you to improve your ad copy over time if you don't want to throw \$150 monthly ad spend out the window.

Expert Secret: One Lead Per Day

If you've got a funnel, you know how important it is to move leads further down the funnel. It's the lifeblood of your business.

How do you acquire new leads? Just contact people. Don't shy away from contacting people manually. Write emails to website owners and offer them a free "strategy session" (see next tip). In effect, you acquire new leads manually.

You may ask: how can this pay off even though it's not scalable? The answer is simple: It doesn't have to be scalable. You just need to get some momentum. As soon as you've convinced a random person to become a lead, you can start to give them value and move them further down the funnel until they become customers. Happy customers will talk, and that'll help you get more

customers. This cycle will repeat and strengthen over time.

Action steps:

- How can you find potential leads?
- What can you offer them to join your list of leads?
- How can you move them further down the funnel so that they ultimately become customers?

Expert Secret: Leverage Testimonial Videos

It sounds simple, but only one in five freelancers (at most) is doing it. Why? I don't know — maybe people are too shy asking their clients for video testimonials?

I only know that the combination video + testimonial converts like crazy.

Video engages people like nothing else on the web. Think about it: video contains all elements of a strong message— audio, image, movement, and text. It's far superior in terms of engagement and conversion rates compared to text only.

On the other hand, people trust other people. They don't trust you and your business because they know that you want to sell them stuff. So your clients know

that you'll present yourself from your best side. Essentially, you can tell them whatever you want, but there's no way for your clients to verify it.

While this sounds like a problem for your clients, it's much more your problem. If clients are not 100% sure that they can trust the quality and professionalism of your services, they are unlikely to buy. The only way of overcoming this is to leverage testimonials. Testimonials—and, to a larger extent, celebrity endorsements—are scientifically proven to increase customer conversion and trust.

Now combine the engagement booster of video with the trust of testimonials, and you get one of the most powerful marketing weapons in existence.

How can you, as a software developer, use this strategy to convert more clients? Simple: ask every single client you work for to send you a video testimonial. Nowadays, many people are comfortable recording selfie videos, so the barrier is low, and many clients will be happy to help you out. I observed compliance rates of 40% and higher when working as a freelance developer. That is, if you give value first and solve the gig satisfactorily.

But how do you get to the first testimonial video if you don't have any clients? Simple: work for free. Seriously. The value of your first few freelancing gigs will be mainly in the experience, the testimonials, and the

client base. The value is not in monetary compensation. So it's a reasonable choice to skip the monetary reward altogether for the first few freelancing jobs. However, after you get your first video testimonials, you should radically ramp up your hourly rate to avoid getting stuck in low-paying jobs.

Testimonial Video

How Adam Earns \$5000 per Month as a Python Freelancer on Upwork



https://youtu.be/IrzUr__AUio

Action steps:

- Ask *each* client to give you a video testimonial.
- Put this on your gig checklist of things to do after each gig to streamline and optimize your value

delivery system.

- Create a video testimonial page on your own website.

Expert Secret: Get the Referral Engine Rolling

If you follow the tips outlined before, you will quickly get more and more clients. But don't stop there. You must increase the demand for your services as long as you live. Why? Because you'll leverage one of the oldest principles of capitalism: supply and demand. Increase the demand while keeping the supply constant, and prices begin to rise—your prices.

The median number of friends of a Facebook user is 99 [Ugander, 2011]. So, every client gives you indirect access to 99 more people. A handful of those clients are potential customers for you. Think about it: Business owners tend to befriend other business owners. If you get the referral engine going, you will quickly have a

super high demand for your services. You will choose the clients for which you work. Thus, you will be able to decline projects that you don't like or that pay poorly.

The key question you should always ask is: how can I get the referral engine going? A few simple things will move the needle:

- **Ask:** Ask every client whether he knows someone who'd be interested in your service. Do this in addition to asking for the testimonial video. Of course, you shouldn't do both at the same time—but if you work together with a client for several projects (or several weeks), you can spread your requests out, giving a lot of value in between.
- **Gift Cards:** Give your client a gift card to be used for your next freelancing service. This one is huge! If you give your client a \$50 gift card for your services, he'll feel the strong desire to take you up on the offer. Nobody likes giving away free money. Do you?
- **Email Tactics:** Every successful business is long-term oriented. Focusing on the long term is one of Amazon's founding principles, and it helped make Amazon the largest retailer in the world. So what are some long-term tactics you can implement in the blink of an eye? Send all your customers an

email at least once per month. You can use a professional email management system, MailChimp, or just an Excel spreadsheet. Your list of former clients will become your most valuable asset. Treat it as such, and take care of your customers. Send them birthday or Christmas ecards, ask them how it's going in their business, and foster communication as much as possible.

