



# My Learning Plan

Mahdi Haghverdi

March 20, 2023

# Contents

<b>Preface</b>	<b>ii</b>
<b>1 Courses</b>	<b>1</b>
1.1 Deep learning . . . . .	2
1.2 FastAPI . . . . .	3
<b>2 Books</b>	<b>4</b>
2.1 Deep Learning for Coders With Fastai and PyTorch . . . . .	6
2.2 Fluent Python . . . . .	8
2.3 Build a Career in Data Science . . . . .	10
2.4 Inside the Python Virtual Machine . . . . .	12
2.5 CPython Internals . . . . .	14
2.6 Rust in Action . . . . .	15
2.7 Docker Deep Dive . . . . .	17
2.8 Docker in Action . . . . .	19
2.9 Zero to One . . . . .	21
<b>3 Habits</b>	<b>23</b>
3.1 Podcast . . . . .	24
3.2 Music . . . . .	24
3.3 Films . . . . .	25
3.4 Biking . . . . .	25
<b>4 Career</b>	<b>26</b>
<b>5 Norouz</b>	<b>30</b>

# Preface

After going up and down in my life, I finally found the way to study and learn well. This document has everything I want to do and resource I want to check. This plan will be started from the summer.

After spending one whole semester in the university, it is got clear to me, that meanwhile studying, learning **new** things, is hard! It's possible but it's hard. So summers are the times which I have to value a lot and to make the most outta 'em.

At the time of writing this, there are 5 days remaining of the year, a long and hard year both for Iran, my love, and myself. I learned and grew a lot during this year. I spend a whole semester in university, experienced a really *Randomly-Generated-but-Related* journey. Studied a lot, learned how to study well and meet many new and nice people.

But after all, I have to create a path to my actual career! After working and searching I've finally made my decision and chosen: *Data Science*. I really like this field, it's very amazing but hard to learn :) Beside this, I wanna learn backend engineering as well, 'cause it is much faster (faster to get result, lol) and has a slightly smaller learning curve.

This plan will be started from the summer of 1402, but I will be working and adding things to this document from time to time; there may be some new chapters or reorganization of the chapter but not now :D

This will be a hot summer. It gets hot because of so many hours I want to study, both with video courses and with reading books. To learn data science I asked someone in our university who already is a data scientist, and he introduced me to the deep learning course of *Jeremy Howard*. This

course is freely available on YouTube. Also this course needs a book nearby (written by Jeremy Howard) called *Deep Learning for Coders With Fastai and PyTorch*.

Besides learning data science, learning new things and refreshing my Python skills is something that I would never miss and enjoy a lot. The focus of this summer will be on the CPython implementation itself, so I've chosen some nice books on this field. I have gathered some more general books on Python too, but they would have a lesser priority.

Watching video courses and reading books are the main activities of the summer, but without spending or more precisely, *acquiring* time to rest, nobody can learn anything; listening to music and podcast and watching movies are the main non-active and biking is the main physical activity of my summer.

*Some knowledge needs to be learned by video courses  
in a short, and then be completed by reading books  
and researching in a long journey.*

Me

# 1

## Courses

Well, the major part of the summer is the courses, in this chapter I will introduce the courses I want to watch and some useful information about them.

Each course is tagged with these:

- Learning course
  - **Hard**
  - **Easy**
- Video duration
  - **Long**
  - **Short**
- Video Count: **Many**

## 1.1 Deep learning

The career I chose, “The sexiest job of the 21st century,” “The best job in America,” “One of the newest and coolest jobs in IT world”

---

Hard

Long

Well as the quote says, this is exciting, BUT it really needs knowledge and effort to gain, the starting point is here: this nice deep learning course at <https://course.fast.ai/>.

### Practical Deep Learning

*A free course designed for people with some coding experience, who want to learn how to apply deep learning and machine learning to practical problems.*

*This free course is designed for people (and bunnies!) with some coding experience who want to learn how to apply deep learning and machine learning to practical problems.*

Each lesson of the course has a video and a dedicated page of the website, like [Lesson 1](#).

Notes:

- Each video is pretty long, on average they last one hour and 30 minutes! The longer the video, the more knowledge they contain AND the more attention they need AND the more practice as well.
- Each lesson has a *How to complete lesson N* section (like [here](#)), in which says: *As well as watching the video and working through the notebooks, you should also read the relevant chapter(s) of the fast.ai book, Practical Deep Learning for Coders. Each lesson will tell you what chapter you need to read, just below the video*

After all, the dedicated time and effort must be huge.

