

Forum:[Hack Forums](#)

Thread: [My vision on how to learn Python](#)

Sapience - April 19, 2017, 7:49 am

Personally, I believe the best way to learn Python is not through following some book or tutorial but to simply try making something your own way and then comparing your code to that of someone else (preferably someone experienced in the programming language) who made the same project.

That's why I would say docs.python.org is the best source of information when learning Python (and the only one you'll ever need): it shows you all the functions while not spoonfeeding you with programs. With a little bit of common sense and some dedication, you can learn the language and in the end be a better programmer than the people who chose to be spoonfed: learning programming isn't just about knowing all the different functions, it's about being able to figure out how to put them all together on your own as well (you won't always have a step-by-step tutorial to guide you through the process).

That doesn't mean I dislike forums like these where users can go to ask for help with a certain problem, but it just hurts me to see that some people come here expecting others to hold their hand and guide them through the process step by step and that some people even take the time to do it without realizing it's actually lost effort.

For those who are reading this in order to receive guidance on how to get started, here are some projects that might be interesting to make, ordered from easiest to harder (please do not just look up a step by step tutorial on these): Take a number chosen by the user and show its multiplications from 1 to 10 (e.g., 3 would be 3, 6, 9, 12, 15, 18, 21, 24, 27, 30.) Preferably with the exact format of the example (with a dot at the end, not a comma)

- Display the numbers of Fibonacci until they exceed a number chosen by the user.

- Make a shell-based calculator that takes three user inputs: two numbers and an operator.

- Use a .txt file as a vocabulary list (word: definition) and interrogate the user on it by displaying the definition and asking for the word in a random order. Then, display the user's score along with the correct answers to any words they got wrong.

- Make GUI-based versions of the previous programs using Tkinter.

- Add threading to the Fibonacci project, so that it can go up to incredibly high numbers without taking too long.

If you're in need of some more projects, and others to compare your code with, there's a handy subreddit called /r/beginnerprojects that contains some useful stuff. Also, Google can definitely help you with some ideas. Good luck!

CaptainBlack - April 19, 2017, 8:12 am

Personally, I believe the best way to learn Python is not through following some book or tutorial but to simply try making something your own way and then comparing your code to that of someone else (preferably someone experienced in the programming language) who made the same project.

You need to combine both imho. A good textbook will ensure that you learn proper coding habits and pick a programmer's mindset, but it's also trivial to apply what you learn by coming up with a project of your own.

SIM-1866423 - April 19, 2017, 8:24 am

I have short way to do this, bruteforce your brain. Read everything that related python and decide

which one is more important to you that's all, lol.

Sapience - April 19, 2017, 8:25 am

(Apr 19, 2017 08:12 AM)CaptainBlack Wrote: [\[CITE\]](#) *Personally, I believe the best way to learn Python is not through following some book or tutorial but to simply try making something your own way and then comparing your code to that of someone else (preferably someone experienced in the programming language) who made the same project.*

You need to combine both imho. A good textbook will ensure that you learn proper coding habits and pick a programmer's mindset, but it's also trivial to apply what you learn by coming up with a project of your own.

I can definitely see how a combination of the two can be used, but I personally think it's better to discover things on your own and learn the proper habits by examining other people's code rather than by simply being told what to do. Everyone learns in a different way, in the OP I was simply saying what I believe to be the best way but that might not apply to you.

(Apr 19, 2017 08:24 AM)Mnemosyne Wrote: [\[CITE\]](#) I have short way to do this, brute force your brain. Read everything that related python and decide which one is more important to you that's all, lol.

I'm sorry, what exactly do you mean by "and decide which one is more important to you that's all"? I don't really understand.

[BLANK] - April 19, 2017, 5:43 pm

Your method is a bit like taking teaching someone how to swim by throwing them into the ocean a mile out and tell them that if they can make it to each boeey they will get small pieces that will help them eventually make it to shore. Starting with a book or course would be like teaching a person how to swim by taking them to the shallow end of a pool and having an instructor giving them the fundamentals of swimming and some practices to enforce the basics before they move on to the deep end or the ocean.

I agree that after someone learns the fundamentals of not only Python but of Object Oriented Programming in general, then taking on projects is the best way to enhance skills and not always rely on books or python recipes.

matt dax - April 19, 2017, 7:16 pm

(Apr 19, 2017 08:24 AM)Mnemosyne Wrote: [\[CITE\]](#) I have short way to do this, brute force your brain. Read everything that related python and decide which one is more important to you that's all, lol.

I prefer this dudes way of learning, its how I originally got into hacking. If you can focus for a long time and not get bored, its the way.

Sapience - April 20, 2017, 9:10 am

(Apr 19, 2017 05:43 PM)EasyGhost Wrote: [\[CITE\]](#) Your method is a bit like taking teaching someone how to swim by throwing them into the ocean a mile out and tell them that if they can make it to each boeey they will get small pieces that will help them eventually make it to shore. Starting with a book or course would be like teaching a person how to swim by taking them to the shallow end of a pool and having an instructor giving them the fundamentals of swimming and some practices to enforce the basics before they move on to the deep end or the ocean.