There are many more tactics to ensure that customers of your freelance developer business are encouraged to refer more clients to you. But the best strategy is to provide awesome service. Give them great code, answer any of their messages, and ask them if everything's all right. And clients will come back and refer you to their friends.

Action steps:

- Outline your referral engine on a piece of paper. Why would clients refer you to other clients?

Expert Secret: Give, Give, Give, Right Hook

This is the title of a popular book of social media guru Gary V. There's some truth to it: to build a warm relationship with your customers, users, and business clients as a freelance developer, you must first give. The more you give, the more you'll receive. Don't be pushy and try to receive first. Don't ask for a raise before you haven't proven that you're worth it. Don't negotiate hard before you haven't given value to the other party. Of course, you can violate those rules, but it'll bring you no good.

As a freelance developer, you're in the people business, not just the coding business. You must learn to handle people. And if people sense that you're in it only to

extract maximum value for yourself, they will start to push against you. On the other hand, if you have the other person's best interests at heart, they will usually reciprocate and have your best interests at heart too. Some people will indeed rip you off. But you cannot protect yourself against it by ripping other people off—this will only increase the likelihood of it happening.

Video 30: *[Python Freelancer] You Are in The People Business (Not in the Coding Business)!*



<https://python-freelancer.com>

Action steps:

- Brainstorm and write down 20 things you can do to give value to your clients in advance.
- Include at least three of them in your acquisition and lead generation process.

Expert Secret: Befriend Colleagues

This is a no-brainer. Your colleagues are also freelance developers. If you have many of them as friends, it may easily happen that you can ignore all other tips listed here and still thrive in your coding business. Why? Because they will refer business your way, give you invaluable advice, show you what works to acquire clients and what doesn't, and teach you all the industry-specific tips and tricks that they've learned over the years.

The most effective strategies are often easy to do. But they are also easy not to do. Be the one who does them relentlessly.

How can you befriend a lot of freelance developers if you currently don't know a single one? Here are some tips:

Action steps:

- Invest 10-20 minutes in LinkedIn to connect with peers and potential clients and give value where you can. With this attitude, friendships will emerge quickly.
- Go to local startup meetups. This will help you increase your local community—rarely will you develop deep friendships 100% online.
- Go to technical talks and conferences in the area of computing and software engineering.
- Search for partners who want to build a thriving freelance developer business with you. This is the most prolific thing you can do to boost your speed of building a thriving coding business and winning lifelong friendships. Use any channel there is (including Facebook groups) to find possible partners for your business.
- Visit co-working spaces to find possible partners.

Expert Secret: On Investing

Investing is the most natural thing in the universe. Trees invest energy into seeds. Parents invest time, money, and sleepless nights into their children. Companies invest in employees. States invest in public school systems. They invest because they expect growth in their investments.

The one who invests more and invests smarter has a huge structural advantage. This is true for species, states, and corporations alike. It's also true for Python coders! So what is the best way to invest as a Python programmer?

There are fundamentally different ways you can invest as a Python coder. Here are the best ways of investing something now in order to get more back later:

Invest learning time now to save much more debugging time later. The more you learn the basics, the less time you'll spend debugging and solving problems. Any minute you use sharpening your skills now will come back to you many times over. As Abraham Lincoln said: *"Give me six hours to chop down a tree, and I will spend the first four sharpening the ax."* Learning Python is your way to sharpen your ax.

Invest money in the right learning material. For example, say you want to create your own thriving coding business. You can spend thousands of hours figuring out how to do it—and you'll be smarter after you've failed again and again. Or you can spend a few hundred bucks on learning material. Decide for yourself.

Invest in your network and your credibility. This one is huge—the ROI is much higher than anything else. Give everything you have and don't worry about it. The world rewards people who are willing to give more than they receive. Becoming loved and respected in your community is one of the most important things you can do, especially in the Python community, where people help and assist each other like no other programming language.