## 1.2 FastAPI

“FastAPI framework, high performance, easy to learn, fast to code, ready for production”

---

[fastapi.tiangolo.com](https://fastapi.tiangolo.com)

Easy

Short

Many

We all know FastAPI and there is no need for more introduction. The course I wanna watch is <https://youtu.be/0s0vCWFmrtA>

The course is a 19-hour long video which is downloaded and cut already. The important things are:

- The course is created one year ago and it is slightly outdated, so the [documentation](#) must be read along the course.
- Course will just show you how to use FastAPI and the real learning happens when doing projects.
- It's good to define nice projects whenever I came to an idea e.g. *The API to send pictures of <https://unsplash.com> to my friends :)*

*I find television very educating. Every time somebody turns on the set, I go into the other room and read a book.*

Groucho Marx

*There is more treasure in books than in all the pirate's loot on Treasure Island.*

Walt Disney

## 2 Books

You may argue me about what I want to say, however I found it true: “If you want to learn *something*, the real source is the books about it.” This is absolutely true. No video course and no article and nothing else may contain the most valuable information about something. All the video courses and simple article ARE come from the books.

In this chapter I may introduce many invaluable books (most technical books but it may contain other books as well.)

Each section of this chapter has the name of the book which is a link to the post of that book in the store: [Skybooks.ir](https://skybooks.ir), where I buy the books from. Then there maybe some tags under the name of the books, the whole tags are categorized in three categories:

### 1. Reading priority

- **must-be-read**
- **better-to-read**
- **later-to-read**

### 2. Buying considerations:



(a) Number of pages

- near-one-thousand-pages
- near-seven-hundred-pages
- near-five-hundred-pages
- near-two-hundred-pages

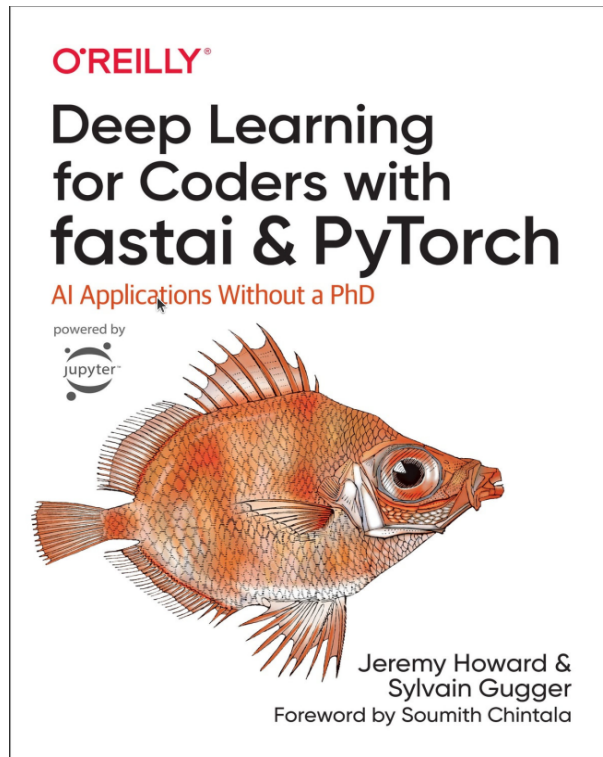
(b) Price:

- number T
- number T
- number T
- number T
- number T

3. Book read: read

4. Printed or not: printed

## 2.1 Deep Learning for Coders With Fastai and PyTorch

**must-be-read****near-seven-hundred-pages****297 T**

Deep learning is often viewed as the exclusive domain of math PhDs and big tech companies. But as this hands-on guide demonstrates, programmers comfortable with Python can achieve impressive results in deep learning with little math background, small amounts of data, and minimal code. How? With fastai, the first library to provide a consistent interface to the most frequently used deep learning applications show you how to train a model on a wide range of tasks using fastai and PyTorch.

You'll also dive progressively further into deep learning theory to gain a complete understanding of the algorithms behind the scenes. Train models in computer vision, natural language processing, tabular data, and collaborative filtering. Learn the latest deep learning techniques that matter most in

practice Improve accuracy, speed, and reliability by understanding how deep learning models work Discover how to turn your models into web applications Implement deep learning algorithms from scratch Consider the ethical implications of your work.

