

BootUP

FOUNDATION COURSE PROGRAM

# CURRICULUM

 Python Programming

*No prior experience with programming is required. You will need to be comfortable with basic computer skills, such as managing files, running programs, and using a web browser to navigate the Internet. You will need to be self-driven and genuinely interested in the subject. No matter how well structured the program is, any attempt to learn programming will involve many hours of studying, practice, and experimentation. This requires some tenacity, and it is especially difficult to do if you don't find the subject interesting or aren't willing to play around and tinker with your code —so drive, curiosity, and an adventurous attitude are highly recommended!*

## Educational Objective

This introductory Foundation program teaches the foundational skills all programmers use with Python language, whether they program mobile apps, create web pages, or analyze data. It is ideal for beginners who want to learn new skills, make informed choices about career goals, and set themselves up for success in career-track Foundation programs.

<b>Length of Program*</b>	: 6 Hours
<b>Textbooks required</b>	: None
<b>Instructional Tools Available</b>	: Video Lectures, Practice Assignment, Student and Mentorship Group Chat, Q&A Consultation

## Our Instructor



### Mohammad Gharachorloo

Graduated in the field of telecommunication and networking from Queensland University of Technology (QUT). More than 300 hours of teaching experience in Python Programming for elementary and advanced students. Currently working on AI-Driven projects such as implementing smart call centers using Natural Language Processing (NLP) techniques. Head programming supervisor at Zharfiran Co, an Iranian R&D group with its main focus centered around machine-learning applications in various fields.

## What You Will Learn

- ✔ Have a fundamental understanding of the Python programming language.
- ✔ Add the Python Object-Oriented Programming (OOP) skills to your résumé.
- ✔ Have the skills and understanding of Python to confidently apply for Python programming jobs.
- ✔ Understand how to create your own Python programs.
- ✔ Acquire the prerequisite Python skills to move into specific branches - Machine Learning, Data Science, etc..
- ✔ Learn Python from experienced professional software developers.

*\*The length is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. Actual hours may vary.*

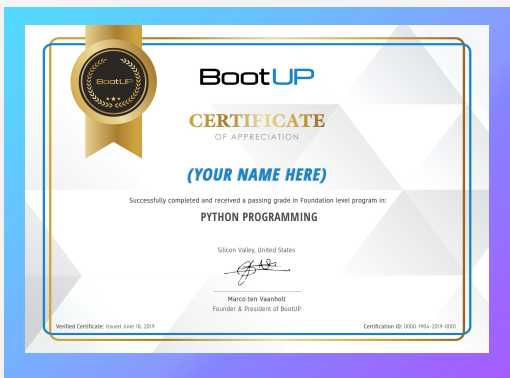
## The Benefits of Further Education

A **Foundation Program** from BootUP could be ideal for anyone looking to:

- Pursue promotion in their current line of work
- Explore an entirely new career path in Technology Industry
- Brush up on existing knowledge, skills and professional traits
- Pave the way of further studies at a much higher level
- Start your own business from scratch or setup as a freelancer

The right qualification has the potential to transform your career prospects and education alike. BootUP believes that this learning should be open to anyone with an interest in improving their prospects.

## Our Certification



BootUP is proud to offer you the credible and respected learning certification with World-Class Experts around the world through our **Foundation, Flex and Full-Immersive Course Program**.

With a very affordable cost and no entry requirements whatsoever, anyone interested in further education has the incredible chance to study and get the World-Class certification.

## Certification Requirement

There are tests throughout the modules. Each module contains exercises and scenarios for students to answer and self-assess their learning. Students are required to complete all content elements in each course and earn at least a 80% average test score on to earn their certificate. Students will have unlimited attempts at all graded tests.

## Enrol Today

We make enrolment as quick and easy as possible. Simply pay a visit to our program page, or get in touch with our admissions team at any time for more information. Enrolment is open throughout the year and there are no deadlines or time restrictions. Study in your own time and at your own pace, with helpful advice from our team whenever you need us.

