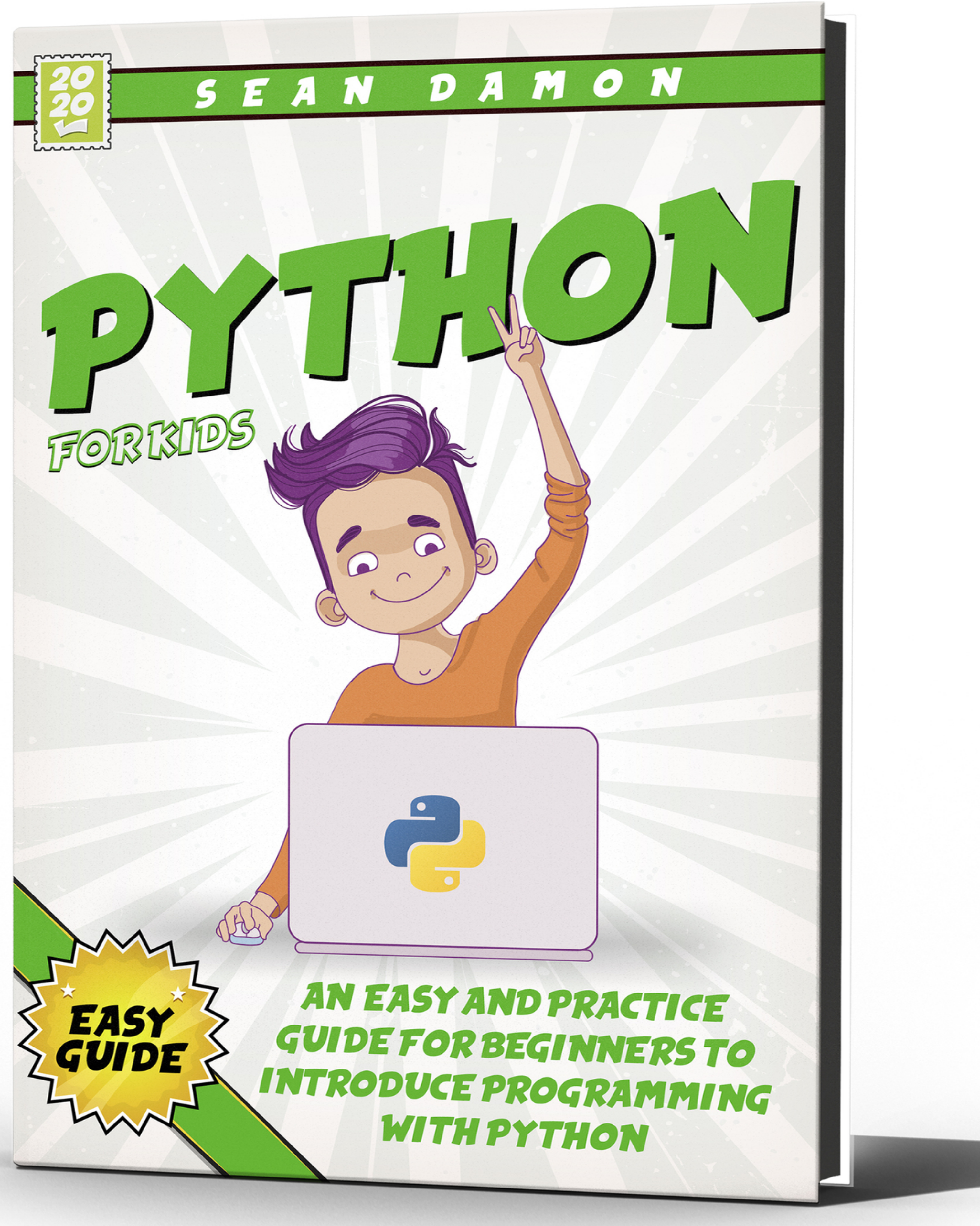
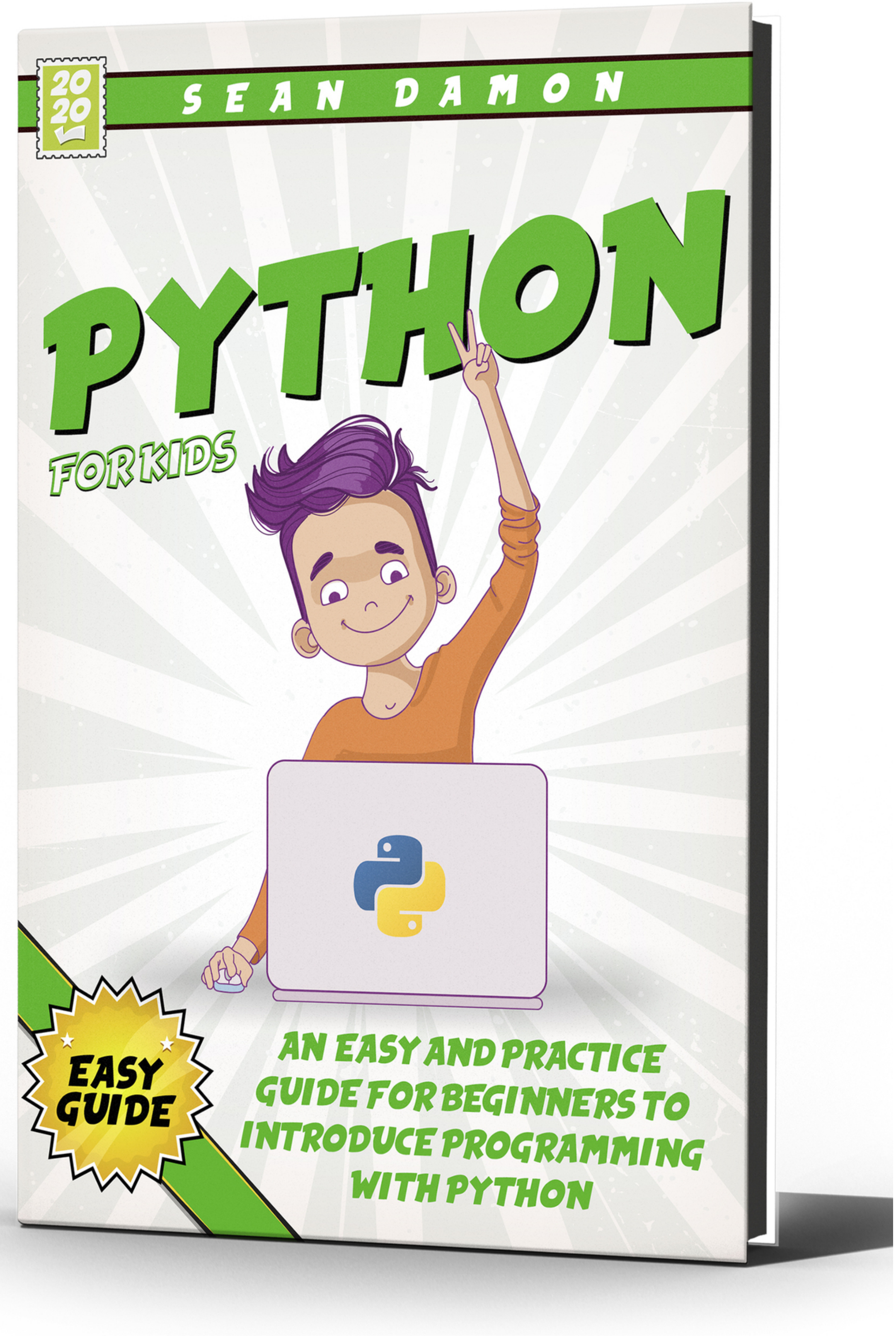


SEAN DAMON

PYTHON

FOR KIDS

AN EASY AND PRACTICE GUIDE FOR BEGINNERS TO INTRODUCE PROGRAMMING WITH PYTHON



2020

SEAN DAMON

PYTHON

FOR KIDS

EASY GUIDE

AN EASY AND PRACTICE GUIDE FOR BEGINNERS TO INTRODUCE PROGRAMMING WITH PYTHON

PYTHON FOR KIDS

An Easy And Practice Guide For Beginners To
Introduce Programming With Python

SEAN DAMON

Legal Notice

Disclaimer Notice

Please note the information contained within this document is for educational and entertainment purposes only. All effort has been executed to present accurate, up to date, and reliable, complete information. No warranties of any kind are declared or implied. Readers acknowledge that the author is not engaging in the rendering of legal, financial, medical or professional advice. The content within this book has been derived from various sources. Please consult a licensed professional before attempting any techniques outlined in this book.

By reading this document, the reader agrees that under no circumstances is the author responsible for any losses, direct or indirect, which are incurred as a result of the use of information contained within this document, including, but not limited to, — errors, omissions, or inaccuracies.

Table of Contents

Introduction

Coding in Python is straightforward and promises a fun experience that beats basic routine programming. Below are some benefits of why Kids should start learning on how to code with Python:

Easier and Better Debugging

Debugging refers to a process of searching for bugs and defects and resolving them. These bugs and errors tend to inhibit a program or system from working correctly. In this vein, debugging aims to resolve problems in programs. In terms of debugging, Python is one of the highest-ranking programming languages owing to the tools it provides, such as PyDebug, pudb, and pdb, which help make the debugging experience smooth and manageable.

Take, for instance, the tool—pudb, can help programmers analyze the core of codes and track down a problem. When compared, while other languages keep working to improve their debugging experiences and tools, Python only gets better with time.

There Are No Restrictions to its Programming Syntax

Unlike other programming languages, Python can be easily understood by anyone, even up to a non-programmer. The reason is primarily attributed to its high readability and problem-oriented design, which lets one pay attention to coding logically rather than expressly delving into the nitty-gritty of the syntax.

As proof of this unique benefit, consider the sample program shown below to calculate a simple interest.

```
Print (‘Simple Interest Calculator :’)
Amount = float (input (‘Value of Principal?’))
roe = float(input(‘Rate of Interest ?’))
time = int(input(‘Duration (no. of years) ?’))
```

```
total = (amount * pow(1 + (roi/100), time)
interest = total - amount
print(‘\nInterest = %0.2f’ %interest)
```

Taking a look at the program, you can note how relatively easy it is to decode and write the code in Python.

An Explicit Declaration is not Required

When coding in Python, type specifiers should not be used in the declaration of variables because it can be done without declaring any of the data types in Python. Additionally, it is unnecessary to make use of separators, such as semicolons (;) in marking the end of a statement or command. Indentation in Python takes the place of parenthesis in grouping a block of code. And to make indentations in codes, you can make use of either spaces or tabs. Albeit, some rules are enforced in Python, like the use of four spaces for each level of indentation, many other facets of Python, such as this contribute to making learning easier for beginners.

Great Object-Oriented Programming Support

OOP (Object-Oriented Programming) comes as a built-in element in Python. It creates a model that produces solutions by devising objects, binding data, and defining connections. Albeit, the procedural programming methodology takes a different approach from top to bottom, solving one problem at once and splitting it into bit sizes. Object-Oriented Programming, on the one hand, is a different ball game altogether. It adopts a bottom to top problem-solving approach that looks for blueprints to the solution from the onset, thereby leaving implementation to a much later time.