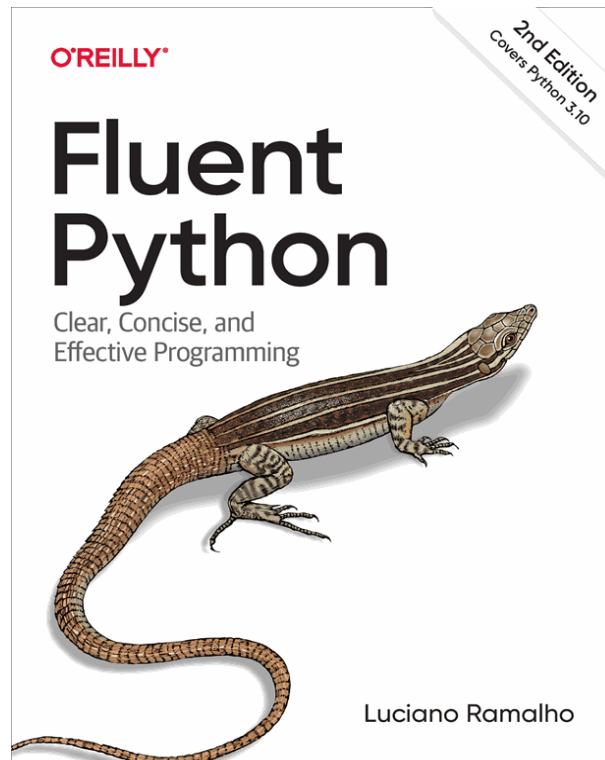
As a 622-page and comprehensive book, it needs attention and time. The book may get printed, just to be read easily.

## 2.2 Fluent Python

better-to-read

near-one-thousand-pages

505 T



Python's simplicity lets you become productive quickly, but often this means you aren't using everything it has to offer. With the updated edition of this hands-on guide, you'll learn how to write effective, modern Python 3 code by leveraging its best ideas. Don't waste time bending Python to fit patterns you learned in other languages. Discover and apply idiomatic Python 3 features beyond your past experience. Author Luciano Ramalho guides you through Python's core language features and libraries and teaches you how to make your code shorter, faster, and more readable.

Featuring major updates throughout the book, *Fluent Python*, second edition, covers:

1. Special methods: The key to the consistent behavior of Python objects

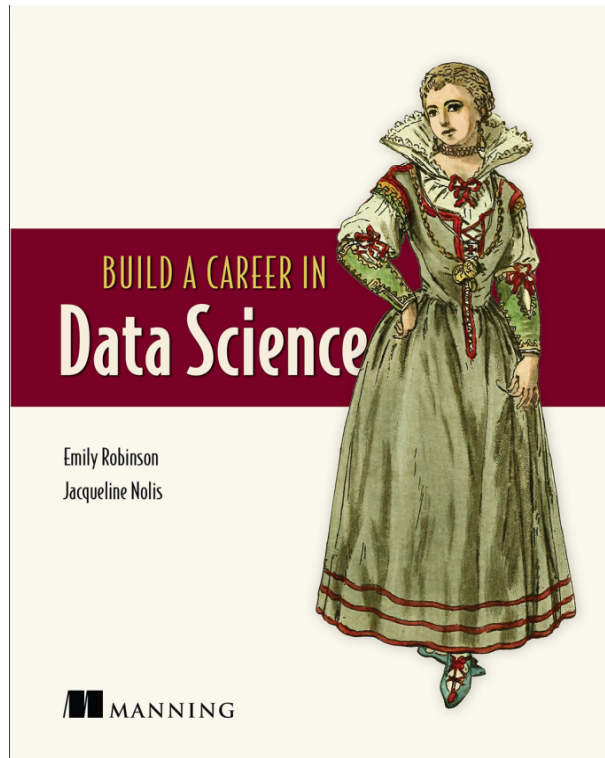
2. Data structures: Sequences, dicts, sets, Unicode, and data classes
3. Functions as objects: First-class functions, related design patterns, and type hints in function declarations
4. Object-oriented idioms: Composition, inheritance, mixins, interfaces, operator overloading, static typing and protocols
5. Control flow: Context managers, generators, coroutines, `async/await`, and thread/process pools
6. Metaprogramming: Properties, attribute descriptors, class decorators, and new class metaprogramming hooks that are simpler than meta-classes

## 2.3 Build a Career in Data Science

later-to-read

near-five-hundred-pages

194 T



What are the keys to a data scientist's long-term success? Blending your technical know-how with the right "soft skills" turns out to be a central ingredient of a rewarding career.

Build a Career in Data Science is your guide to landing your first data science job and developing into a valued senior employee. By following clear and simple instructions, you'll learn to craft an amazing resumé and ace your interviews.