Invest in a good computer and internet connection. The value of time is much higher than the value of money. If you have time, you can always get more

money. But the reverse doesn't hold true: you cannot necessarily get more time by giving up money. Well, sometimes you can—by investing in a fast notebook and a fast internet connection, you'll work 10% faster. It'll cost you less than \$2000 per year. Although earning \$50,000 per year as a Python developer in the US can be seen as minimum wage, this investment would still return \$5000 per year (against the \$2000 invested). And the more you earn, the higher your ROI.

Invest 8h in your daily sleep and recreation. By sleeping properly, you can easily be 20% more productive. An additional benefit is that you'll be less prone to errors, which is especially important for coders: every bug can cost you tens of hours of debugging time—and much more if your users find it first. You have to sleep anyways, and there is nothing to gain by reducing your sleep time. So sleep longer and optimize your sleep quality.

Invest 10% of everything you earn. This is the traditional advice on how to invest—and it's sound. All the wealth you will ultimately acquire will be the one you have saved rather than spent on goods. By investing 10% of all capital that flows through your hands, you'll build yourself a growing capital base that increases by the amount you contribute and its appreciation in value. The smartest way to invest capital is to buy and hold a broad index fund. It's a mathematical necessity that,

over time, you'll beat even world-class investors with this simple strategy.

Proof sketch: divide the world into active and passive investors. Active investors trade a lot. Passive investors simply hold their portion of all the companies in the world. As a group, passive investors will match the index. As a group, active investors match the index minus the costs for trading. The expected return of a passive investor is higher than that of an active investor. QED.

These are only a few examples of how you can benefit from the idea of investing in yourself as a Python programmer.

Action steps:

- Write down how you will invest and create an automatic investment system. For example, *set up a savings plan into an exchange-traded fund owning a broad world index of diversified stocks*. This is not meant to be investment advice—only you can know what suits your needs and risk profile best.
- Save at least 10% of your income and invest it safely and steadily.

Expert Secret: Leave Freelancing Platforms

As a freelance developer, you can get a lot of traction on general freelancing platforms such as Upwork, Freelancer.com, or Fiverr and developer-specific platforms such as Topcoder. Check out which one is best for you.

However, these platforms are generally expensive. It's not part of a sound business strategy to rely forever on a third-party platform to send clients your way. You'll not only pay 20% of your income for platforms such as Upwork, but you'll also lose control of your clients and pricing strategy. Upwork can also kick you off their platform—and you could lose the business in minutes that took you years to build.

The best defense is to use, and not to get used by, these

freelancing platforms. Get a list of clients fast and then build from there. Move the contract to the outside, earn more money, and gain total control sooner, not later.

Action steps:

- Ask your best repeat customer to work with you outside of a freelancing platform.

Expert Secret: Maximize Control and Payout

How can you optimize your earning potential and control? You create your own platform. For example: set up a WordPress page where you offer your details and service offerings. You retain 100% of the control over your income, your projects, and even your testimonials. The sky's the limit. You can earn hundreds of dollars per hour if you are smart about it!

However, there is also the need for you to market your services. You need to install a marketing funnel. For example, you attract potential customers using Facebook ads. Then, you set up a landing page with a lead magnet so that they are motivated to give you their email address. Finally, you nurture your leads and send them

tons of value via email, thereby building a relationship with them.

While this all may seem complicated, it is the most profitable long-term strategy. But in the short term, most people should gain experiences and testimonials on the freelancing platforms and then gradually shift their focus towards their own platform as they get to know more and more clients.

Action steps:

- Do you want to sell your services on your own platform?
- Set up your WordPress page now! You can optimize it later as you see fit—but set it up now!

Expert Secret: Seek Independence

Let's stay for a moment and further explore the previous expert tip.

Freelancing platforms offer you convenience and speed in starting your own freelancing business. Examples are Upwork, Freelancer, or Fiverr. Some of those platforms (e.g., Upwork) will manually approve and activate your profile, so it will take some time to sign up. The reason is that these platforms receive thousands of applications every day. They have to be selective in order to guarantee a certain quality of service. I have read about many cases where Upwork rejected freelancers with attractive profiles.

On first impression, these platforms seem to be very attractive for your emerging freelance business. But be careful! Many existing freelancers rely too heavily on

these platforms. By using a platform such as Upwork, you make yourself vulnerable. Your income depends on the decisions of the platform owners. And don't think they have your best interest at heart. For example, Upwork already takes a 20% cut (!) from your hourly rate, just for brokering your services to potential clients. And their cut is pre-tax. You are working the first 12 minutes of each hour for Upwork and the next 28 minutes for the government.