To develop you as a professional Python programmer, constructing objects, building inheritance, and using classes are fundamental approaches to take. What’s more, is that these concepts can be easily mastered in Python for the production of quality programs with quicker implementations than other programming languages. Python encourages minimalism in code sizes, allowing for faster and better applications.

CHAPTER 1:

What is Coding

When we talk about the word coding, we will quickly see that it is a vast and big world to look at. It is not going to include just one or two parts. It is going to include hundreds of languages, thousands of software programs, and so much more. For example, all of the software that you are going to use on your desktop and laptop, and even the games and the apps that you enjoy on your smartphone are all going to be products of this coding. Even if you are working with a gadget or a device that does not have a screen, such as an RC drone, or even the toy Furby, you will not be able to behave in the proper manner without some of the software that coders have been able to develop.

That is why we are going to get started on some of the different parts that are going to show up when it is time to work with the idea of coding and programming. No matter what kind of coding or programming language you are looking to get into, there are always a bunch of options, and figuring out how they all work, and the basics that they will all share, can make a difference in the amount of success you are going to see.

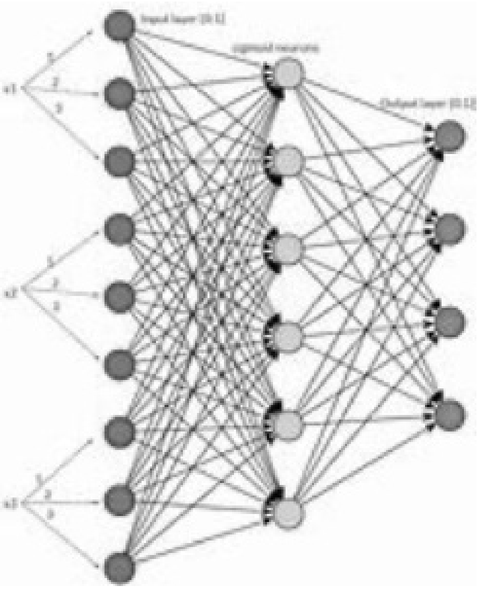
The first thing that we need to do is to explore the most common terms and the definitions that go with them in Coding. No matter what kind of code you would like to do, or the coding language that you want to choose, knowing these basics can be important. Some of the terms that you need to know to get started will include:

Program and Code

Code is going to refer to the set of instructions that you can write out for the computer, or your compiler, to follow. As soon as you made a new code on a computer, and you wrote it out in a manner that the computer can process from the beginning all the way to the end without an error, you can already compile it into the program that you want.

Each programming language is going to come with its own rules when it comes to how the code should be written, and how you will work with each part of the program. Just know for now that the code is going to be the instructions that you are able to send through to the compiler, telling it how you want that program to behave.

Algorithms

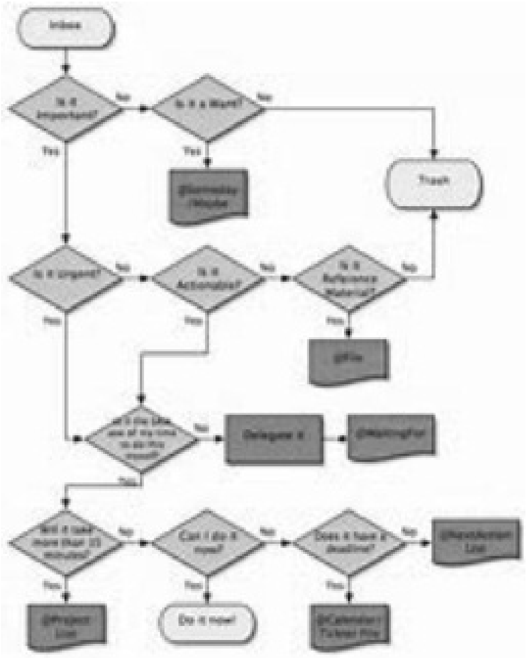


Even as a beginner, it is a good idea for you to learn a bit more about flowcharts and algorithms, even before you get really into some of the other complex stuff. When people code, they are working on providing a good solution to a problem, and they have to do this in a manner that the computer is able to understand. But if you spend too much of your time trying to craft each line of code without having a clear objective, you may find that you also have to spend a lot of time editing the code when you find out that the solution you came up with did not really address the problem, or at least not all of the situations of the problem.

Remember here that programs can be really complex to work with. According to Wired, in 2015, all of the services that Google provided over the internet were powered by 2 billion lines of code. Even some of the easier and more simple options that we would see with code, like the Windows 10 Calculator, comes with over 35,000 lines of code. Even with all of this coding, we have to remember that we can’t get through all of this or make it work well if we don’t go through with a game plan right from the beginning, and this is where the algorithms are going to come in to help.

The neat thing that you will notice about these algorithms is that you can do some work with them, even if you are not familiar with coding in the first place. You don’t even need to work with a computer, though this does add an element of ease to the process. An algorithm is similar to what we are going to find with the code we talked about before, but it is going to be done in a manner that humans, not just the computer can understand, and it helps to give us a game plan of how to handle some of the more complex types of codes that we are going to work with.

Flowcharts



If this works, there should be some cookies that will come to your chosen website. You can then use the cookies that you have collected. You are able to use the information from the cookies, which should be saved to the website of your choice, for whatever purpose you need.

A good way for us to think about these flowcharts is as a graphical representation of an algorithm. Since these algorithms are simply just a translation of code into a human-friendly manner, a flowchart is going to provide us with more of a bird’s eye view of the functionality of the code from start to finish. If you have ever done or even seen a flowchart to help see how the workflow should go, you will find that the flowcharts for coding work in a similar manner.

Flowcharting is useful in many cases because it is going to help us to make some of the more complicated programs that we need to handle significantly easier to understand. And it helps us to illustrate these complicated programs before we proceed through the coding, ensuring that we find the flaws that we want, and can make it so that we see additional features where they need to be. However, as a beginner, the best place to start is to learn about the five flowchart symbols that are going to be used in any coding flowchart that you want to work with. These are going to include:

Oval Start and End

The oval, or the ellipse, is going to be a symbol that will tell us the beginning and the end of the flow of the program. To make the flowchart stay as simple as we can, you need to only use two of these ovals one for the beginning and one for the end. You can go with more than this though if your program is designed to have more than one ending.

Arrow

This is going to be the line that we are able to use to help connect the shapes of your flowchart together, with the arrowhead being used to tell us which direction the flow of the program is going. This is used to help eliminate the need for putting in numbered steps like you need to do when making one of the algorithms from before. Arrows add in some flexibility to the flow of your chart and can cut down on some of the clutter when the program is more complicated.
Parallelogram Input and Output: This is going to help us to see the operation for the input or the output. Whenever you would like to accept a value from the user, which is the input, or you would like to display a value, which is the output, in your program, make sure that the parallelogram shows up.

Rectangle the process

This is going to help us to see any operation that is performed internally by the computer. Basically, any operation that is direct, that doesn’t accept input or doesn’t display an output needs to be represented with this rectangle. For example, calculations similar to the modular division can use this kind of symbol.

Rhombus Decision

This is going to be a symbol that will be used to help branch out the flow of the program or help out when handling conditional statements. It is still going to count as a process because the computer will internally do the operation, but it has to decide and then pick out the right path based on the condition that you set.

These symbols are going to be recognized universally by programmers and others who do coding, so make sure that when you are using one of these charts, you are also working with the right symbols along the way.

CHAPTER 2:

Installation and Running of Python

Official Version Installation

Mac

Python is already pre-installed on the Mac and can be used directly. If you want to use other versions of Python, we recommend using the Homebrew installation. Open the Terminal and enter the following command at a command-line prompt, which brings you to Python’s Interactive Command Line:

```
$python
```

The Python input above is usually a soft link to a version of a Python command, such as version 3.5. If the corresponding version is already installed, you can run it in the following manner:

```
$python3.5
```

The terminal will display information about Python, such as its version number, followed by a command-line prompt for Python. If you want to exit Python, type:

```
>>>exit()
```

If you want to run a Python Program in your current directory, append the name to Python or Python 3:

```
$python installation.py
```

If the file is not in the current directory, you need to specify the full path of the file, such as:

```
$python/home/authorname/installation.py
```

We can also change the installation.py to an executable script. Just add the Python interpreter you want to use to the first line of the installation.py:

```
#!/usr/bin/env Python
```

In the terminal, change the installation.py to executable:

```
$chmod installation.py
```

Then, on the command line, type the name of the program file, and you're ready to run it using the specified interpreter:

```
$./installation.py
```

If the installation.py is in the default path, then the system can automatically search for the executable and run it in any path:

```
$installation.py
```

Linux

Linux systems are similar to MAC systems, and most come preloaded with Python. There are many Linux systems that offer something like Homebrew's software manager, which, for example, is installed under Ubuntu using the following command:

```
$sudo apt-get install Python
```

Under Linux, Python is used and run in much the same way as on the MAC, and I won't go into that again.

Windows Operating System

For the Windows operating system, you need to download the installation package from the official Python Website. If you don't have access to Python's website, search engines for keywords like "Python Windows download" to find other download sources. The installation process is similar to installing other Windows software. In the install screen, Customize the installation by selecting Customize, in addition to selecting Python components, check:

Add python.exe to Path

Once installed, you can open the Windows command line and use Python as you would on a Mac.

Other Python Versions

The official version of Python mainly provides compiler/interpreter functionality. Other unofficial versions have richer features and interfaces, such as a more user-friendly graphical interface, a text editor for Python, or an easier to use module management system for you to find a variety of extension modules. In unofficial Python, the two most commonly used are:

1) Anaconda

2) Thought Python Distribution (EPD)

Both versions are easier to install and use than the official version of Python. With the help of a module management system, programmers can also avoid annoying problems with module installation. So, it's highly recommended for beginners. Anaconda is free, and EPD is free for students and researchers. Because of the graphical interface provided, their use method is also quite intuitive. I strongly recommend that beginners choose one of these two versions to use. The exact usage can be found in the official documentation and will not be repeated here.

Virtualenv

You can install multiple versions of Python on a single computer, and using virtualenv creates a virtual environment for each version of Python. Here's how to install virtualenv using Python's included pip.

```
$pip install virtualenv
```

You can create a virtual space for a version of Python on your computer, such as:

```
$virtualenv –p /usr/bin/Python3.5 virtualPythonexample
```

In the above command, /usr/bin/Python3.5 is where the interpreter is located, and virtualPythonexample is the name of the newly created virtual environment.

