Automate the Boring Stuff

# Who is this course for?

On its own, this book won’t turn you into a professional software developer any more than a few guitar lessons will turn you into a rock star. But if you’re an office worker, administrator, academic, or anyone else who uses a computer for work or fun, you will learn the basics of programming so that you can automate simple tasks such as these:

* Moving and renaming thousands of files and sorting them into folders
* Filling out online forms—no typing required
* Downloading files or copying text from a website whenever it updates
* Having your computer text you custom notifications
* Updating or formatting Excel spreadsheets
* Checking your email and sending out prewritten responses

These tasks are simple but time-consuming for humans, and they are often so trivial or specific that there’s no ready-made software to perform them. Armed with a little bit of programming knowledge, however, you can have your computer do these tasks for you. Example:

➊ passwordFile = open('SecretPasswordFile.txt')  
➋ secretPassword = passwordFile.read()  
➌ print('Enter your password.')  
   typedPassword = input()  
➍ if typedPassword == secretPassword:  
   ➎ print('Access granted')  
   ➏ if typedPassword == '12345':  
       ➐ print('That password is one that an idiot puts on their luggage.')  
  else:  
   ➑ print('Access denied')

You might not know anything about programming, but you could probably make a reasonable guess at what the previous code does just by reading it. First, the file SecretPasswordFile.txt is opened ➊, and the secret password in it is read ➋. Then, the user is prompted to input a password (from the keyboard) ➌. These two passwords are compared ➍, and if they’re the same, the program prints Access granted to the screen ➎. Next, the program checks to see whether the password is 12345 ➏ and hints that this choice might not be the best for a password ➐. If the passwords are not the same, the program prints Access denied to the screen ➑.

# What is Python?

*Python* is a programming language (with syntax rules for writing what is considered valid Python code) and the Python interpreter software that reads source code (written in the Python language) and performs its instructions.

Chapter 1 – Python Basics

REPL – Interactive shell which stands for Read Evaluate Print loop.

Graphical user interface, text, application

Description automatically generated

Figure 1 - First Python expression

2 + 2. 2 is a value while + is an operator. Expression will contain values and operators, which will result in an evaluation. Just typing 4 on it’s own and pressing Enter will also result in an evaluation, where 4 simply evaluates itself to 4.

Graphical user interface, application

Description automatically generated

Figure 2 - Math Operators from highest to lowest precedence

Precedence refers to the order of operations similar to mathematics.

# String Operations

**String example**: >>> ‘Hello, world!’

## String Concatenation and Replication

‘Alice’ + ‘Bob’ = AliceBob

In Python, it is possible to perform a **string replication** operation as follows:

‘Alice’ \* 5 will result in Alice printed five times.

# Variables

>>> spam = 40

>>> spam

40

A variable is *initialised* the first time a value is stored in it.

Graphical user interface, application

Description automatically generated

Figure 3 - Valid and Invalid Variable Names

Variables are case sensitive.

Continue from Hello World - <https://automatetheboringstuff.com/2e/chapter1/>