

Career Track • Curriculum

# Python Programmer|

Qualification for the Job Role  
of a **Python Programmer**





## Why this course?

**Python is the no. 1 programming language** for machine learning and data science and is relatively easy to learn, even for beginners. Whether you want to develop web applications, websites or APIs - if you want to change to a career in software development successfully, you can't do it without Python. IT professions across all industries are among the most sought-after in Europe and offer a wide range of career and development opportunities. Software development in particular is the perfect entry point, so that you have the best conditions on the job market in the coming decades and can specialize professionally in other fields, such as full stack development or data security.

As a Python programmer you will develop software alone or collaboratively in a team. You will apply design principles to write efficient code that is easy to maintain in the future. Whether it's a product or processes in the background, software accompanies all aspects of our day-to-day work. As a Python programmer, you will be responsible for gathering relevant information for software development, performance capabilities, and providing high-performance interfaces within the company.

**With the certified online course to become a Python Programmer, you will enter the world of software development with Python.** You will learn the basics of the Python programming language and familiarize yourself with the most important programming concepts such as variables, types, functions and methods. Based on this, you will independently develop and extend classes and modules. You will perform unit tests and ensure the functionality of your code.

You will work with the popular Python Standard Library and explore advanced concepts of object-oriented programming such as inheritance and composition. You will finish the program with a business-relevant final project to configure a password manager, and upon successful completion of the career path, you will qualify for a job role as a Python Programmer, Python Developer, or Software Developer.

**Employers in Germany are currently looking for more than **96,000** IT-specialists. Software developers and programmers are among the most sought-after job roles.**

## Course content.

- Processing data and text in Python
- Working with selected Python libraries
- Practice project to automate work processes
- Object-oriented programming (OOP) with a focus on classes and attributes
- Using inheritances for code reusability
- Get to know more advanced functions for simplifying classes
- Final project to configure a password manager



## At a glance.

Duration

**72 hours (4 months)**

Type

**Online course**

Structure

**2 modules + 1 final project**

Language

**German, English**

Level

**Entry level**

Completion

**Certificate of completion**

Prerequisites

**Placement test, basic knowledge mathematics & statistics.**

Target group

**The Python Programmer course is suitable for anyone who wants to learn Python as a programming language and use it professionally. You should have an enthusiasm for logical thinking and solving complex problems.**

**The further training is suitable for career changers and the perfect entry into software development to become a Python developer, software developer or data scientist.**



# Module overview.

## Module 1

### Python basics

#### Chapter 1: Data types

In this chapter you will navigate through our programming environment - the Data Lab - for the first time and execute your first code commands. You will learn what data or text represents in Python. Using a set of rules, you will learn how to create, assign and test data in **variables**. You will then learn how to read out common **error messages** and practice how to use them productively in your day-to-day work.

You will also learn basic **Python standard functions**, such as `type()` or `str()` and use them in application examples. After you've learned about the **if statement** and can use it to control the flow of your code with **conditions**, you will finish the chapter with the first part of a two-hour mini-project: You will program a user interface that reacts flexibly to the user's input.

#### Chapter 2: Flow control

In the second chapter you will work on two essential operations to make your code even more flexible: You will learn about **lists and for-loops**. Lists allow more flexibility in storing data and are a prerequisite for advanced programming. You will learn to create them, read them and change them purposefully.

You will use lists to extend the functionality of your user interface from the first chapter and complete the first mini-project. Afterwards you will work with for-loops, which you can use to automatically execute your code several times and level up your programming.



# Module overview.

## Module 1

### Python basics

#### Chapter 3: Functions, modules and methods

In the third chapter, you will round off your programming skills in Python and learn some advanced techniques. These include **functions and methods**. You learn how to define your own functions to structure your code better. You will also combine the individual programming elements such as conditions, loops and functions in programs.

You will use various methods and learn how to import Python modules correctly and how to ensure functionality as a whole. Linked to this you will learn how to import and export data as a simple test. In a one-hour mini-project involving telephone data, you will consolidate what you have learned in the chapter.

#### Chapter 4: Python applications

In the fourth chapter, you will recap content from chapters 1-3 and round it off with additional material. You'll learn what **dictionaries** are and how they can make your code more efficient. You will get to know **complex data structures** that you will need for your final project.

In a four-hour hands-on project to create a complex **user interface** for automatically processing customer requests, you will need to bring together everything you learned in the previous chapters.





# Module overview.

## Module 2

### Object-oriented-programming

#### Chapter 1: Advanced Python

In the introductory chapter, you'll review the most important content from the previous module before learning a series of concepts that will catapult you to the next level of programming with Python. You will take a deeper look at defining functions and learn about **default values**, **type hints** and **assert statements**. Afterwards, you will be able to use functions even more effectively as tools for your projects.

You will also cover the concepts of **list comprehension** and **dictionary comprehension** to create lists and dictionaries efficiently. At the end of the chapter you will learn how to adapt your code to the industry standard PEP8 with layout and structure.