The following command can start using the MYENV virtual environment:

```
$source virtualPythonexample/bin/activate
```

To exit the virtual environment, use the following command:

```
$deactivate
```

CHAPTER 3:

The Importance of Data Types and Variables

```
<li><a href="index.html">Home</a></li>
<li><a href="home-events.html">Home Events</a></li>
<li><a href="multi-col-menu.html">Multiple Column Menu on Larger Viewports
<li class="has-children"> <a href="#" class="current">Header Options</a>
  <ul>
    <li><a href="tall-button-header.html">Tall Button Header</a></li>
    <li><a href="image-logo.html">Image Logo</a></li>
    <li class="active"><a href="tall-logo.html">Tall Logo Image</
  </ul>
</li>
<li class="has-children"> <a href="#">Carousels</a>
  <ul>
    <li><a href="variable-width-slider.html">Variable Image Wi
    <li><a href="testimonial-slider.html">Testimonial Slider</
    <li><a href="featured-work-slider.html">Featured Work Sli
    <li><a href="equal-column-slider.html">Equal Column Slid
    <li><a href="video-slider.html">Video Slider</a></li>
    <li><a href="mini-bootstrap-carousel.html">Mini Slider
  </ul>
```

Now it is time for us to take a moment to look at some of the work that can be done with the data types and the variables that come with the Python language. Both of these are going to be important and will tell us a lot when it is time to work with the Python language as well. We are not going to get very far in some of the programmings that we want to do if we aren't able to learn some of the different types of data that are out there, and we aren't able to have a good understanding of how the variables work as well. Let's take a look at some of the different types of data that come with Python, and how these variables work, so we are able to put it all together and see some great results in the process.

The Python Data Types

The first thing that we need to focus on here is the data types that come with Python. Each of the values that we are going to work within Python is going to have a data type. Since everything in this programming language is going to be an object, the types of data are actually going to be classes, and then the variables are going to be instances, or objects, of these classes. This is how all of the work is going to be tied in together.

Now, you will find that there are various types of data that can show up in Python. Some of the ones that are the most important ones, and the ones that you are likely going to spend the majority of your time on in Python will be tackle below:

First on the list is the Python numbers. These are going to include floating-point numbers, complex numbers, and integers. These are going to be brought out with the help of the function of type (), which can help us to know which class a variable or our value is going to belong to. This helps us to make sure that we are able to pull out the numbers that we would like in this coding language.

We can also send out the Python list. This list is going to simply be an ordered sequence of items that we can work with. It is going to actually be the type of data that we are able to work within Python that is used the most, and it is very flexible. All of the items that are on the list do not need to end up being the same type of data either. In addition, the lists are going to be mutable, which simply means that the elements that are on the list can be altered in any way that you want.

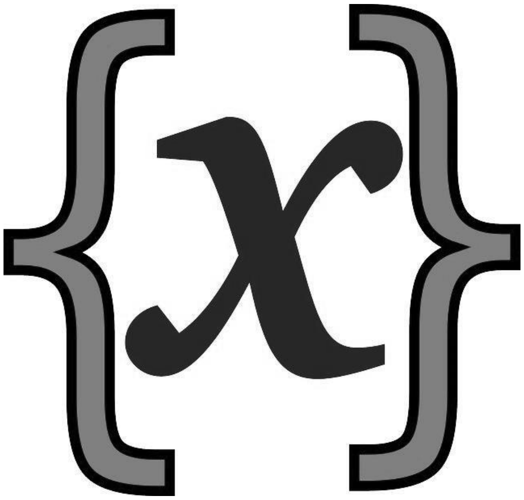
The Python tuple is also going to be important. The tuple is going to be an ordered sequence of items just the same as we see with the list. The biggest difference that you are going to see with these tuples is that they are immutable. Once you create one of these tuples, you are not able to make modifications to them. Tuples are going to be there to help us write/protect the data, and are usually going to perform faster than a list because it is not going to change dynamically at all.

The Python string is the next thing that we are able to focus some attention on. The string is basically going to be a sequence of Unicode characters. We are going to have the option of working with either a single quote or a double quote to help represent strings. You can also work with some multi-line strings.

The Python set is going to be another type of data that we are able to work with, and it is basically a collection of items that are unique and will not be ordered. The set is going to be defined by values that are separated by a comma, and then these items are going to be inside of braces. Items in a set are not going to be ordered at all.

And finally, we are going to take a look at the Python dictionary. The dictionary is basically going to be an unordered collection of key-value pairs. This is the one that is generally going to be used when we have to handle a very large amount of data. Dictionaries are going to be optimized to help us retrieve the data we want. We also have to make sure that we know the key before the data is going to be available. With the Python language, we are going to be able to define these dictionaries within braces, with each item being a pair in the form of key: value. You are able to have the key and the value of any type that you would like.

The Python Variables



The next topic that we need to spend some time on in this guidebook is going to be the variables. The variables are basically anything that is able to hold onto a value or one of the data types that we talked about from before, that can change. Often we are able to think about these variables as a box that is capable hold onto the values that we are using in Python. These are important for us to focus on because they will save some space on our memory, and then allow us to pull that variable back out at a later time.

These variables are something that all programmers need to learn about because they will basically be saved spaces in the memory of the computer, and then you are able to tell the computer that it is time to pull them out at any time when you are ready. The variables that you take the time to create are going to be found in certain locations in the memory of the system you are using, and then they can be found and executed when the code is running. Depending on the type of data that you would like to focus on, the variable is going to be the part that will tell the compiler where to save that information to make it available when needed.

One of the most important things that we need to focus on when it comes to working with these variables is making sure that we assign a value to it. If you do not assign one or more values to your variable, you are going to end up saving some empty space on your memory, and nothing is going to come up in the code when you pull it up. If the variable has some kind of value that has been assigned back to it, and sometimes it will have more than one value on it, then it is going to react in the manner that you would like in this code.

Any time that we are working on some of these variables, you will find that there are basically three main options that you are able to choose from for it. Each of these is going to be useful, and it is often going to depend on the codes you are trying to write, and the type of value that you are sending over to the variable. The options that you are able to pick from include:

- Float: this would include numbers like 3.14 and so on.
- String: this is going to be like a statement where you could write out something like “Thank you for visiting my page!” or another similar phrase.
- Whole number: this would be any of the other numbers that you would use that do not have a decimal point.

When you are working with variables in your code, you need to remember that you don’t need to take the time to make a declaration to save up this spot in the memory. This is automatically going to happen once you assign a value over to the variable using the equal sign. If you want to check that this is going to happen, just look to see that you added that equal sign in, and everything is going to work.

Assigning a value over to your variable is pretty easy. Some examples of how you can do this in your code would include the following:

```
x = 12#this is an example of an integer assignment
pi = 3.14#this is an example of a floating-point assignment
```

customer name = John Doe this is an example of a string assignment

Another option that you can do here, and that we have mentioned a little bit in this phase already, is a variable assigned to two or more values. There are some instances when you are writing code, and you will need to take two values and place them with the same variable.

To make sure that you are able to go through this, we simply need to work with some of the same procedures that we talked about through this phase. We just need to make sure that we are putting in the equal signs in the right spots so that the compiler knows which values are going to the variable. So, you could write out something like a = b = c = 1 to show the compiler how each of these variables is going to equal 1. Or you could write out something like 1 = b = 2 to show that this variable is going to have two values assigned to it.

One of the most important things that we are able to focus on here is that the variable has to be assigned over to a value in order to work the way that we want in the code, and that these variables are basically just an empty spot in the memory that is reserved for the value that you decide to assign to the variable. Then, when it is time for the compiler to call up some of these values, it is going to be able to call up the variable, and all of it is going to show up the way that you want.

There are many times when you will need to work with these variables in your code, and they are often going to be assigned with some of the types of data that we were talking about earlier in this phase. When we are able to make all of these parts fit in with one another, we will find that it is so much easier for us to get the results that we would like in the process. Make sure to learn about these variables, and how to properly set it up with the right value in order to get the code to work the way that you would like.

CHAPTER 4:

Strings, Lists, Dictionaries, and Tuples

Programmers often use strings, lists, dictionaries, and tuples when they write a script in Python. These data types are known as data structures. Strings are pieces of text that are grouped, while tuples and lists are groups of individual data items that have been grouped. A dictionary is a group of pairs that have the highest considerations. The different methods that are used to access the data in these structures are the same.

You can also look at these data types differently, depending on whether the values that the variable holds can be modified. This is called the mutability of the data type. A string and tuple cannot be modified, but they can be used to create new tuples and strings. A list is mutable, which means that you can either remove or add items to it.

Items in Sequences

You can fetch individual items in a sequence using an index. This index will indicate the position of the element. The index is often an integer that is written in square brackets immediately after the name of the variable. So, you can obtain the variable in a list by specifying the name of the list, followed by the index. You can also access a single character in a string.

```
>>> vegetable = ‘pumpkin’
>>> vegetable [0]
‘p’
```

Or an item in a list:

```
>>>vegetable = [‘pumpkins’, ‘potatoes’, ‘onions’, ‘eggplant’]
>>>vegetable [1]
‘Pumpkins’
```

You will notice that indexing in Python is zero-based. This means that you can only start counting the variables at zero. An index with the number 3 in the square brackets will look at the fourth item in the list since the first item will be indexed as zero. So, you can use any number of integers beginning from zero to index the variables in your data set. A negative index will count the list from the end to the beginning:

```
>>>vegetable [-1]
'eggplant'
```

Slices can be used to grab the different sections in any sequence. This method is used to fetch many items in a sequence. A slice is written using the same notation as an index, the only difference is that a colon separates the integers. The first value is the starting point, and this value is included. The second number in the notation is the endpoint of the slice, and it is exclusive. If you look at s[0:2], the compiler will slice the list from the variable with the index zero and stop exactly before the variable with the index two. You do not necessarily have to use the third value, and this is an additional step. This can be negative; therefore, you can retrieve all the other items instead of picking this item from the sequential list. Alternatively, you can retrieve items backward as well. So, s [i: j: step] will give you the slice that begins from the variable i, but will not include the variable j. Here, s is the sequence.

If you ignore the initial point, the slice will always start at the beginning of the sequence. If you forget the end, the slice will continue to the end of the original or main sequence. Slicing and indexing do not change the original sequence, they will develop a new sequence. The actual data items in the sequence will be the same. So, if you want to modify an individual item in the sequence, you will see that the item has changed in the slice as well.

Tuples

Tuples are a group of items or elements that are ordered and immutable. You should think of a tuple as a sealed packet of information.

A tuple is specified as a comma-separated list of values. These values can be enclosed within parentheses if necessary. In some cases, these parentheses are required, so always use them regardless of whether or not you think they are necessary. The values in the tuple do not necessarily have to be of the same data type. Some values can also be other tuples.

Creating a Tuple

Tuples can be created with no items in it using the round brackets ().

```
>>>empty_tuple= ()
```

If you do not want more than one item in the tuple, you should enter the first item, followed by a comma.

```
>>>one_item = (‘blue’,)
```

Changing Values in a Tuple

The values in a tuple cannot be changed. These tuples are sealed packets of information that are often used in situations where a set of values need to be passed on from one location to another. If you wish to change the sequence of the data, you should use a list.

List

A list is a comma-separated and ordered list of items that are enclosed within square brackets. The items within the list do not have to be of the same data type. You can also include a list within a list.

A list can be concatenated, indexed, and sliced just like any other sequence you can use in Python. You can change some items within a list when compared to a tuple or string. Lists are very flexible when compared to tuples. You can either clear a list or change the list completely by slicing the list and assigning the data to other variables.