In this demanding, rapidly changing field, it can be challenging to keep projects on track, adapt to company needs, and manage tricky stakeholders. You'll love the insights on how to handle expectations, deal with failures, and plan your career path in the stories from seasoned data scientists included in

the book.

What's Inside: • Creating a portfolio of data science projects • Assessing and negotiating an offer • Leaving gracefully and moving up the ladder • Interviews with professional data scientists For readers who want to begin or advance a data science career.

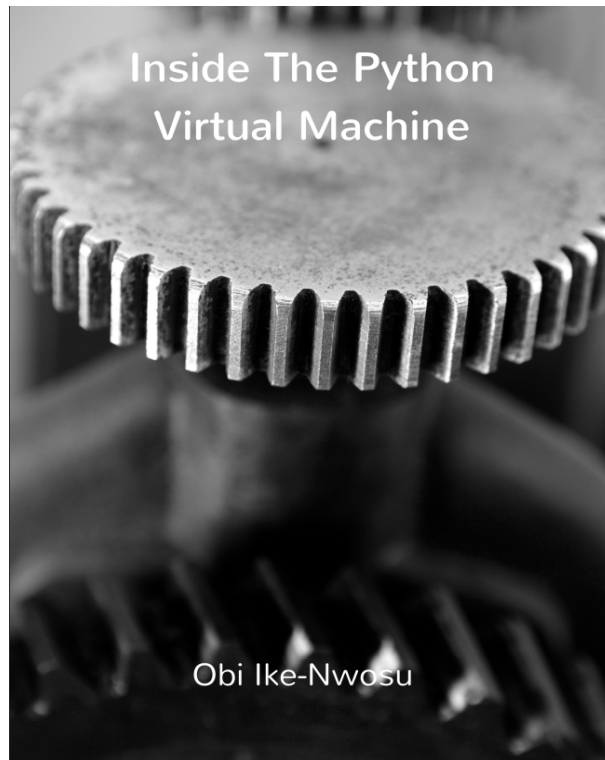
Emily Robinson is a data scientist at Warby Parker. Jacqueline Nolis is a data science consultant and mentor.

## 2.4 Inside the Python Virtual Machine

**must-be-read**

near-two-hundred-pages

88 T

**printed**

Inside the Python Virtual Machine provides a guided tour under the covers of the Python interpreter for the curious pythonista. It attempts to show the user what happens from the moment the user executes a piece of Python code to the point when the interpreter returns the result of executing the piece of code.

This book will provide the readers with an understanding of the various processes that go into compiling and executing a python program removing most of the mystery surrounding how the python interpreter executes source code.

The books starts out with a description of the compilation phase with emphasis on the less generic parts of the compilation phase. It then proceeds to discuss python objects and their implementation in CPython. This is



followed by a discussion of various objects types that are central to the interpreter such as frame objects and code objects. The process of evaluating code objects by the interpreter loop is also discussed as well as how to extend the Python programming language with your own constructs.

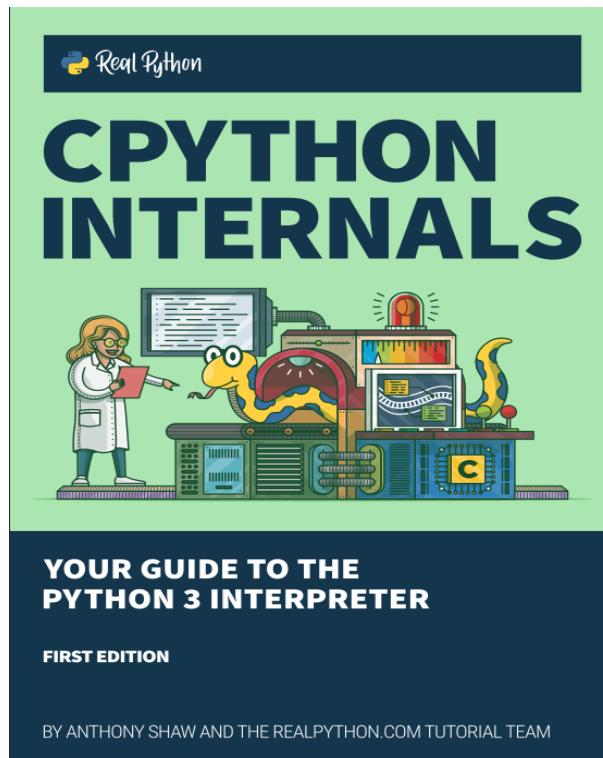
## 2.5 CPython Internals

must-be-read

near-five-hundred-pages

209 T

printed



CPython Internals: Your Guide to the Python 3 Interpreter.