What are the risks of depending on a freelancing platform? The platform owners can suddenly increase their cut. They can throw you from the platform for no reason whatsoever. They will stop sending clients your way as soon as their algorithm decides that you are not the optimal fit for a client project. The main problem is that you don't control your customer base. On the back of only these platforms, you cannot build a robust and sustainable business.

So what's the alternative? If you are serious about becoming a Python freelancer, setting up your own website is the way to go. This allows you to offer your services to clients all over the world. You can establish trust, and clients will perceive you as a professional freelancer. Use freelancing platforms to attract clients, but retain them for yourself.

To increase your conversions, add testimonials to your

website. Collect them during each of your freelancing activities. Of course, this is a bit harder than just setting up a freelancing account on Upwork. But you can figure it out with Google and YouTube. Building it this way is a much more sustainable foundation for your freelancing business. A nice plus is that your professional website sets you apart from your competition. It increases your chances of getting clients.

Only after creating your own web presence should you use these platforms to get new clients. Make sure to always refer to your professional website within any project application.

How do you retain clients beyond the first project? Focus on always over-delivering to new clients! Make them happy. Ask them to contact you directly the next time they need a similar service. And give them a special offer for their next freelancing service. Finally, ask them for referrals and testimonials after the job is done.

As you establish a growing client base that you control, you will find yourself using Upwork, Freelancer, and Fiverr less and less. Because of that, your profit margin will grow over time. Not only will your income increase, but it will also stand on a solid foundation. You own the foundation of your business which is your client base.

Action steps:

- Get a meaningful domain such as *python-freelancing-services.com*.
- Create a WordPress page introducing your services.

Expert Secret: Consultant, Not a Freelancer

One of my breakthroughs as a self-employed Python coder was to reposition myself as a consultant and coach rather than a freelance developer. You may ask (and rightly so): what's the difference?

The most important difference is in the clients' perception: A consultant is a trusted advisor while a freelance developer is a commodity. You want to be the trusted advisor because, as an advisor, you do less work yourself while charging higher hourly rates. As a freelance developer, you often end up doing the ugly stuff (like finding nasty small bugs in a complicated server setting) that takes hours and hours. As a consultant, you can talk about strategy and overall system architecture. Both

skill sets are highly valuable to the client, but the latter skill tends to get paid better. I don't know the reason, but that's just how the world works. Maybe it's just because the client cannot appreciate the complexity of setting up a program or finding a nasty bug. It just seems to be a small thing. As a rule of thumb: always sell big results.

After all, the client pays you for results, not for time invested. This may easily be the most important advice in this list of freelance developer tips. If you currently work in a job, you have the deeply ingrained feeling of getting paid for your efforts. If you put in little effort, you expect to get paid little. However, in the self-employment space, it's all about results. Time invested has little to no effect on your compensation. And it shouldn't have. What takes you minutes can take hours for someone with less experience. Work should always gravitate to the people that can finish it most effectively.

The rule to sell big results can also mean that you find nasty bugs. But now you're doing it only if the client is terrified of this bug. The best-case scenario is that they have already invested tens of hours but didn't find the bug. If you find the bug now (even if it takes you only an hour or so), they will gladly pay you large sums of money. Why? Because you saved them large sums of time and, thus, money.

It's all about the results.

Action steps:

- Write down how you plan to position yourself as an IT consultant rather than a freelance developer.
- What unique value will you provide as a consultant after your upgrade?

Expert Secret: Convert with Strategy Sessions

What's a strategy session? A strategy session is a free consultancy service you do for your client. It's the kick-off meeting that starts a regular consultancy service.

As a freelance developer with many leads but no customers—what's your problem? Most likely, the clients do not see how you can improve their lives. That's what strategy sessions are for. The most successful consultants that earn six or seven figures have done thousands of strategy sessions. That's their primary lead conversion strategy, and it works beautifully.

Why does it work?

There are many reasons why strategy sessions are among the most effective ways of converting a lead into a cus-

tomers. A few of them are listed as follows: You tap into the power of reciprocity. As you've given a lot to the lead, they feel the urge to give back—by buying your service. You show the client how useful your service is. If you can give clients value in your strategy session, they see exactly how you can help them, and they wouldn't like to miss you.