Creating a List

It is easy to create a list.

```
>>> shopping_list = [‘detergent’, ‘deodorant’, ‘shampoo’, ‘body wash’]
```

Modifying a List

A new value can be added to a list using the assignment operator.

```
>>> shopping_list [1] = ‘candles’
>>> shopping_list
```

```
[‘detergent,’ ‘candles,’ ‘deodorant,’ ‘shampoo,’ ‘body wash’]
```

Stacks and Queues

You can use lists to store and retrieve data or variables in a specific order since lists are ordered data types. The main models that one can use to do this are by using stacks and queues. A stack uses the last in first out (LIFO) approach. A real-world example of this approach is how the discard pile is used in a card game. You add cards to the top of the pile and remove the card from the top. You can include items into a stack using the list.append() function and remove the items from a stack using the pop() function. There are no additional index arguments that you will need to include when you use these functions, so the last item in the list is removed.

```
>>> shopping_list.append (‘brush’)
>>> shopping_list.pop()
‘candles’
>>> shopping_list
[‘detergent,’ ‘deodorant,’ ‘shampoo,’ ‘body wash’]
```

The second approach is to create the first in first out (FIFO) structure. A queue uses this type of approach. This method works like a pipe where the first item is pushed out of the pipe before the remaining items. You can use the same functions, append() and pop(), to either push items into the queue or remove them from the queue. You will, however, need to use the index zero to indicate that the items should be popped from the start of the list.

```
>>> shopping_list.append (‘brush’)
>>> shopping_list.pop(0)
‘detergent’
>>> shopping_list
[‘deodorant,’ ‘shampoo,’ ‘body wash,’ ‘brush]’
```

Dictionaries

A dictionary is much like an address book. If you know the name of the person you wish to contact, you can obtain the details of that person. The name of the person is the key, while the details of the person are the value.

The key that you use in a dictionary should be an immutable data type; that is, it can be a number, tuple, or string. The value can be anything. A dictionary is a mutable data type, and it is for this reason that you can add, modify, or remove any pairs from the dictionary. The keys are mapped to an object, and it is for this reason that a dictionary is also known as mappings. This will show you that a dictionary behaves different to a sequence.

A dictionary can be used anywhere you want to store a value or attribute that will describe an entity or a concept. For instance, you can use a dictionary to count the number of instances of a specific state or object. Since every key has a unique identifier, you cannot have duplicate values for the same key. Therefore, the key can be used to store the items in the input data, and the values can store the result of the calculation.

CHAPTER 5:

Python Modules

Modules, also known as packages, are a set of names. This is usually a library of functions/object classes that are made available to be used within different programs. We used the notion of modules earlier in this phase to use some function from the math library. In this phase, we are going to cover in-depth on how to develop and define modules. To use modules in a Python program, the following statements are used: import, from, reload. The first one imports the whole module. The second allows import only a specific name or element from the module. The third one, reload, allows reloading a code of a module while Python is running and without stopping in it. Before digging into their definition and development, let's start first by the utility of modules or packages within Python.

Modules Concept and Utility Within Python

Modules are a very simple way to make a system component organized. Modules allow reusing the same code over and over. So far, we were working in a Python interactive session. Every code we have written and tested is lost once we exit the interactive session. Modules are saved in files that make them persistent, reusable, and sharable. You can consider modules as a set of files where you can define functions, names, data objects, attributes, and so on. Modules are a tool to group several components of a system in a single place. In Python programming, modules are among the highest-level unit. They point to the name of packages and tools. Besides, they allow the sharing of the implemented data. You only need one copy of the module to be able to use across a large program. If an object is to be used in different functions and programs, coding it as a module allows to share it with other programmers.

To have a sense of the architecture of Python coding, we go through some general structure explanation. We have been using so far in this book very simple code examples that do not have high-level structure. In large applications, a program is a set of several Python files. By Python files, we mean files that contain Python code and have a .py extension. There is one main high-level program, and the other files are the modules. The high-level file consists of the main code that dictates the control flow and executes the application. Module files define the tools that are needed to process elements and components of the main program and maybe elsewhere. The main program makes use of the tools that are specified in the modules.

In their turn, modules make use of tools that are specified in other modules. When you import a module in Python, you have access to every tool that is declared or defined in that specific module. Attributes are the variables or the functions associated with the tools within a module. Hence, when a module is imported, we have access to the attributes of the tools as well to process them. For instance, let's consider we have two Python files named file1.py and file2.py, where the file1.py is the main program, and file2.py is the module. In the file2.py, we have a code that defines the following function:

```
def Xfactorial (X):
P = 1
for i in range (1, X + 1):
P *= i
return P
```

To use this function in the main program, we should define code statements in the file1.py as follows:

```
Import file2
A = file2.Xfactorial (3)
```

The first line imports the module file2.py. This statement means to load the file file2.py. This gives access to the file1.py to all tools and functions defined in file2.py by the name file2. The function Xfactorial is called by the second line. The module file2.py is where this function is defined using the attributes' syntax. The line file2.Xfactorial() means fetch any name value of Xfactorial and lies within the code body of file2. In this example, it is a function that is callable. So, we have provided an input argument and assigned the output result to the variable A. If we add a third statement to print the variable A and run the file file1.py, it would display 6 which is the factorial of 3. Along with Python, you will see the attribute syntax as object.attribute. This allows calling the attributes that might be a function or data object that provides properties of the object.

Note that some modules that you might import when programming with Python are available in Python itself. As we have mentioned at the beginning of this book, Python comes with a large stand library that has built-in modules. These modules support all common tasks that might be needed in programming from operating system interfaces to graphical user interface. They are not part of the language. However, they can be imported and comes with a software installation package. You can check the complete list of available modules in a manual that comes with the installation or goes to the official Python website: www.Python.org. This manual is kept updated every time a new version of Python is released.

How to Import a Module

We have talked about importing a module without really explaining what happens behind in Python. Imports are a very fundamental concept in Python programming structure. In this phase, we are going to cover in-depth how really Python imports modules within a program. Python follows three steps to import a file or a module within the work environment of a program. The first step consists of finding the file that contains the module. The second step consists of compiling the module to a byte-code if required. Finally, the third step runs the code within the module file to build the objects that are defined. These three steps are run only when the module is imported for the first time during the execution of a program. This module and all its objects are loaded in the memory. When the module is imported further in the program, it skips all three steps and just fetch the objects defined by the module and are saved in memory.

At the very first step of importing a module, Python has to find the module file location. Note that, so far in the examples we presented, we used import without providing the complete path of the module or extension .py. We just used import math, or import file2.py. Python import statement omits the extension and the path, we just simply import a module by its name. The reason for this is that Python has a module that looks for paths called ‘search path module.' This module is used specifically to find the path of module files that are imported by the import statements.

In some cases, you might need to configure the path search of modules to be able to use new modules that are not part of the standard library. You need to customize it to include these new modules. The search path is simply the concatenation of the home directory, directories of PYTHONPATH, directories of the standard library, and optionally if the content of files with extension .pth when they exist. The home directory is set automatically by the system to a directory of Python executable when launched from the interactive session, or it can be modified to the working directory where your program is saved. This directory is the first to be searched when import a module is run without a path. Hence, if your home directory points to a directory that includes your program along with the modules, importing these modules does not require any path specification.

The directory of the standard library is also searched automatically. This directory contains all default libraries that come with Python. The directories of PYTHONPATH can be set to point toward the directory of new modules that are developed. In fact, PTYHONPATH is an environment variable that contains a list of directories that contains Python files. When PTYHONPATH is set, all these paths are included in the Python environment, and the search path directory would search these directories too when importing modules. Python also allows defining a file with .pth extension that contains directories, one in each line. This file serves the same as PTYHONPATH when included appropriately in a directory. You can check the directories' paths included when you run Python using sys.path. You simply print sys.path to get the list of the directories that Python will be searching for.

Remember, when importing a module, we just use the name of the module without its extension. When Python is searching for a module in its environment paths, it selects the first name that matches the module name regardless of the extension. Because Python allows using packages that are coded in other languages, it does not simply select a module with .py extension but a file name or even a zip file name that matches the module name being imported. Therefore, you should name your modules distinctly and configure the search path in a manner that makes it obvious to choose a module.

When Python finds the source code of the module file with a name that corresponds to the name in the import statement, it will compile it into byte code in case it is required. This step is skipped if Python finds an already byte code file with no source code. If the source code has been modified, another byte code file is automatically regenerated by Python while the program runs in other further executions. Byte code files have typically .pyc extension. When Python is searching and finds the module file name, it will load the byte code file that corresponds to the latest version of the source code with .py extension. If the source code is newer than the byte code file, it will generate a new one by compiling the source code file. Note that only imported files have corresponding files with .pyc extension. These files, the byte code files, are stored on your machine to make the imports faster in future use.

The third step of the import statement is running the module's byte code. Each statement and each assignment in the file are executed. This allows generating any function, data objects, and so on defined in the module. The functions and all attributes are accessed within the program via importers. During this step, you will see print statements if they exist. The 'def ' statement will create a function object to be used in the main program.

To summarize the import statement, it involves searching for the file, compiling it, and running the byte code file. All other import statements use the module stored in memory and ignore all the three steps. When first imported, Python will look in the search path module to select the module. Hence, it is important to configure correctly the path environment variable to point to the directory that contains new defined modules. Now that you have the big picture and the concept of modules, let's explore how we can define and develop new modules.

CHAPTER 6:

Classes and Objects

The Python Classes

The next thing that we need to focus on in the Python language is the classes. There is a lot that we are able to focus on when it comes to the Python classes, and knowing how to make this work is going to be so important to the work that you do in Python. Classes, to keep things as simple as possible, are going to be a type of container that can hold onto your objects, as well as some other parts of your code. You have to make sure that these classes are always named in the proper manner, and that you store them in the right spots to ensure they are able to work the way that you want. Then, we have to double-check that we are storing the right objects into these classes as well.

The neat thing here is that it is possible to take any object and put it into one of the classes that you choose to design. But there are a few rules. First, we have to make sure that the objects that are similar to one another end up in the same class.

Of course, you are not required to put the identical items in each class, but when you take a look at any class in your code, there has to be some sense as to why those items are in there together.

For example, you can create a class that will hold onto all types of vehicles, and it is not required that you fill it with just blue cars. You can mix and match the items or the objects in your classes, just make sure that the items and objects that are in a particular class make sense to be in there with one another, rather than just being random.

There are a lot of things that you are able to do with some of the classes that you are going to work on, but we are going to keep our focus on some of the steps that you can take in order to create one of these new classes. Because of the way that the Python language is set up, you will need to create a lot of classes in order to keep the information as organized as possible.

In Python, it is very important that we are able to spend our time working on creating some of our classes because this is one of the best ways for the code to be organized and to make sure that nothing is going to get lost in the process.

To make one of these classes on your own, though, you have to use the right keywords to name that class. You have some freedom here in naming the class, just make sure that the name you want to work with comes right after the keyword, and that it is a name that you will be able to remember at a later time.