## Ask Our Team

For more information on any of our program courses or to discuss enrolment, get in touch with the BootUP team today on **Call or WhatsApp at (+62)-811-992-1500**, or drop us an **email anytime at info@bootup.ai**.

## MODULE 1

### INTRODUCTION TO COMPUTER PROGRAMMING

This module mainly focuses on the definition of programming from a general perspective. In this module you will learn about different concepts and terminologies such as algorithms, problem solving, programming paradigms and programming languages.

NO	TOPICS	DURATION (in minutes)
1.1.	<b>What is a computer system?</b> <ul style="list-style-type: none"> <li>• Hardware components</li> <li>• Software components</li> </ul>	10
1.2.	<b>What is programming?</b>	5
1.3.	<b>Computer programming</b> <ul style="list-style-type: none"> <li>• Algorithms and Algorithmic thinking</li> <li>• Problem solving</li> </ul>	10
1.4.	<b>Programming paradigms</b> <ul style="list-style-type: none"> <li>• What is a paradigm</li> <li>• Functional programming</li> <li>• Procedural programming</li> <li>• Object oriented programming</li> </ul>	10-15
1.5.	<b>Programming languages</b> <ul style="list-style-type: none"> <li>• What is a programming language</li> <li>• Levels of programming languages</li> <li>• Scripting languages</li> <li>• Compiler vs. Interpreter</li> </ul>	12
1.6.	<b>Introduction to IDEs</b> <ul style="list-style-type: none"> <li>• Pycharm</li> <li>• Jupyter</li> <li>• Spyder</li> </ul>	15
<b>Total Online Learning in Minutes</b>		<b>62 - 67</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*

## MODULE 2

### PYTHON INTRODUCTION AND BASIC DATA TYPES

The main objective of this module is to get you started with Python programming language, starting from a brief introduction to this language and leading to your first project where you will learn about basic data types such as ints, floats and strings.

NO	TOPICS	DURATION (in minutes)
2.1.	<b>History of Python</b>	5
2.2.	<b>Python in action</b> <ul style="list-style-type: none"> <li>Distinguishing features of Python</li> <li>How famous is Python?</li> <li>Where can I use Python (or for what projects can I use Python?)</li> </ul>	10 - 15
2.3.	<b>Installing python interpreter</b>	8
2.4.	<b>Getting started with Pycharm IDE</b>	5
2.5.	<b>Basic data types - part 1</b> <ul style="list-style-type: none"> <li>Numbers in Python (int, float)</li> <li>Arithmetic operations</li> <li>The concept of variables</li> <li>Print command</li> </ul>	15
2.6.	<b>Basic data types - part 2</b> <ul style="list-style-type: none"> <li>String and its methods</li> <li>String to number conversion</li> <li>Input command</li> <li>Boolean and logical operators</li> </ul>	20
<b>Total Online Learning in Minutes</b>		<b>63 - 68</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*

## MODULE 3

### COLLECTION DATA TYPES

Collection data types are one of the major and most important topics in any programming language. In this module, you will learn about such data types used in Python. You will also learn about what sort of operations can be done on each type.

NO	TOPICS	DURATION (in minutes)
3.1.	<b>Lists</b> <ul style="list-style-type: none"> <li>List concept</li> <li>List manipulation and methods</li> <li>List-to-string conversion</li> </ul>	20
3.2.	<b>Tuples</b> <ul style="list-style-type: none"> <li>Concept of mutability</li> <li>Tuple features</li> </ul>	10
3.3.	<b>Dictionaries</b> <ul style="list-style-type: none"> <li>Concept and methods</li> </ul>	10 - 15
3.4.	<b>Sets and frozensets</b> <ul style="list-style-type: none"> <li>Concept of a set</li> <li>Set operations</li> </ul>	10
3.5.	<b>Summary of collection data types</b>	10
<b>Total Online Learning in Minutes</b>		<b>60 - 65</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*

## MODULE 4

### CONTROL STRUCTURES

This module entails a crucial topic called “Control Structures”. This module starts with conditional statements by reviewing logical expressions for boolean values. Loops are another type of control structures covered in this module. You will learn about different type of loops and recommendations on where to use them each.