Are there certain parts of Python that just seem like magic? Once you see how Python works at the interpreter level, you'll be able to optimize your applications and fully leverage the power of Python.

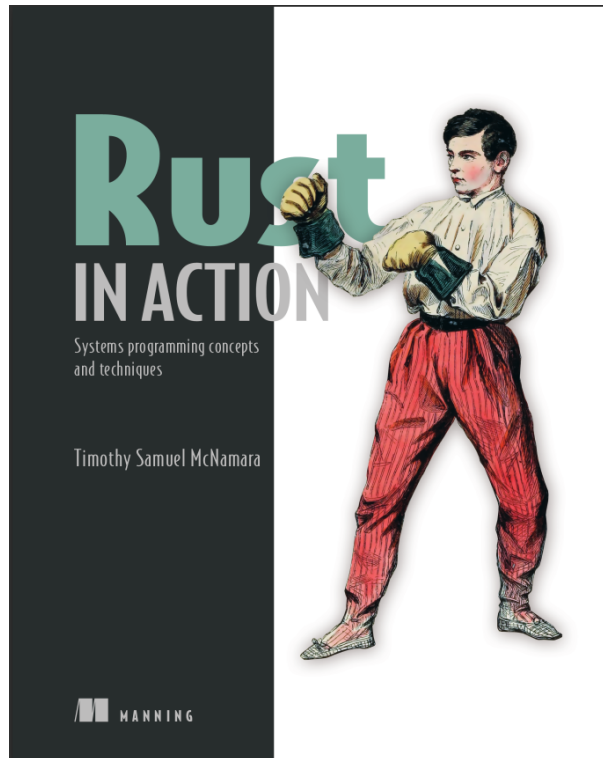
In CPython Internals, you'll unlock the inner workings of the Python language, learn how to compile the Python interpreter from source code, and cover what you'll need to know to confidently start contributing to CPython yourself!

## 2.6 Rust in Action

later-to-read

near-five-hundred-pages

234 T



*Rust in Action* introduces the Rust programming language by exploring numerous systems programming concepts and techniques. You'll be learning Rust by delving into how computers work under the hood. You'll find yourself playing with persistent storage, memory, networking and even tinkering with CPU instructions. The book takes you through using Rust to extend other applications and teaches you tricks to write blindingly fast code. You'll also discover parallel and concurrent programming. Filled to the brim with real-life use cases and scenarios, you'll go beyond the Rust syntax and see what Rust has to offer in real-world use cases.

### about the technology

Rust is the perfect language for systems programming. It delivers the low-level power of C along with rock-solid safety features that let you code fear-

lessly. Ideal for applications requiring concurrency, Rust programs are compact, readable, and blazingly fast. Best of all, Rust's famously smart compiler helps you avoid even subtle coding errors.

**about the book**

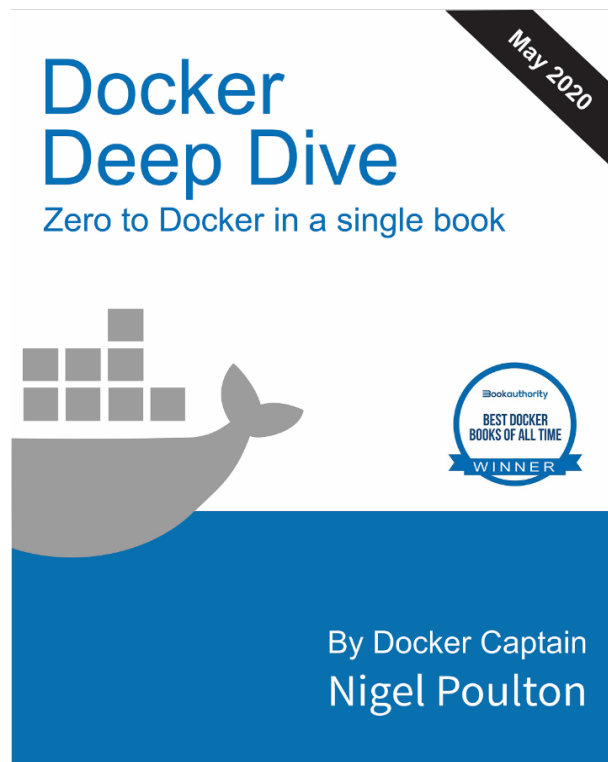
*Rust in Action* is a hands-on guide to systems programming with Rust. Written for inquisitive programmers, it presents real-world use cases that go far beyond syntax and structure. You'll explore Rust implementations for file manipulation, networking, and kernel-level programming and discover awesome techniques for parallelism and concurrency. Along the way, you'll master Rust's unique borrow checker model for memory management without a garbage collector.