Using strategy sessions is a great sales conversion strategy (“the puppy sale”). If you want someone to buy a puppy from you, just give it to them in a free trial. As soon as they have tried the lovely puppy with its dark eyes and cute attitude, the puppy is sold. Free strategy sessions represent a low barrier to clients. You give value for free. This way, you draw them into your funnel, and there's virtually no logical reason for the clients to reject your offer. You learn what the client wants. The more strategy sessions you've implemented, the more you've learned what your clients want and how to sell to them. This is probably the most important point in this list.

What do free strategy sessions look like for software developers or data scientists? Just talk about where you see the potential of improving the code base, analyze parts of their company code, tell them the benefits (such as how much money they lose for every second delay), and give them a broad picture and clarity.

Virtually every business can benefit significantly from

improved IT solutions, so for a freelance developer, strategy sessions are a huge way of driving business. This is probably the most underutilized conversion strategy in the world. Use it and crush your competition!

Action steps:

- Every week, write down 10-20 clients to contact.
- Every day, write to one client with an offer for a free strategy session that's very specific to their business: how can you help them?
- Prepare for the clients that take you on the offer and create a nice (PowerPoint or Keynote) presentation. Show them how they can earn more money and where you come into the play.

Expert Secret: Alternative Tutoring

I work closely with hundreds of coders who want to become Python freelancers on the side or as their main profession.

It quickly became apparent that most upcoming Python freelancers don't believe that they can bring value to the marketplace. They think in terms of finishing software projects for companies. That's it.

But this is not the only route to take as a Python freelancer. An attractive industry is often overlooked: creating your own education business.

In your city today, there is a small percentage of people who want to learn Python. It may be 1% or 10%, but it's a significant number of people.

If you are a born teacher, consider starting your career as

a Python freelance teacher. Start with 1-on-1 education. Use Craigslist, Facebook groups, or the job board at your local university to find students you can teach, even if you are just starting. You only need to be one step ahead of them. Your student will appreciate that you can see their perspective. As you learn and become better, you can gradually increase your class sizes until you offer full-fledged seminars. I've seen people (e.g., the German Python seminar teacher B. Klein) who've done exactly that and now earn \$400 per seminary student—and they have dozens of students per seminar.

A great bonus is that as you create your local seminar business, you can gradually collect Python learning materials and publish them online to create a more scalable online business that exists independently of your own time.

To supplement your income, you can also register on an online tutoring site to educate people online as a Python tutor.

A clear road to success: your Python freelancing finances your education and your scalable online business creation.

Action steps:

- Is giving local seminars for you?
- Are you a teacher?

- Consider shifting your freelancing business to a more educational venture.

Expert Secret: Your Personal Brand

As a freelance developer, you need to connect with potential clients (leads) to generate business for you. Who hires freelance developers? Other businesses. Think about it. You're now part of the B2B marketplace. You need to attract other business owners and show them how you can be a valuable part of their business. You need to show them how you can make them more money than they pay you. It's a simple game once you've understood the rules.

So what's the best social media platform to connect with business owners in a professional environment? Correct: LinkedIn.

Microsoft's platform is the incarnation of a professionally-focused B2B social media platform. It's far easier to connect with business owners you don't know and still

make a professional, trustworthy impression. Why? Because the platform ensures that there's little spam. You need to act as a real human, not as an anonymous corporation. If you spam other people, you do this in your name, losing credibility. You cannot hide. That's why LinkedIn is superior to creating professional contacts.

Give more value than you're getting paid for.

This is how you slowly and steadily build your network. I recommend you to focus on one social network, and that's it. Don't lose your focus. Don't spend too much time on social media—that's what everyone else is doing, and it's not an effective use of your time. Instead, spend ten to twenty minutes per day on LinkedIn, increasing your network and improving your appearance. Then, focus on the other points outlined here.

Action steps:

- Create a LinkedIn account now.
- Dump the other social networks that distract you from your main objective: building a thriving coding business.

Expert Secret: Create a Blog

Video 31: *How to Get Passive Income as a Programmer with Blogging?*



<https://python-freelancer.com>

Creating a blog is a long-term strategy to build your market value. And it's fundamentally different from creating a social media account. Why? Because you'll create your own community—for yourself, not for some Silicon Valley billionaire.