I agree that after someone learns the fundamentals of not only Python but of Object Oriented Programming in general, then taking on projects is the best way to enhance skills and not always rely on books or python recipes.

"throwing them into the ocean a mile out"

I'm sorry, but I disagree with this comparison. There's a huge difference between giving someone

the very basic building blocks they need (docs.python.org, check my original post) to do some basic calculations with a user input to then move on to stuff that requires a few more functions each time and just throwing someone who can't swim into the ocean a mile out.

When you start off learning with the method I described in my original post, your habit in programming will be to find a way to make things work using just the documentations instead of just taking chunks of code from someone else. I believe that that's a much better idea than just spoonfeeding them their first projects, which every book I've read so far does.

When you have the mindset of taking your time to figure out how to make something work with only some basic information, that'll make you a better programmer because it'll be far easier for you to identify mistakes if you've trained yourself in doing that.

DSTAT - April 20, 2017, 3:33 pm

I didn't learn Python though any books. Reading books, in my opinion, is boring and tedious. I think the best way to learn is interactively and then try to create something, search things to learn new modules.

Origin.69 - April 20, 2017, 3:36 pm

The real way to learn it is time, Its just like learning anything else takes time and practice.

Sapience - April 21, 2017, 7:57 am

(Apr 20, 2017 03:33 PM)DSTAT Wrote: [\[CITE\]](#) I didn't learn Python though any books. Reading books, in my opinion, is boring and tedious. I think the best way to learn is interactively and then try to create something, search things to learn new modules.

I completely agree (when it comes to programming, I don't agree that reading books is boring).

But let's be fair here, there is no real "best way to learn Python". There are many different ways and you just have to find the one that works best for you personally.

Folks - April 21, 2017, 8:34 am

Agreed with sapience, you can't just formulate the best way to learn something. The human brain isn't like the computers we are working with in that we know everything about it to an extent. Each person learns differently so there is no real need to come up with "The best way to learn python" aside from helping the people who have tried other ways and couldn't.

For me, I think attempting to make something you want and then googling everything you don't know until you do know is the way to go because after a certain point, you won't be googling everything and you'll get practice in what you want to do.

Malicious Intent - April 21, 2017, 8:40 am

(Apr 19, 2017 08:12 AM)CaptainBlack Wrote: [\[CITE\]](#) You need to combine both imho. A good textbook will ensure that you learn proper coding habits and pick a programmer's mindset, but it's also trivial to apply what you learn by coming up with a project of your own. Agree with this fella .. if you just start copy paste everything you google then you will never learn the concepts.. reading a good book or going through course will speed up the learning process in the long term and teach you good habits as well as all the do's and don'ts .

Sapience - April 21, 2017, 8:43 am

(Apr 21, 2017 08:40 AM)Malicious Intent Wrote: [\[CITE\]](#) Agree with this fella .. if you just start copy paste everything you google then you will never learn the concepts.. reading a good book or going through course will speed up the learning process in the long term and teach you good habits as well as all the do's and don'ts .

Where did you see anyone in this thread claim that copy pasting can teach you anything?

There's a huge difference between making something yourself using Python's own documentation

to then go on and compare your code to other programmers' code and just copy pasting everything.

Malicious Intent - April 21, 2017, 8:50 am

(Apr 21, 2017 08:43 AM)Sapience Wrote: [\[CITE\]](#) Where did you see anyone in this thread claim that copy pasting can teach you anything?

There's a huge difference between making something yourself using Python's own documentation to then go on and compare your code to other programmers' code and just copy pasting everything.

If you will try to create a project without any basic knowledge of syntax and concepts you will end up copy pasting everything as you won't understand how it works. Documentation can be tricky at times as you will probably realize soon enough.

Folks - April 21, 2017, 8:53 am

(Apr 21, 2017 08:50 AM)Malicious Intent Wrote: [\[CITE\]](#) If you will try to create a project without any basic knowledge of syntax and concepts you will end up copy pasting everything as you won't understand how it works. Documentation can be tricky at times as you will probably realize soon enough.

If the whole c+p thing was in response to me, I don't mean google stuff to copy paste. When I started out by googling everything I normally check out SO, documentation, or examples and learn from that.

When I started on PyQt I copied and pasted some skeleton code and modified it to what I wanted. Now I could do the whole thing from scratch.

Malicious Intent - April 21, 2017, 8:58 am

(Apr 21, 2017 08:53 AM)Folks Wrote: [\[CITE\]](#) If the whole c+p thing was in response to me, I don't mean google stuff to copy paste. When I started out by googling everything I normally check out SO, documentation, or examples and learn from that.

When I started on PyQt I copied and pasted some skeleton code and modified it to what I wanted. Now I could do the whole thing from scratch.

Everyone gotta start somewhere... my point was that going through the book or course will save you lots of time.