After you have had a chance to name your class, it is time to work on naming the subclass that you are working with. This subclass is going to end up in the parenthesis if you would like to maintain some of the programming rules that we have in place. Make sure that at the end of the first line, when you are creating a new class, that there is a semicolon in place. While your code will technically work if you forget to add in this part, it is considered part of the coding rules to have this and will look a lot better if you make sure that it is there.

Writing a class right now probably sounds pretty confusing, and you may be worried that you won't be able to figure out how to make it happen, or how to get it to work for you. This is actually a fairly simple process and will not take up as much time and effort as you would like. To help us make this as easy as possible, let's take a break here and discuss the coding that is needed to write a class and how it is meant to look. If you would like to create one of your own classes, then the following code is the right one for you:

```
class Vehicle(object):
#constructor
def_init_(self, steering, wheels, clutch, breaks, gears):
self._steering = steering
self._wheels = wheels
self._clutch = clutch
self._breaks =breaks
self._gears  = gears
```

```
#destructor
def_del_(self):
      print(“This is destructor….”)
#member functions or methods
def Display_Vehicle(self):
    print(‘Steering:’ , self._steering)
    print(‘Wheels:’, self._wheels)
    print(‘Clutch:’, self._clutch)
    print(‘Breaks:’, self._breaks)
    print(‘Gears:’, self._gears)
#instantiate a vehicle option
myGenericVehicle = Vehicle(‘Power Steering’, 4, ‘Super Clutch’, ‘Disk Breaks’, 5)
myGenericVehicle.Display_Vehicle()
```

If you would like, you can try out this code. Just open up your text editor and type the code inside. Once you have a chance to write out and then execute this code, let’s divide it up and see what happened above.

The Python Objects

Now that we have taken a look at the Python classes, we need to take a look at some of the Python objects as well. The classes are going to be like the containers in this language, and then the objects are going to be the parts that we put into the containers to keep them nice and safe. This works because Python is considered an OOP language, or Object-Oriented Programming language.

That is why we are going to take a bit of time here to look at what this means for your coding, and why it is such an important part to include in some of our coding as well.

One of the features that programmers, especially those who are new to programming, will enjoy when it comes to using an OOP language is that the procedures of any object you use will have the power needed to access data fields, and there are even some situations where the objects are even able to modify these data fields. With this kind of language, we are able to create the kind of program that does this simply by building it from a series of objects that are capable of not only interact, but also talk with one another.

This may seem like a simplistic way to look at the whole process, but it helps us to get a better understanding of how this process works and how we are able to use it. You will find though that even with some of this simplicity, the idea that Python is an OOP language adds in some diversity to the process. Each language is going to come in with some differences, but the ones that you may use, including Python, are going to be known as class-based.

This means that the code is going to have each of the objects fall into a class of some sort, which is one of the best ways to make sure that all of this is as organized as possible.

As you start with some of the coding that is needed with Python, you will find that the fact that Python is an OOP language is going to be to your benefit. It can help make the writing of code easier than ever, and will ensure that it is going to make coding easy even if you are a beginner. This is especially true when we consider how hard it is to work with some of the older languages that are not going to be considered OOP.

Before we go further into OOP languages and what they mean, we need to do a quick summary of classes and how they work. Classes are like small containers. You can pick any name that you want for the classes and then add in any item that you would like. Of course, to keep things organized and to help you call up these classes later on in your code, it may be a good idea to pick out a name that describes what is held inside.

When you work on your objects in the code, you will find that they actually match up with real things outside of the programming world. For example, one of your objects might be a car, one might be a book, and one might be a ball. You also have the option of picking out an object that is a bit more abstract but sticks with things that work in the code that you are creating. These objects are going to stay inside any of the classes that you create and then place the object of sin. You want to make sure that any object that is inside the same class has some kind of similarities to each other. These objects don’t have to be exactly alike. But if another programmer comes to look at the code, they should be able to figure out why one group of objects is placed in the same class.

An example of this would be a class with dogs inside. You don’t have to just put St. Bernard’s in the class unless you want to. You could include big dogs, small dogs, medium dogs, fluffy dogs, or any other kind of dog that you would like to put into this class. Other programmers will see that the objects aren’t identical, but they will understand that your class is for dogs, rather than a specific type of dog.

These objects, as well as the classes that come with them, will be able to work with one another because they will make sure that the code you do in Python as neat and organized as they can. You are able to spend some time putting the objects that you have into the right class, which is going to keep things organized and allows your code to work in a more efficient manner. Even as a beginner, you will notice how much easier than the whole coding process can be when you want to keep things working efficiently with the Python language.

CHAPTER 7:

IDLE

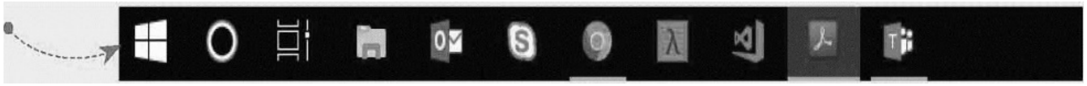
When you download and install Python, it will also install an application called IDLE. Integrated Development and Learning Environment or also known as IDLE, it is an integrated development environment, or IDE, that helps us with writing Python programs.

Think of it as an electronic notepad with some additional tools to help us write, debug, and run our Python code.

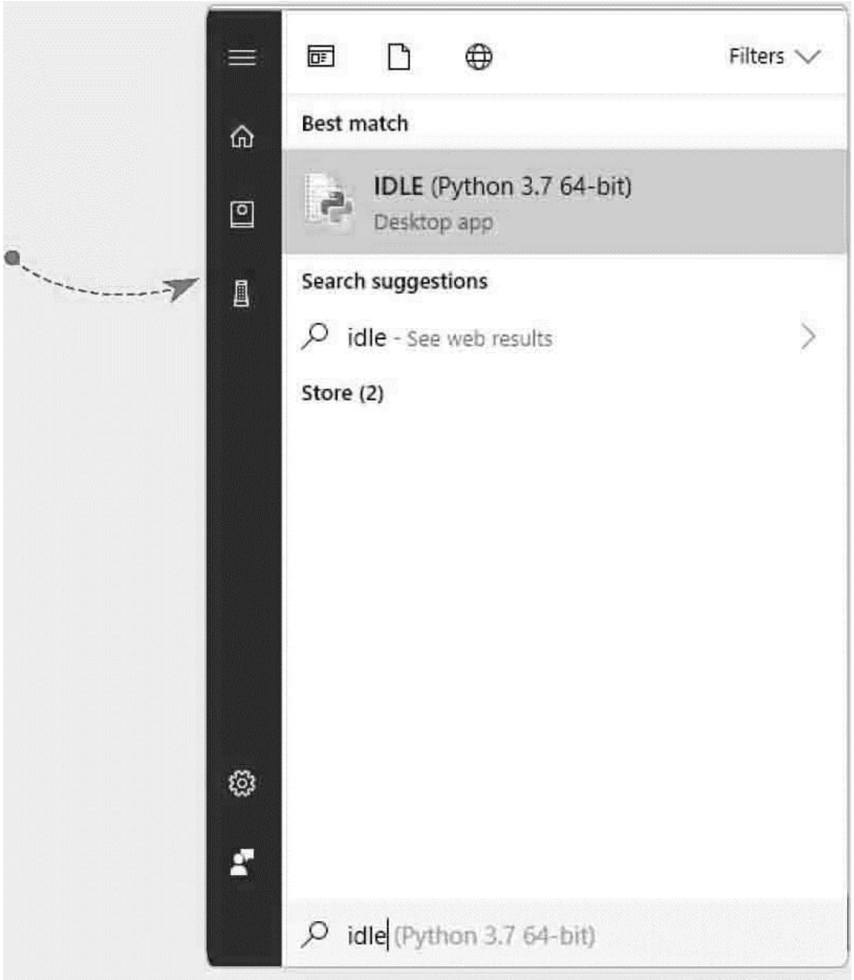
To work in Python, you will need to open IDLE because opening Python files directly won’t work!

On a PC

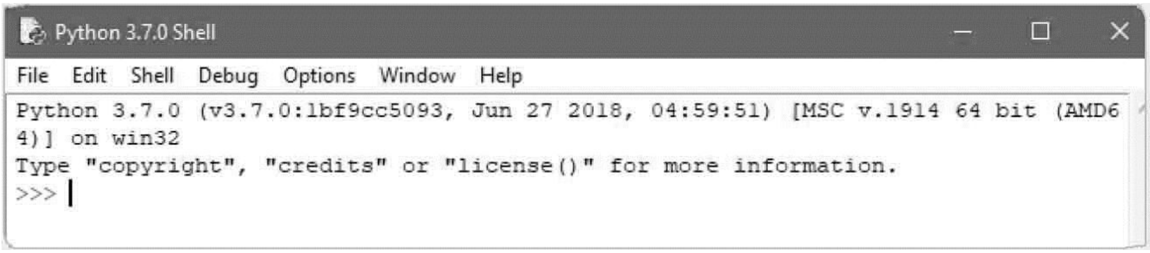
Step 1. Click the Windows Start menu.



Step 2. Start typing “idle”, then select the search result IDLE (Python 3.7 64-bit). Note: Yours might say IDLE (Python 3.7 32-bit) if that’s the kind of machine you have.



Step 3. A window should pop up that looks like this:

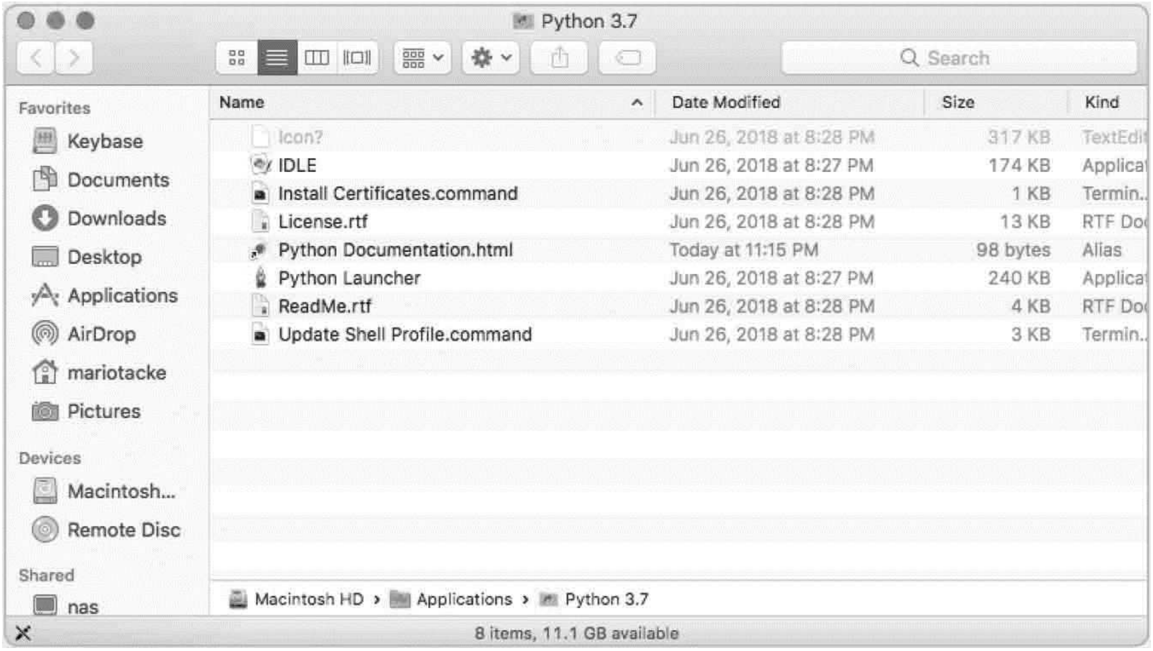


```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Step 4. Nice! You opened IDLE on Windows and are now ready to start writing some codes in Python!

On a Mac

Step 1. Navigate to GO > APPLICATIONS.

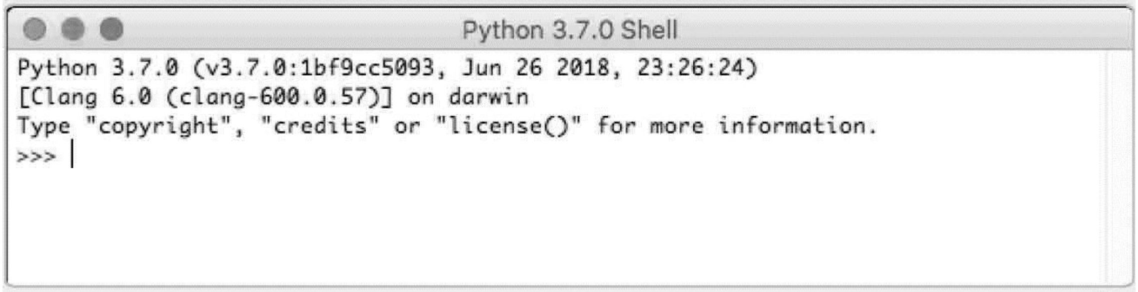


Name	Date Modified	Size	Kind
Icon?	Jun 26, 2018 at 8:28 PM	317 KB	TextEdi
IDLE	Jun 26, 2018 at 8:27 PM	174 KB	Applica
Install Certificates.command	Jun 26, 2018 at 8:28 PM	1 KB	Termin...
License.rtf	Jun 26, 2018 at 8:28 PM	13 KB	RTF Do
Python Documentation.html	Today at 11:15 PM	98 bytes	Alias
Python Launcher	Jun 26, 2018 at 8:27 PM	240 KB	Applica
ReadMe.rtf	Jun 26, 2018 at 8:28 PM	4 KB	RTF Do
Update Shell Profile.command	Jun 26, 2018 at 8:28 PM	3 KB	Termin...

Step 2. Find the Python 3.7 folder and open it.

Step 3. Double-click on the IDLE icon.

Step 4. A window should pop up that looks like this:



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 26 2018, 23:26:24)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Step 5. Congratulations! You opened IDLE on a Mac and are now ready to start writing some code in Python!

Python IDLE has the following features:

- Syntax highlighting
- Auto-completion of code statements
- Smart indentation
- Integrated debugger with the stepping, persistent breakpoints, and call stack visibility features.

Launching Python

To get started, you have to understand how to launch the Python app. You can launch Python from the Terminal or use the desktop environment for starting the IDLE app. Simply launch the Terminal and type: “idle” at the command prompt. Now that you’ve launched the Python, it’s now time to begin coding.

Let’s now create our first program in Python. Follow these steps to write your first Python program:

- Open the Python IDLE.
- Write the Python language statements (instructions) in the IDLE window.
- Run the program

That’s it! Simple. Isn’t it?

Now, here’s a quick way to see the programming process in action...Proceed and copy/paste the following code into your Python IDLE window.

```
print (“Hello World! This is my first Machine Learning program”)
```

Run the program. What do you see as the output?

Well, the phrase “Hello World! This is my first Machine Learning program” appears.

Congratulations! You’ve just written your first Python code. I know you’re now excited to begin coding ML systems. Don’t worry so much about the meaning of statements. If you are a Machine Learning novice, mastering some Python programming concepts will help you understand how to design ML applications.

Next up, let’s dive in together and get to the basics of Python programming.

An Overview of Python

Now that you have executed your first Python program, what else do you need to know? Well, it’s now time to understand the vital components of any Python code, including its structure. All Python programs have the following structure:

```
import sys
def main ():
main ()
{
```

```
Program statements
}
```

As you can see from this program structure, all Python codes should always start with the keyword “import.” Now, what are we importing? Python language is object-oriented. Therefore, it has components of all the Object-Oriented Programming languages. One such property is inheritance or in simple terms, code reuse. The ability to inherit features of codes in Python allows programmers to reuse pieces of codes that had been written elsewhere.

Technically speaking, the import statement tells the Python interpreter to declare classes that have already been used in other Python packages without referring to their full package names. For instance, the statement: “import sys” informs the interpreter to include all the system libraries such as print whenever the Python program is starting.

What does the statement “def main ( ):” mean?

Whenever a Python program is loaded and executed, the computer’s memory—the Random Access Memory—contains the objects with function definitions. The function definitions provide the programmers with the capabilities of instructing the control unit to place the function object into the appropriate section of the computer’s memory. In other words, it’s like instructing the control unit to check the main memory and initialize the program that needs to be executed.

The function objects in the memory can be specified using names. That’s where the statement “def main ( ):” comes in. It simply tells the control unit to start executing the Python code statements that are placed immediately after the statement “def main ( ):”

For example, the Python code below creates a function object and assigns it the name “main”:

```
def main ():
    if len (sys.argv) == 10:
      name = sys.argv [2]
    else:
     name = “Introduction to Machine Learning.”
    print (“Hello”), name
```

In the above code, the Python interpreter will run all the function statements in the Python file by placing the set of functions objects in the memory and linking each of them with the namespace. This will happen when the program is initialized with the import statement.

But more fundamentally, “What are the different elements of Python code?” Well, all Python programs have the following components:

- **Documenting the program.** Any statement in the program (except the first) that starts with “#” is treated as a command line or comment line and will be ignored during execution. This will allow you to comment on sections of the code for proper documentation.
- **Keywords.** The keywords are instructions that the interpreter recognizes and understands. For instance, the word “print” in the earlier program is a keyword. In Python, there are two main types of keywords: the functions and the control keywords. Functions are simple verbs such as print that tell the interpreter what to do while the control keywords control the flow of execution.
- **Modules.** Python program is shipped with a large list of modules that increase its functionality. The modules will help you to organize your code in a manner that’s easy to debug and control the code.
- **Program statements.** The program statements are sentences or instructions that tell the control unit to perform a given operation. Unlike most programming languages, the Python statements don’t need a semicolon at the end.
- **Whitespace.** The white spaces are a collective name given to tabs, the spaces, and newlines/carriage returns. The Python language is strict on where the white space should be placed in the code.
- **Escape sequences.** The Escape sequences are special characters that are used for output. For instance: the sequence “\n” in the program tells Python to output on a new line.

Python Variables

There's so much that goes on in the main memory of the computer whenever you run a program. Understanding the concept of variables and data types will help you to write efficient programs.

A program is simply a sequence of instructions (statements) that directs your computer to perform a particular task. For instance, the previous program printed the phrase "Hello World! This is my first program" on the screen when it was executed. But before you could see the output on the screen, some data had to be kept in the computer's memory.

The use of data applies to all programming languages—Python included—therefore, understanding the mechanisms of data management in the computer's memory is the first step towards developing robust, efficient, and powerful applications.

A variable can be conceived as a temporary storage location in the computer's main memory and specifically in the Random Access Memory. This temporary storage location is what will hold the data that you would like to use in the program. In other words, the variable location of memory that holds data whenever your program is executing. So, whenever you define a variable, you'll actually be reserving a temporary storage location in the computer's memory.

All the variables that you define must have names and an equivalent data type —a sort of classification of the variable that specifies the type of value the variable should hold. The data types help to specify what sort of mathematical, relational, or even logical operations that you can apply to the variable without causing an error. Ideally, when you assign variables to data types, you can begin to store numbers, characters, and even constants in the computer's main memory.

Since Python language is an oriented programming language, it is not "statically typed." This means that the interpreter regards every variable as an object. Therefore, you have to declare the variables before using them in your program. So, how can you declare variables in Python?

Names or identifiers usually declare Python variables. Just like any other programming languages that you have so far learned, the conventions for naming the variables must strictly be adhered to. Below are some naming conventions that you should follow when declaring variables:

- All variable names should always begin with a letter (A to Z) or an underscore. For instance, “age” is a valid variable name while “–age” isn’t a valid variable name.
- Any variable name you declare cannot start with a number. For instance, 9age is not a valid variable name.
- Special symbols shouldn’t be used when declaring variable names. For instance, @name isn’t allowed as a variable name.
- The maximum number of characters to use for your variable name shouldn’t exceed 255.

To reserve a temporary memory location in the name of a variable, you don’t have to use the explicit declaration like other programming languages. If you’ve had experience in other programming languages such as Pascal or C, I am sure you know that declaring a variable explicitly before assigning any value is a must.

In Python, the declaration of variables usually occurs automatically the moment you assign a value to it. For instance, the statement:

age=10

Automatically reserves a temporary storage location in memory space called “age” and assigns 10 to it.

It is also possible to assign a single value to several variables simultaneously. For instance, the statement below reserves temporary memory spaces for 2 variables namely: age and count, and assigns the value 30:

age, count=30

Python language has different categories of data types that are used to define the storage methods and mathematical operations. Below are examples of data types in Python language:

- Numbers
- String
- List
- Tuple
- Dictionary

Numbers

The Number data types stores numeric values. The number of objects will automatically be initialized whenever you assign a specific value to the variable. For instance, the code illustrated below creates 2 variable objects (age and count) and assigns them the values 10 and 30, respectively:

```
age = 10
count= 30
```

If you want to delete the reference to the Number object, you'll use the word "del" followed by the variable name that you wish to delete. Consider the code below that deletes two variables: age and count that have already been declared and used."