#### Chapter 2: OOP basics

In the second chapter you will learn what OOP is, which program principles are based on it, and which conclusions you can draw from it, using simple examples. In the main part of the chapter you explore how **classes and attributes** are defined and used. You will use examples to examine **instance methods** and how to use them and define them with **method chaining**.

You'll learn what the **self keyword** is, as well as how to distinguish **debugging from class definitions**. Finally, you will test your previous knowledge in an interactive intermediate project and repeat the exercises from the chapter.





# Module overview.

## Module 2

### Object-oriented-programming

#### Chapter 3: Inheritance and composition

In the third chapter, you will learn what **inheritance** and **composition** are and how to use these concepts in use cases. In addition to simple inheritance, you will also learn more advanced methods such as **inheritance hierarchy** and **multiple inheritance**. In doing so, you'll practice deep inheritance methods used in the reusability of data from parent to child classes, compensating for data loss.

Finally, we will provide you with the most important best practices for **unit tests**, so that you can detect errors in your code before your users find them.

#### Chapter 4: Advanced OOP

In the fourth chapter you will deal with more advanced terms of object-oriented programming, which will accompany you in your daily work. You will study how programs and modules differ and what role `__main__` plays. You will learn what **decorators** are and how to use **property decorators** optimally. You will look at static and class methods and what special methods and class representations can be used with `__str__()` and `__repr__()`.

Based on this, you will learn about the options for representation **operator overloading** offers as well as other important methods from the Python Standard Library and then apply the learned content in a company-relevant interim project.





## Module overview.

### Module 2

#### **Object-oriented-programming**

##### **Chapter 5: OOP applications**

In the fifth chapter, you will demonstrate your knowledge in two larger projects that represent classic use cases of object-oriented programming. In the first project, you'll build your own interface to a popular **data science library** that enables further uses for machine learning and data or text analytics.

The second project deals with programming your own **blockchain**, where you will learn more about the underlying concepts. By the end of Module 2, you'll be equipped to apply OOP in the corporate world.

### Module 3

#### **Final project**

As part of the final project, you will expand on what you learned in the Python Basics and Object-Oriented Programming modules and independently program a **password manager**.

To do this, you will set up a programming environment and, using a terminal, fill your file with records that create a set of rules that configure the password manager.



## About StackFuel.

We are your strategic learning partner, suitable for every career level and professional orientation.

Whether you're changing careers, an employee or a manager, our certified and state-supported training courses in data, AI and programming will keep you up to date with the latest technologies.

## Course schedule.



\*Sample course schedule for our Data Analyst course.

● Onboarding & offboarding ● Training contents ● Mentoring & support

## Training philosophy.



### Online & flexible

Do your course part-time or full-time and learn 100% online in your browser on your PC or laptop at home.



### Hands-on practice with real-life projects

In our Data Lab you will write your own algorithms with industrial data sets in interactive exercises and coding challenges.



### Mentoring & career coaching

Your personal mentoring team will accompany you with coaching, feedback, and weekly group webinars.



### Certified & eligible

As a certified training provider, you can get our courses fully subsidized by the job center and the employment agency.



## Facts.

**91%**  
completion  
rate

**80%**  
interactive  
exercises

**20%**  
expert videos  
& text lessons

**+150.000**  
learning hours in the Data Lab

**4,3 von 5**  
recommendation rate

## Sponsorship opportunities.

### Installments or part payment

Use our installments or part payment options to spread out the costs of your course over several months so you can remain financially flexible.

### Education voucher

With an [education voucher](#) (Bildungsgutschein in German), you can get your course financed up to 100% by the Jobcenter or the Employment Office, if you are currently out of work or looking for employment.

### Training opportunities act

If you are working, you can get your course partially or completely funded by your employer thanks to the [Training Opportunities Act](#) (Qualifizierungschancengesetz in German) - regardless of your qualifications, age or the size of the company.

### StackFuel scholarships

We regularly award various [scholarships](#) for our courses, to promote more diversity in the field of data. We want to encourage more people to take an interest in programming, and more women to work in data roles.

## Support & mentoring.

### Coaching

- > Assessment
- > Final project and evaluation
- > 1:1 project feedback consultation
- > Official certificate of completion

### Personal mentoring

- > Kick-off session
- > Webinars with other course participants
- > Support via email or phone
- > Online forum

### Career services

- > CV and application coaching
- > Talent pool and career intros
- > Data community
- > Career events



# Ready to enroll?

Get in touch with our **consultants**  
and build up the **skills** you need!

For Individuals



**Maria Schwenke**

Education Advisor

**Book free consultation**

For Businesses



**Ginesh Koottakara**

Head of Sales

**Book free demo**

August 2023

**Version:** 23/08/21

**Publisher:** StackFuel GmbH

**Web:** [www.stackfuel.com](http://www.stackfuel.com)

**Email:** [info@stackfuel.com](mailto:info@stackfuel.com)

**Phone:** +49 (0)30 544 533 420

Errors and omissions excepted.

© 2023 StackFuel GmbH