Sapience - April 21, 2017, 8:59 am

(Apr 21, 2017 08:50 AM)Malicious Intent Wrote: [\[CITE\]](#) If you will try to create a project without any basic knowledge of syntax and concepts you will end up copy pasting everything as you won't understand how it works. Documentation can be tricky at times as you will probably realize soon enough.

What are you trying to say by "as you will probably realize soon enough"?

I already know that, and at that point you try stuff out and see how it works from there.

(Apr 21, 2017 08:58 AM)Malicious Intent Wrote: [\[CITE\]](#) Everyone gotta start somewhere... my point was that going through the book or course will save you lots of time.

It may save you some time, but in my opinion that's not really worth it when you consider the amount of practice in figuring things out by yourself and sorting out bugs you will get from not following any book or course.

Malicious Intent - April 21, 2017, 9:01 am

(Apr 21, 2017 08:59 AM)Sapience Wrote: [\[CITE\]](#) What are you trying to say by "as you will probably realize soon enough"?

I already know that, and at that point you try stuff out and see how it works from there.

Thats the whole difference in time. Trial and error can sure teach you things but the same can be achieved going by the book.. the difference is you dont have to waste time on errors.

Sapience - April 21, 2017, 9:03 am

(Apr 21, 2017 09:01 AM)Malicious Intent Wrote: [\[CITE\]](#) Thats the whole difference in time. Trial and error can sure teach you things but the same can be achived going by the book.. the difference is you dont have to waste time on errors.

Fixing errors is a really valuable skill, though, so I don't see it as time wasted but rather as practice.

Malicious Intent - April 21, 2017, 9:38 am

(Apr 21, 2017 09:03 AM)Sapience Wrote: [\[CITE\]](#) Fixing errors is a really valuable skill, though, so I don't see it as time wasted but rather as practice.

Those same skill you gain while trying to figure out why your code fails you can learn from the book which will make you understand it instantly. There is no shortcuts in life man. You go to school and start from the first class not the highest one for a reason... sure you can follow along and learn as you go but you will always be missing basics which will cause your code to fail constantly and result in waste of time right through out your career. I understand that books and courses get boring but they are there for a reason.. they are not meant to make you a developer straight away, just teach you to think like a developer. Your method might work for you and for many others but its not advised so don't force your method on everyone. You keep your defensive attitude all you want. Im not telling you to open a book just asking not to inspire people to take shortcuts.

Sapience - April 21, 2017, 10:06 am

(Apr 21, 2017 09:38 AM)Malicious Intent Wrote: [\[CITE\]](#) Those same skill you gain while trying to figure out why your code fails you can learn from the book which will make you understand it instantly. There is no shortcuts in life man. You go to school and start from the first class not the highest one for a reason... sure you can follow along and learn as you go but you will always be missing basics which will cause your code to fail constantly and result in waste of time right through out your career. I understand that books and courses get boring but they are there for a reason.. they are not meant to make you a developer straight away, just teach you to think like a developer. Your method might work for you and for many others but its not advised so don't force your method on everyone. You keep your defensive attitude all you want. Im not telling you to open a book just asking not to inspire people to take shortcuts.

"don't force your method on everyone"

I'm sorry, did I not say multiple times in this thread that the actual best method is different from person to person and you should figure out what works best for you personally? Please don't put words into my mouth.

This method of learning a programming language is actually far from a shortcut: it'll take you longer to get started initially, but over time you'll actually do better.

JustAnotherRandomDude - April 23, 2017, 9:13 pm

For the calculator I came up with:

Code # Basic Calculator

```
import operator
operations = {
    "+": operator.add,
    "-": operator.sub,
    "*": operator.mul,
    "/": operator.div,
    "%": operator.mod
}
```

```
print "Welcome to the calculator. \n Please enter the first number: \n"
```

```
num1 = input()
print "\n Please enter the operator: \n"
```

```
operator = raw_input()
calculate = operations[operator]
```

```
print "\n Please enter the second number \n"
num2 = input()
```

```
result = calculate(num1,num2)
```

```
print "\n The result is: " + result
```

I'd like to give credit to <http://stackoverflow.com/a/2983144> for recommending the operator module and providing an example so I could work this into the final program.

Sapience - April 24, 2017, 8:28 am

(Apr 23, 2017 09:13 PM)JustAnotherRandomDude Wrote: [\[CITE\]](#) For the calculator I came up with:

Code # Basic Calculator

```
import operator
operations = {
    "+": operator.add,
    "-": operator.sub,
    "*": operator.mul,
    "/": operator.div,
    "%": operator.mod
}
```

```
print "Welcome to the calculator. \n Please enter the first number: \n"
num1 = input()
print "\n Please enter the operator: \n"
```

```
operator = raw_input()
calculate = operations[operator]
```

```
print "\n Please enter the second number \n"
num2 = input()
```

```
result = calculate(num1,num2)
```

```
print "\n The result is: " + result
```

I'd like to give credit to <http://stackoverflow.com/a/2983144> for recommending the operator module and providing an example so I could work this into the final program.

Looks pretty good! :D

Nice work