```
del age, count
```

Python language supports four different categories of number types. These are:

- int. when used in a declaration, it refers to signed integers. These include those whole numbers that range from 8 bits to 32 bits.
- Long. These are long integers. They can be represented either in octal and hexadecimal numbering notation.
- float. These are floating real point values. They may range from 8 bits to 64 bits long.
- Complex. These are complex numbers.

Strings

Strings are stored as consecutive sets of characters in the computer's memory locations. Python language allows you to use either pair of single or double quotes when defining the strings. Other subsets of string variable types can be specified using the slice operator ([ ] and the [:]) with the indexes that range from 0 at the beginning of the string. The plus (+) operator performs string concatenation (joining of two or more strings) while the asterisk (*) operator performs string repetition.

CHAPTER 8:

Numbers and Operators

Since it is called a “computer,” mathematical calculation, of course, is the basic computer skill. The operations in Python are simple and intuitive. Open up the Python Command Line, type in the following numeric operation, and you’re ready to run it:

Numbers

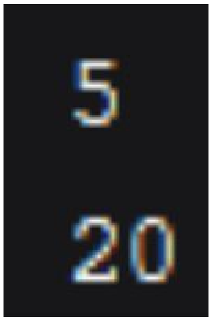
Numbers are used for storing values, which are numeric. We create them by assigning some value to them. See the example given below:

```
number1 = 5
number2 = 20
```

When you call the variable “number1,” you will get 5, and if you call variable “number2,” you will get 20 as the result. This is shown below:

```
number1 = 5
number2 = 20
print (number1)
print (number2)
```

The program will give the following result:



```
5
20
```

Mathematical Operators

1) Addition

```
>>>4 + 2
```

2) Subtraction

```
>>>4 - 2
```

3) Multiplication

```
>>>4 * 2
```

4) Division

```
>>>4 / 2
```

5) Remainder

```
>>>4 % 2
```

With these basic operations, we can use Python as if we were using a calculator. Take buying a house. A property costs 20000 dollars and is subject to a 5% tax on the purchase, plus a 10% down payment to the bank. Then, we can use the following code to calculate the amount of cash to be prepared:

```
>>>20000*(0.5+ 0.1)
```

In addition to the usual numeric operations, strings can also be added. The effect is to concatenate two strings into one character.

String

Input:

```
>>>“ I am a follower of “ + “Christianity”
```

Output:

```
I am a follower of Christianity
```

Input:

```
>>>“Example” *2
```

Output:

```
Example
```

Multiplying a string by an integer n repeats the string n times.

Comparison Operator

Python uses comparison operators like ==, >, and < in its program. Below, we will explain with an example.

Program code is below:

```
first = 34
second = 44
if ( first > second)
print  “First one is larger”
else
print “Second one is larger”
```

Output is:

```
Second one is larger
```

Logical Operators

In addition to numerical operations, computers can also perform logical operations. It’s easy to understand the logic if you’ve played a killing game or enjoyed a detective story. Like Sherlock Holmes, we use logic to determine whether a statement is true or false. A hypothetical statement is called a proposition, such as “player A is a killer.” The task of logic is to find out whether a proposition is true or false.

Computers use the binary system, where they record data in zeros and ones. There are technical reasons why computers use binary. Many of the components that make up a computer can only represent two states, such as the on and off of a circuit, or the high and low voltages. The resulting system is also relatively stable. If you use the decimal system, some computer components will have 10 states, such as the voltage into 10 files. That way, the system becomes complex and error-prone. In Binary Systems, 1 and 0 can be used to represent the true and false states. In Python, we use the keywords True and False to indicate True and False. Data such as True and False are called Boolean values.

Sometimes, we need further logical operations to determine whether a complex proposition is true or false. For example, in the first round, I learned that “player A is not a killer” is true, and in the second round, I learned that “player B is not a killer” is true. So, in the third round, if someone says, “player A is not a killer, and player B is not a killer,” then that person is telling the truth. If the two propositions connected by “and” are respectively true, then the whole proposition is true. Virtually, we have a “and” of the logical operation.

In the operation, when both sub propositions must be true, the compound proposition connected by and is true. The “and” operation is like two bridges in a row. You must have both bridges open to cross the river. Take, for example, the proposition that China is in Asia and Britain is in Asia. The proposition that Britain is in Asia is false, so the whole proposition is false. In Python, we use and for the logical operation of and.

```
>>>True and True  # True
>>>False and True  # false
>>>False and False  # True
```

We can also compound the two propositions with “or.” Or is humbler than an aggressive “and.” In the phrase “China is in Asia, or Britain is in Asia,” for example, the speaker leaves himself room. Since the first half of this sentence is true, the whole proposition is true. “Or” corresponds to the “or” logic operation.

In the “or” operation, as long as there is a proposition for true, then “or” connected to the compound proposition is true. The “or” operation is like two bridges crossing the river in parallel. If either bridge is clear, pedestrians can cross the river.

The above logic operation seems to be just some life experience and does not need a computer such as complex tools. With the addition of a judgment expression, a logical operation can really show its power.

Operator Precedence

If there is more than one operator in an expression, consider the precedence of the operation. Different operators have different precedence. Operators can be grouped in order of precedence. Below is the list of operator precedence in an order.

Exponent powers have the highest precedence, followed by the mathematical operator multiplication, division, addition, and subtraction. And the next comes Bitwise operators followed by logical operators at the end.

CHAPTER 9:

Operators in Python

You will realize that some program requests specific information or show the text on the screen. Sometimes we start the program code by informing the readers about our programs. To make things look easy for the other coders, it is important to give it the name or title that is simple and descriptive.

As a programmer, you can use a string literal that comprises the print function to get the right data. String literal is a line of the text surrounded by the quotes. They can be either double or single quotes. Although the type of quotes a programmer use matters less, the programmer must end with the quotes that he/she has used at the beginning of the phrase. You can command your computer to display a phrase or a word on the screen by just doing as mentioned above.

Files

Apart from using the print function to obtain a string when printing on the screen, it can be used to write something onto the file. First, you will have to open up the myfile.txt and write on it before assigning it the file, which is a variable. Once you have completed the first step, you will have to assign "w" in the new line to tell the program that you will only write or make changes after the file has opened. It is not mandatory to use print function; just use the right methods like read method.

Read method is used to open specific files to help you read the available data. You can use this option to open a specific file. Generally, the read method helps the programmers to read the contents into variable data, making it easy for them to open the program they would like to read.

Integers

Always make sure that the integers are kept as whole numbers if you are using them. They can be negative or positive only if there are no decimals. However, if your number has a decimal point, use it as a floating number. Python will automatically display such integers on the screen.

Moreover, you cannot place one number next to others if you are using the integers because Python is a strongly typed language; thus, it will not recognize them when you use them together. However, you put both the number and the string together by making sure you turn the number into a string first before going to the next steps.

Triple Quotes

After reading and understanding both the single and double quotes, it is now time to look at the triple quotes. The triple quotes are used to define the literal that spans many lines. You can use three singles, double, or single when defining an authentic.

Strings

Although a string is seen as a complicated thing to many beginners, it is a term used by the programmers when referring to a sequence of characters and works just like a list. A string contains more functionality, which is specific than a list. You will find it challenging to format the strings when writing out the code because some messages will not be fixed easily due to its functionality. String formatting is the only way to go away within such a situation.

Escape Sequences

They are used to donate special characters which are hard to type on the keyboard or those that can be reserved to avoid confusion that may occur in programming.

Operator Precedence

It will help you to track what you are doing in Python. In fact, it makes things easy when ordering the operation to receive the right information. So, take enough time to understand how the operator precedence works to avoid confusion.

Variables

Variables refer to the labels donated somewhere in the computer memory to store something like holding values and numbers. In the programming typed statistically, the variables have predetermined values. However, Python enables you to use one variable to store many different types. For example, in the calculator, variables are like memory function to hold values that can be retrieved in case you need them later. The variables can only be erased if you store them in the newer value. You will have to name the variable and ensure it has an integer value.

Moreover, the programmer can define a variable in Python by providing the label value. For instance, a programmer can name a variable count and even make it an integer of one, and this can be written as; count=1. It allows you to assign the same name to the variable, and in fact, the Python interpreter cannot read through the information if you are trying to access values in the undefined variable. It will display a message showing syntax error. Also, Python provides you with the opportunity of defining different variables in one line even though this not a good according to our experience.

The Scope of a Variable

It is not easy to access everything in Python, and there will be differences in the length of the variables. However, the way we define the variable plays a vital role in determining the location and the duration of accessing the variables. The part of the program that allows you to access the variable is called the Scope, while the time taken for accessing the variable is a lifetime.

Global variables refer to the variables defined in the primary file body. These variables are visible throughout the file and also in the file that imports specific data. As such, these variables cause a long-term impact, which you may notice when working on your program. This is the reason why it is not good to use global variables in the Python program. We advise programmers to add stuff into the global namespace only if they plan to use them internationally. A local variable is a variable defined within another variable. You can access local variables from the region they are assigned. Also, the variables are available in the specific parts of the program.

Modifying Values

For many programming languages, it is easy for an individual to define a particular variable whose values have been set. The values which cannot be modified or changed in the programming language, are called constants. Although this kind of restrictions are not allowed in Python, they are used to ensure some variables are marked, indicating that no one should change those values. You must write the name in capital letters, separated with underscores. A good example is shown below.