NO	TOPICS	DURATION (in minutes)
4.1.	<b>Conditional statements</b> <ul style="list-style-type: none"> <li>Logical statements</li> <li>If, elif and else expression</li> <li>Nested if</li> </ul>	15
4.2.	<b>Loops</b> <ul style="list-style-type: none"> <li>Concept of loops</li> <li>Types of loops</li> </ul>	8 - 10
4.3.	<b>For loop</b> <ul style="list-style-type: none"> <li>Iterability</li> <li>Stack variables</li> <li>Enumerate and zip commands</li> <li>Unpacking + for loop</li> <li>Comprehension</li> </ul>	25
4.4.	<b>While loop</b>	15
<b>Total Online Learning in Minutes</b>		<b>63 - 65</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*

## MODULE 5

### FUNCTIONAL PROGRAMMING

This module discusses one of the most prominent structures in programming (i.e. functions). You will learn why we build functions, how we do that and how we put those functions into use. You will also learn about collection of functions which leads to the concept of module and package in Python.

NO	TOPICS	DURATION (in minutes)
5.1.	<b>Dry principle and the concept of a function</b>	8 - 10
5.2.	<b>Functions in action</b> <ul style="list-style-type: none"> <li>• Defining a function</li> <li>• Local variables</li> <li>• Input/output arguments</li> <li>• Lambda functions</li> </ul>	25
5.3.	<b>Modules and packages</b> <ul style="list-style-type: none"> <li>• What is a module and a package?</li> <li>• Standard library of python</li> <li>• Introducing some modules for different areas</li> <li>• How to install modules through pip</li> </ul>	20
5.4.	<b>File operation</b>	10
<b>Total Online Learning in Minutes</b>		<b>63 - 65</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*



## MODULE 6

### OBJECT ORIENTED PROGRAMMING (OOP)

Object Oriented Programming is an programming paradigm in Python offering a standard philosophy of coding. In this module you will learn about different terminologies related to OOP along with how to use this paradigm to implement your code.

NO	TOPICS	DURATION (in minutes)
6.1.	<b>Definition and concept</b>	10
6.2.	<b>Terminology</b> <ul style="list-style-type: none"> <li>• Class</li> <li>• Object or instance</li> <li>• Instantiation</li> <li>• Attributes</li> <li>• Inheritance</li> <li>• Polymorphism</li> </ul>	15 - 20
6.3.	<b>Constructing and destructing</b> <ul style="list-style-type: none"> <li>• Class attributes vs. Object attributes</li> <li>• Static methods</li> </ul>	20
6.4.	<b>A simple example</b>	15 - 20
<b>Total Online Learning in Minutes</b>		<b>60 - 70</b>

*\*Please be advised that the time flow breakdown provided in the table is only an estimate.*

BootUP

SILICON VALLEY LEARNING EXPERIENCE  
WITH WORLD-CLASS EXPERTS

# Discover your potential NOW.

JOIN OUR PYTHON PROGRAMMING CLASS NOW



For more information about courses,  
and to join our complete programs, visit [www.bootup.ai](http://www.bootup.ai)

# BootUP

## **INDONESIA**

District 8 - Prosperity Tower, 9th Floor, SCBD  
Jl. Jend. Sudirman No. 52-53  
South Jakarta, 12190, Indonesia  
+62 811 992 1500  
info@bootup.ai

## **UNITED STATES**

68 Willow Rd  
Menlo Park, CA 94025  
Silicon Valley, USA  
+1 800 493 1945  
info@bootupventures.com