## 2.7 Docker Deep Dive

better-to-read

near-two-hundred-pages

155 T



Most applications, even the funky cloud-native microservices ones, need high-performance, production-grade infrastructure to run on. Having impeccable knowledge of Docker will help you to thrive in the modern cloud-first world. With this book, you'll gain the skills you need to work with Docker and its containers.

The book begins with an introduction to containers and explains its functionality and application in the real world. You'll then get an overview of VMware, Kubernetes, and Docker and learn to install Docker on Windows, Mac, and Linux. Once you've understood the Ops and Dev perspective of Docker, you'll be able to see the big picture and understand what Docker exactly does. The book then turns its attention to the more technical aspects,

guiding you through practical exercises covering Docker engine, Docker images, and Docker containers. You'll learn techniques for containerizing an app, deploying apps with Docker Compose, and managing cloud-native applications with Swarm. You'll also build Docker networks and Docker overlay networks and handle applications that write persistent data. Finally, you'll deploy apps with Docker stacks and secure your Docker environment.

By the end of this book, you'll be well-versed in Docker and containers and have developed the skills to create, deploy, and run applications on the cloud.

What you will learn:

- Become familiar with the applications of Docker and containers
- Discover how to pull images into Docker host's local registry
- Find out how to containerize an app
- Build and test a Docker overlay network in the swarm mode
- Use Docker compose to deploy and manage multi-container applications
- Securely share sensitive data with containers and Swarm services

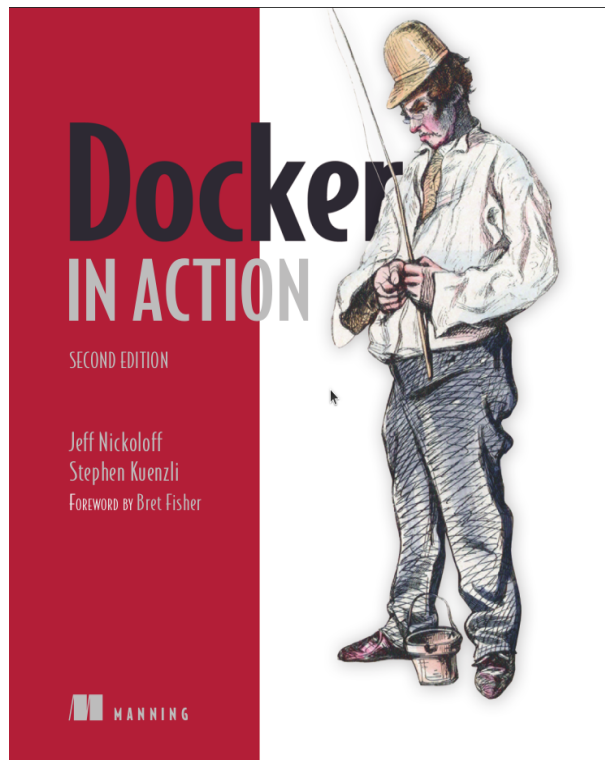
## 2.8 Docker in Action

must-be-read

near-five-hundred-pages

189 T

printed



*Docker in Action, Second Edition* teaches you the skills and knowledge you need to create, deploy, and manage applications hosted in Docker containers. This bestseller has been fully updated with new examples, best practices, and a number of entirely new chapters.

### about the technology

The idea behind Docker is simple—package just your application and its dependencies into a lightweight, isolated virtual environment called a container. Applications running inside containers are easy to install, manage, and remove. This simple idea is used in everything from creating safe, portable development environments to streamlining deployment and scaling for microservices. In short, Docker is everywhere.

**about the book**

*Docker in Action, Second Edition* teaches you to create, deploy, and manage applications hosted in Docker containers running on Linux. Fully updated, with four new chapters and revised best practices and examples, this second edition begins with a clear explanation of the Docker model. Then, you go hands-on with packaging applications, testing, installing, running programs securely, and deploying them across a cluster of hosts. With examples showing how Docker benefits the whole dev lifecycle, you'll discover techniques for everything from dev-and-test machines to full-scale cloud deployments.