Creating your own blog is long-term because it takes a few months until your articles rank for keywords on Google or other search engines. You have to spend time

writing articles, testing and tweaking the designs, maintaining your community, and learning about search engine optimization. You must invest significant effort for a long period.

But if you commit to staying in the game, it'll be the one thing that boosts your reach from hundreds of people to millions. The demand for your services will skyrocket while supply stays constant. You can charge premium prices like nobody else in your industry: hundreds of dollars per hour. Many bloggers also earn passive income through ads or sales of information products. They don't even sell their time anymore because they don't need to.

In addition to that, writing blog articles in a programming niche is an excellent way of building your expertise. Write about machine learning with TensorFlow, data science with Pandas, or neural network engineering with C++. This is how you'll build detailed, low-level, and robust knowledge in a field.

Action steps:

- Create your programming blog (e.g., with WordPress) now.
- Select a niche that you will master and dominate. The smaller the niche, the better—if you're a be-

ginner coder. You can always expand your niche once you've achieved dominance.

Video 32: *Passive Income as a (Python) Coder: 3 Simple Ideas*



<https://python-freelancer.com>

Final Remarks

Congratulations, you made it through this whole Python freelance developer textbook!

By reading this book, you should now have a much better understanding of how to create a thriving coding business online. In his famous 1962 book *The Success System That Never Fails*, W. Clement Stone proposed that there are three steps that will guarantee you success in life:

- **Inspiration to action:** increase your inherent motivation to thrive in your field.
- **Know-how:** learn which skills will get results for you.

- **Activity knowledge:** acquire knowledge about the small habits and activities that matter in your field.

After reading this book about Python freelancing success, you now know what to do and how to do it (*know-how*). You know how to develop good habits and which activities will boost your success as a freelance developer (*activity knowledge*). And hopefully, you now also have inspiration to take action!

Throw yourself into practice. Get your first client, focus on value creation, and overdeliver. The principles are simple and universal. Apply them quickly and go through the real-world feedback cycle.

- We worked hard to make this book worth your time and money. But no book can reach perfection without feedback from early adopters and highly active readers. For any feedback, questions, or problems you may have, please send me an email to admin@finxter.com.
- To grow your Python skills on autopilot, register for the free Python email course here: <https://blog.finxter.com/subscribe>.
- Do you want to learn how to sell your skills as a Python freelancer? Get the free “*How to Build*

Your High-Income Skill Python” webinar at <https://blog.finxter.com/webinar-freelancer/>.

- To join the Finxter freelancer program with guaranteed success, join us at <https://blog.finxter.com/become-python-freelancer-course/>

Let us express our deepest gratitude that you have spent your time meeting with us throughout this book. Above everything else, we value your time. The ultimate goal of any good textbook should be to *save time*. By working through this book, you have gained insights into building a thriving six-figure coding business online. But, only if you apply your skills to the real world will you experience a positive return on invested time and money. Keep investing in yourself, work on practical projects, and stay active within the Finxter community to continuously improve your Python skills.

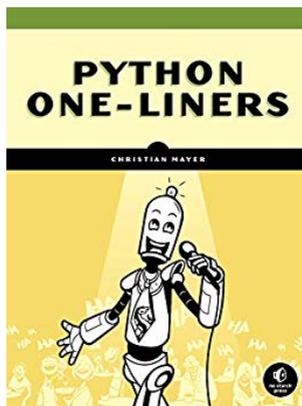
More Python Textbooks

The Finxter textbook series helps you master computer science with a focus on Python coding. Read on to learn about the other textbooks in the series.

Python One-Liners:

Python programmers will improve their computer science skills with these useful one-liners.

Python One-Liners will teach you how to read and write "one-liners": concise statements of useful functionality packed into a single line of code. You'll learn how to unpack and understand any Python code line and write eloquent, powerfully compressed Python like an expert.



Get the print book:

<https://www.amazon.com/gp/product/B07ZY7XMX8>

Coffee Break Python: 50 Workouts to Kickstart Your Rapid Code Understanding in Python.

The first bestselling book of the "Coffee Break Python" series offers 50 educative code puzzles, 10 tips for efficient learning, 5 Python cheat sheets, and 1 accurate way to measure your coding skills.