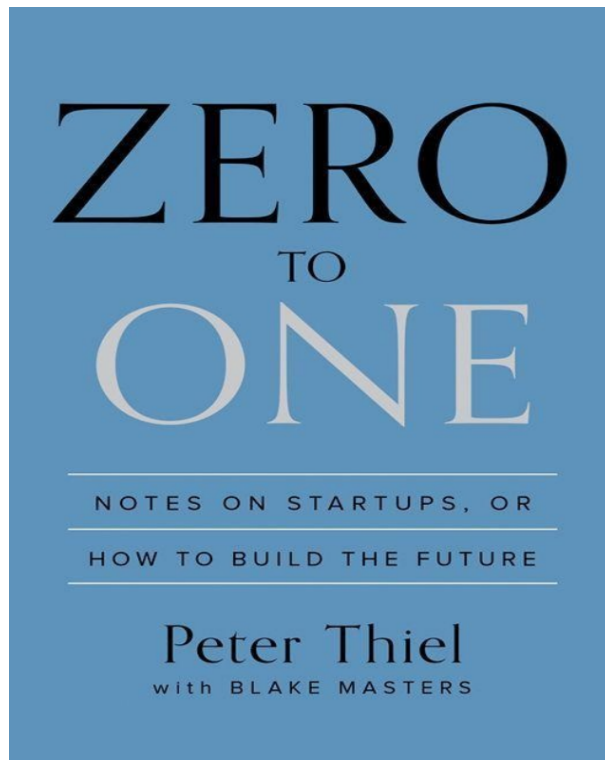


## 2.9 Zero to One

later-to-read

near-two-hundred-pages

110 T



Thiel starts from the bold premise that we live in an age of technological stagnation, even if we're too distracted by our new mobile devices to notice. Progress has stalled in every industry except computers, and globalization is hardly the revolution people think it is. It's true that the world can get marginally richer by building new copies of old inventions, making horizontal progress from "1 to n." But true innovators have nothing to copy. The most valuable companies of the future will make vertical progress from "0 to 1," creating entirely new industries and products that have never existed before. Zero to One is about how to build these companies. Tomorrow's champions will not win by competing ruthlessly in today's marketplace. They will escape competition altogether, because their businesses will be unique. In today's post-internet bubble world, conventional wisdom dictates that all the good ideas are taken, and the economy becomes a tournament in which

everyone competes to reach the top. Zero to One shows how to quit the zero-sum tournament by finding an untapped market, creating a new product, and quickly scaling up a monopoly business that captures lasting value. Planning an escape from competition is essential for every business and every individual, not just for technology startups. The greatest secret of the modern era is that there are still unique frontiers to explore and new problems to solve. Zero to One shows how to pursue them using the most important, most difficult, and most underrated skill in every job or industry: thinking for yourself”—Provided by publisher.

*Replacing just a few key negative habits with a few positive habits can easily be the difference between being mostly unhappy and being happy almost all of the time.*

Tynan, Superhuman by Habit

# 3

## Habits

To continue to live and rest I found some ways to spend my leisure time, just like the books they've got small tags to make them more clear.

The tags:

1. **always** which says this item has no specific time.
2. **weekends** the weekends are good for this item
3. **time-to-time** you can do it whenever you want to
4. **activity** needs body activity
5. **rest** is done to rest and NOT to think about anything else

## 3.1 Podcast

“Listening to podcast, reading a book, listening to an audiobook and watching films isn't waste of time. It's how somebody becomes wise!”

---

Deyth Banger

time-to-time

activity

Talk Python to Me is a weekly podcast hosted by developer and entrepreneur Michael Kennedy. We dive deep into the popular packages and software developers, data scientists, and incredible hobbyists doing amazing things with Python. If you're new to Python, you'll quickly learn the ins and outs of the community by hearing from the leaders. And if you've been Pythoning for years, you'll learn about your favorite packages and the hot new ones coming out of open source.

Listening to old and new Python podcasts, specially from *Talk Python to Me* channel helps a lot, It provides many information, introduces mane new technologies and brings many good people to the show.

## 3.2 Music

“Music doesn't lie. If there is something to be changed in this world, then it can only happen through music.”

---

Jimi Hendrix

always

As always: *I live because of my musics :)*

### 3.3 Films

“Movies are like an expensive form of therapy for me. ”

---

Tim Burton

weekends

rest

Watching new films and series, enjoys a LOT. The dedicated series which I want to watch is *The Mentalist*. A TV show which reminds me of Sherlock!

Also I will download the films which I haven't watch in 1080p quality and enjoy the leisure :)

### 3.4 Biking

“Life is like riding a bicycle. To keep your balance you must keep moving”

---

Albert Einstein

time-to-time

activity

Physical activity to become fresh again, and to empty the mind outta the learned stuff is really important and Biking has to be done in summer.

*Find out what you like doing best, and get someone to pay you for it.*

Katharine Whithorn

*The only way to do great work is to love what you do. If you haven't found it yet, keep looking. Don't settle.*

Steve Jobs

# 4

## Career

Well, **the career to choose**. It may be a simple question: *What do you wanna do buddy?* But the answer? absolutely not. You have to chose something that you will be doing in the rest of your life... I've talked about *What* and *Why* I have chosen in this chapter :D

Choosing what to do. If you are young (somehow under 16 or 17) you may have many options in your head, or have none, it's totally natural. You are not OLD enough and your brain doesn't care about this, but after a while, becoming an adult, your brain, will create something in your head that you say to yourself "I have to finally make my decision and choose something to do and to be!" Well it happened for me. When I was 17 (11th grade at high school) I thought: "well I don't what to be?! How about to follow and be a good example of the best of the things I see?"

Our calculus teacher was the best ever I saw, well I decided to be a very nice math teacher. I talked to him and he told: "You are absoulty a foo, go and find something which really fits you. It will a waste for you to be teacher!!" He was somehow right, I reallized in a while. I love teaching but it is not my type, I needed something more attractive.

Once a day, a friend of mine told me: "Go and find yourself a good video course for programming. You will definitely love programming, I see it inside of you buddy." Well after some searching I found *Python masterclass* by *Tim Buchalka* AND pandemic hit :)) I started watching this reallllllly niiaiiiiice course and finally found it: "**Programming**", with no doubt programming will be the thing I was seeking :D

The journey started. I was seeing nearly 5 videos per day, seriously taking notes and reading Python documentation. I was the happiest Python learner in the world. I read, nearly every part of the documentation, every *type*<sup>1</sup> which was taught by Tim, I read the documentation afterwards and learned way more than the course.

Time passed and I got into 12th grade and abandoned learning Python due to *entrance exam of university* or in short *KonKoor*. I really stopped learning, after a very long year and one day after the exam, I downloaded nearly 60GB of videos just for one course: *Python Deep Dive 1-4*. This is the most brilliant course ever, it really teaches you the best things and the most important things you HAVE TO know. I finished it before starting of the first semester of my university.

---

<sup>1</sup>type: the datatypes which are built-in to Python such as `str`, `int` and `dict` types

Thanks to Allah, I was accepted as a *Computer Engineering* student at *Isfahan University*. I started joining Python communities created in Telegram, and after the first semester finished, I became one of the Admins of those groups :D

Then I found someone who I really like and is one of my best friends :) *Seyed Iliya*. He is really nice, and a really nice Python developer as well. We started working on real world projects and had a collaboration of something called JsonicDB<sup>2</sup>

At that time, WE wanted to be professional back-end engineers and started watching courses on that field, BUT:

- for me: the 3rd semester started and it was not online (thanks Allah for that)
- for him: He had to study for KonKoor

So everything was nearly stopped again :/

However going to university, although just for one semester, changed my way, it really changed it. After studying math, searching for the hot fields in programming and meeting new people with new expertise, I made my one-step-to-final decision: *Data Science*.

Data science is sexy isn't it? It needs a deep knowledge of the both worlds, programming and science, also learning it takes time and effort. The 3rd semester passes and I more fall in love with the field.

After searching and making my final decisions, I met Homayoun Sadeghi, a DevOps expert. I remembered the days I liked back-end and DevOps world and missed it :( but there were a solution for it: *MLOps* field, one of the hardest fields anyone can choose and the final decision was made: "I want to be an MLOps specialist :D"

I have to confess that I really like this journey AND this is a really hard field to learn and to be a good one in it. It needs many knowledge about

---

<sup>2</sup>a database toolkit, written specifically for JSON datatype



maaaaaaanay things, It is shaped from three worlds: Machine learning (data science) + Development (programming) + Operation (the server and all of its stuff world).

After all this is what I chose and this is the things which shows my learning and career way, I may or may not change it future. I just concentrate on learning and learning and learning.

Thanks

*The ancient celebration of going to the new year, I  
love it*

ME

5  
Norouz

What I want to do in norouz :)

What i wanna do

1. Searching about Ghasem Soleimani documentries, vidoes, voices and...
2. Searching about the packages we have to choose for our 5th term
3. Searching about the big plan :)