Get the ebook:

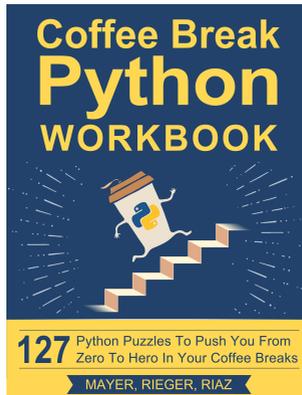
<https://blog.finxter.com/coffee-break-python/>

Get the print book:

<https://www.amazon.com/dp/B07GSTJPFD>

Coffee Break Python Workbook: 127 Python Puzzles to Push You From Zero to Hero in Your Coffee Breaks

This book is the chess grandmaster's way of learning Python. 127 unique and brand-new Python puzzles—each pointing to gaps in your knowledge, challenges you to guess a solution, and then explains potential solutions in an easy-to-understand manner.



Get the ebook:

<https://blog.finxter.com/coffee-break-python-workbook/>

Get the print book:

<https://www.amazon.com/dp/B07WQR91PF>

Coffee Break NumPy: A Simple Road to Data Science Mastery That Fits Into Your Busy Life.

Coffee Break NumPy is a new step-by-step system to teach you how to learn Python's library for data science faster, smarter, and better. It's easy: You solve practical Python NumPy puzzles as you enjoy your morning coffee.



Get the ebook:

<https://blog.finxter.com/coffee-break-numpy/>

Get the print book:

<https://www.amazon.com/dp/B07WHB8FWC>

Coffee Break Python Slicing: 24 Workouts to Master Slicing in Python, Once and for All.

Coffee Break Python Slicing is all about growing your Python expertise—one coffee at a time. The focus lies on the crucial slicing technique to access consecutive data ranges. Understanding slicing thoroughly is vital for your success as a Python developer.

As a bonus, you will track your individual Python coding skill level throughout the book.



Get the ebook:

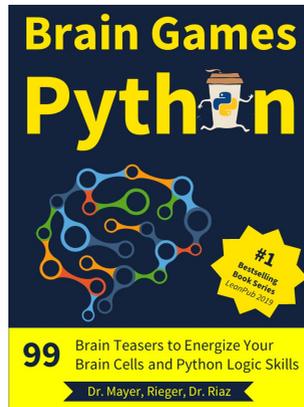
<https://blog.finxter.com/coffee-break-python/>

Get the print book:

<https://www.amazon.com/dp/B07KSHLLG5/>

Brain Games Python: 99 Brain Teasers for Beginners to Energize Your Brain Cells

Tired of brain fog? This groundbreaking new puzzle book revives your brain cells, boosts your computational intelligence, and launches your programming skills in Python—even if you have zero programming experience!



Get the ebook:

<https://blog.finxter.com/python-brain-games/>

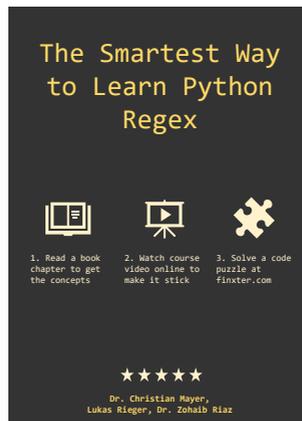
Get the print book:

<https://www.amazon.com/dp/B081NXFLZZ/>

The Smartest Way to Learn Python Regex

Google engineers are regular expression masters. They need to be because their search engine is a massive text-processing engine—when text meets computer science, regular expressions are already here.

The Smartest Way to Learn Python Regex transforms you into a regular expression master with the innovative three-step approach: (i) read a book chapter, (ii) watch the chapter video, and (iii) solve a code puzzle.



Get the ebook:

<https://blog.finxter.com/ebook-the-smartest-way-to-learn-python-regex/>

Get the print book:

<https://www.amazon.com/dp/B0874L1573>

Coffee Break Python – Mastery Workout

How do grandmasters of chess become that way? They study thousands of chess positions. They look at a chess position, commit to a solution, and compare it against the “gold standard”—the best move in a given position.

This book is the chess grandmaster’s way of learning Python. It offers you 99 challenging, exclusive, and brand-new Python puzzles. Every puzzle points to gaps in your knowledge, challenges you to guess a solution, and then explains the result.



Get the ebook:

<https://blog.finxter.com/coffee-break-python-mastery-workout-ebook/>

Get the print book:

<https://www.amazon.com/dp/B0892DP5WY/>