```
NUMBER_OF_HOURS_IN_A_DAY=24
```

It is not mandatory to put the correct number at the end. Since Python programming does not keep tracking and has no rules for inserting the correct value at the end, you are free and allowed to say, for example, that they are 25 hours in a day. However, it is important to put the correct value for other coders to use in case they want.

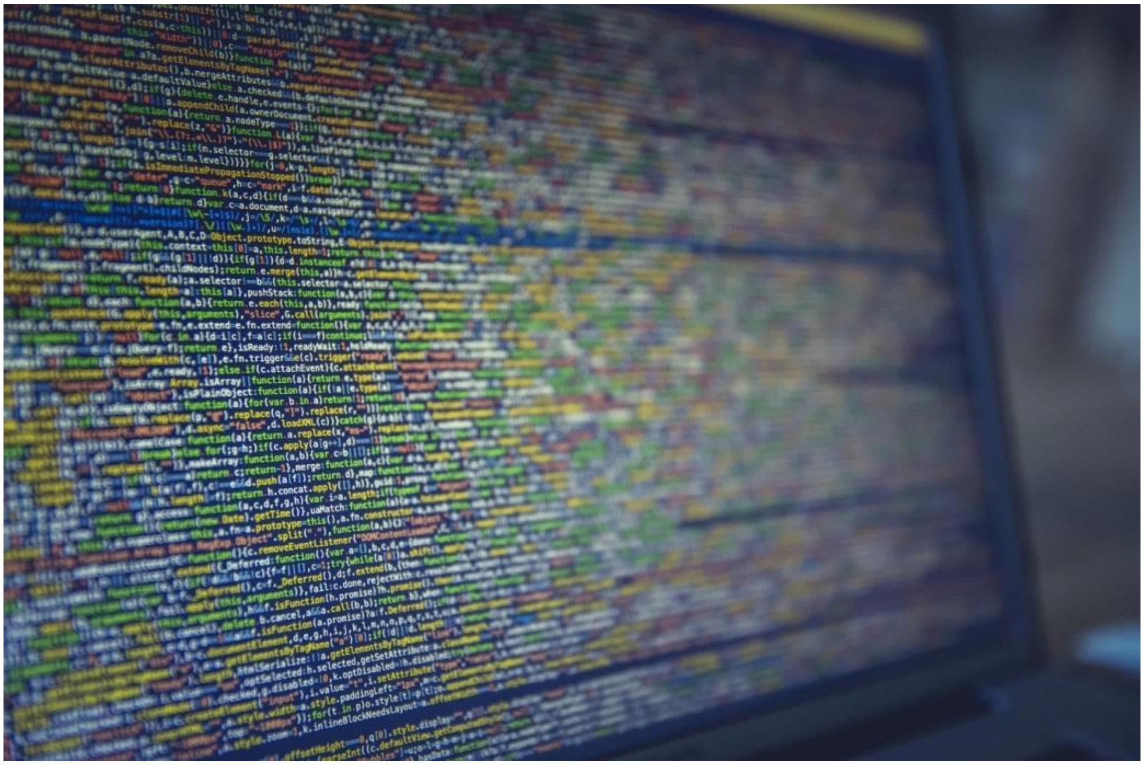
Modifying values is essential in your string as it allows a programmer to change the maximum number in the future. Therefore, understanding the working of the string in the program contributes a lot to the success of your program. One has to learn and know where to store the values, the rules governing each value, and how to make them perform well in a specific area.

The Assignment Operator

We had not given it the specific name. It refers to an equal sign (=). You will be using the assignment operator to assign values to the variable located at the left side on the right of the statement. However, you must evaluate if the value on the right side is an arithmetic expression. Note that the assignment operator is not a mathematical sign in the programming because, in programming, we are allowed to add all types of things and make them look like they are equivalent to a certain number. This sign is used to show that those items can be changed or turned into the part on the other side.

CHAPTER 10:

Essential Libraries for Machine Learning in Python



Many developers nowadays prefer the usage of Python in their data analysis. Python is not only applied in data analysis but also in statistical techniques. Scientists, especially the ones dealing with data, also prefer using Python in data integration. That’s the integration of Web apps and other environment productions.

The features of Python have helped scientists to use it in Machine Learning. Examples of these qualities include consistent syntax, being flexible, and even having a shorter time in development. It also has the ability to develop sophisticated models and has engines that could help in predictions.

As a result, Python boasts of having a series or set of very extensive libraries. Remember, libraries refer to a series of routines and sorts of functions with different languages. Therefore, a robust library can lead to tackling more complex tasks. However, this is possible without writing several code lines again. It is good to note that Machine Learning relies majorly on mathematics. That's mathematical optimization, elements of probability, and also statistical data. Therefore, Python comes in with a rich knowledge of performing complex tasks without much involvement.

The following are examples of essential libraries used in our present.

Scikit–learn

Scikit-learn is one of the best and a trendy library in Machine Learning. It has the ability to supporting learning algorithms, especially the unsupervised and supervised ones.

Examples of Scikit learn include the following.

- K-means
- Decision trees
- Linear and logistic regressions and also
- Clustering

This kind of library has major components from NumPy and SciPy. Scikit-learn has the power to add algorithms sets, that are useful in Machine Learning and also tasks related to data mining. That's, it helps in classification, clustering, and even regression analysis. There are also other tasks that this library can efficiently deliver. A good example includes ensemble methods, feature selection, and more so, data transformation. It is good to understand that the pioneers or experts can easily apply this if at all, they can be able to implement the complex and sophisticated parts of the algorithms.

TensorFlow

It is a form of algorithm which involves deep learning. They are not always necessary, but one good thing about them is their ability to give out correct results when done right. It will also enable you to run your data in a CPU or GPU. That's, you can write data in the Python program, compile it then run it on your central processing unit. Therefore, this gives you an easy time in performing your analysis. Again, there is no need for having these pieces of information written at C++ or instead of other levels such as CUDA.

TensorFlow uses nodes, especially the multi-layered ones. The nodes perform several tasks within the system, which include employing networks such as artificial neutral, training, and even set up a high volume of datasets. Several search engines such as Google depend on this type of library. One main application of this is the identification of objects. Again, it helps in different Apps that deal with the recognition of voice.

Theano

Theano also forms a significant part of Python library. Its vital tasks here are to help with anything related to numerical computation. We can also relate it to NumPy. It plays other roles such as:

- Definition of mathematical expressions
- Assists in the optimization of mathematical calculation
- Promotes the evaluation of expressions related to numerical analysis.

The main objective of Theano is to give out efficient results. It is a faster Python library as it can perform calculations of intensive data up to 100 times. Therefore, it is good to note that Theano works best with GPU as compared to the CPU of a computer. In most industries, the CEO and other personnel use Theano for Deep Learning. Also, they use it for computing complex and sophisticated tasks. All these became possible due to its processing speed. Due to the expansion of industries with a high demand for data computation techniques, many people are opting for the latest version of this library. Remember, the latest one came to limelight some years back. The new version of Theano, that’s version 1.0.0, had several improvements, interface changes, and composed of new features.

Pandas

Pandas is a library that is very popular and helps in the provisions of data structures that are of high level and quality. The data provided here is simple and easy to use. Again, it’s intuitive. It is composed of various sophisticated inbuilt methods, which make it capable of performing tasks such as grouping and timing analysis. Another function is that it helps in a combination of data and also offering filtering options. Pandas can collect data from other sources such as Excel, CSV, and even SQL databases. It also can manipulate the collected data to undertake its operational roles within the industries. Pandas consist of two structures that enable it to perform its functions correctly. That's Series which has only one dimensional, and data frames which boast of two dimensional. Pandas has been regarded as the most strong and powerful Python library over the time being. Its main function is to help in data manipulation. Also, it has the power to export or import a wide range of data. It is applicable in various sectors, such as in the field of data science.

Pandas is effective in the following areas:

- Splitting of data
- Merging of two or more types of data
- Aggregating of data
- Selecting or subsetting of data and
- Data reshaping

Diagrammatic explanations

Series Dimensional

SERIES	
A	7
B	8
C	9
D	3
E	6
F	9

Data Frames dimensional

DATA FRAME

	A	B	C	D
* 0	0	0	0	0

* 1	7	8	9	3
* 2	14	16	18	6
* 3	21	24	27	9
* 4	28	32	36	12
* 5	35	40	45	15

Applications of pandas in a real-life situation will enable you to perform the following:

- You can quickly delete some columns or even add some texts found within the Dataframe
- It will help you in data conversion
- Pandas can reassure you of getting the misplaced or missing data
- It has a powerful ability, especially in the grouping of other programs, according to their functionality.

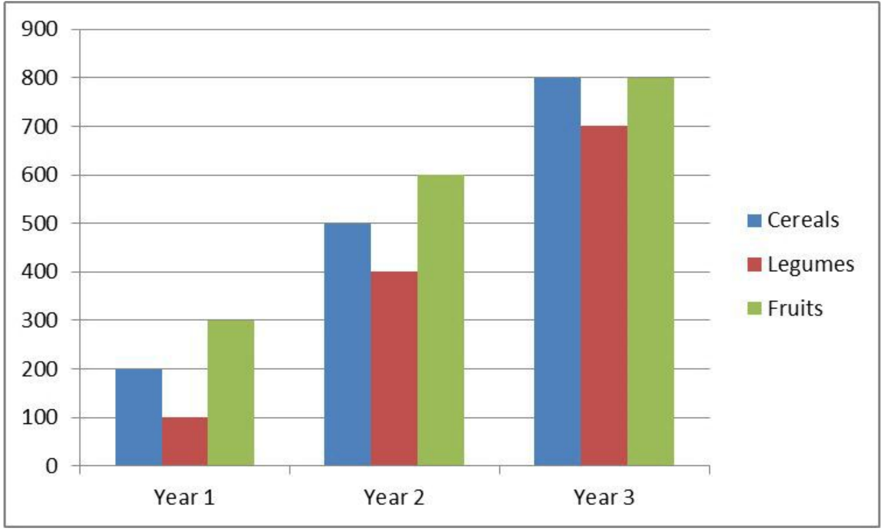
Matplotlib

This is another sophisticated and helpful data analysis technique that helps in data visualization. Its main objective is to advise the industry where it stands using the various inputs. You will realize that your production goals are meaningless when you fail to share them with different stakeholders. To perform this, Matplotlib comes in handy with the types of computation analysis required. Therefore, it is the only Python library that every scientist, especially the ones dealing with data prefers. This type of library has good looks when it comes to graphics and images. More so, many prefer using it in creating various graphs for data analyzation. However, the technological world has completely changed with new advanced libraries flooding the industry.

It is also flexible, and due to this, you are capable of making several graphs that you may need. It only requires a few commands to perform this.

In this Python library, you can create various diverse graphs, charts of all kinds, several histograms, and even scatterplots. You can also make non-Cartesian charts too using the same principle.

Diagrammatic explanations



The above graph highlights the overall production of a company within three years. It specifically demonstrates the usage of Matplotlib in data analysis. By looking at the diagram, you will realize that the production was high as compared to the other two years. Again, the company tends to perform in the production of fruits since it was leading in both years 1 and 2 with a tie in year 3. From the figure, you realize that your work of presentation, representation, and even analyzation has been made easier as a result of using this library. This Python library will eventually enable you to come up with good graphics images, accurate data, and much more. With the help of this Python library, you will be able to note down the year your production was high, thus, being in a position to maintain the high productivity season.

It is good to note that this library can export graphics and can change these graphics into PDF, GIF, and so on. In summary, on this library, the following tasks can be undertaken with much ease. They include:

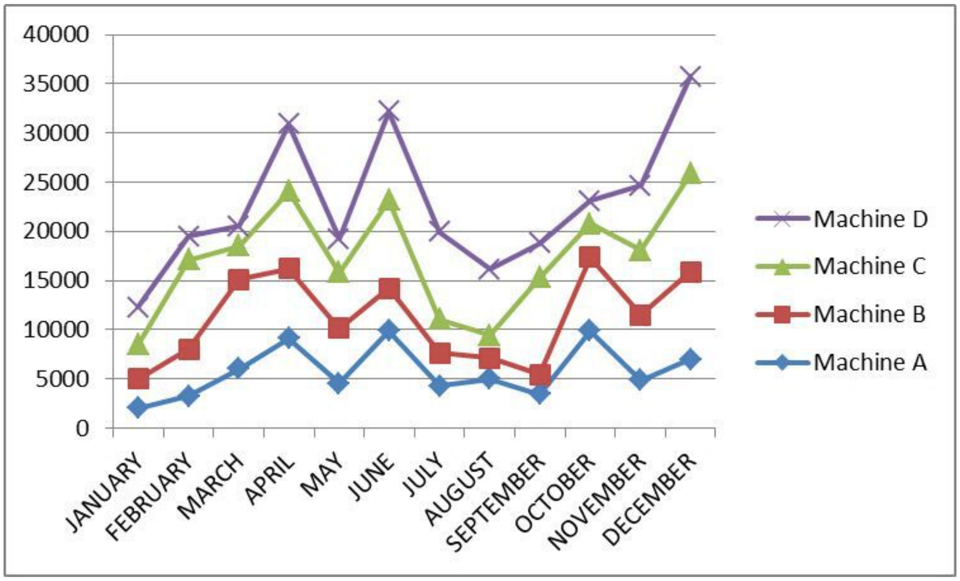
- Formation of line plots
- Scattering of plots
- Creations of beautiful bar charts and building up of histograms
- Application of various pie charts within the industry
- Stemming the schemes for data analysis and computations

- Being bin a position to follow up contours plots
- Usage of spectrograms and lastly
- Quiver plots creation.

Seaborn

Seaborn is also among the popular libraries within the Python category. Its main objective here is to help in visualization. It is important to note that this library borrows its foundation from Matplotlib. Due to its higher level, it is capable of various plots generation such as the production of heat maps, processing of violin plots, and also helping in generation of time series plots.

Diagrammatic Illustrations



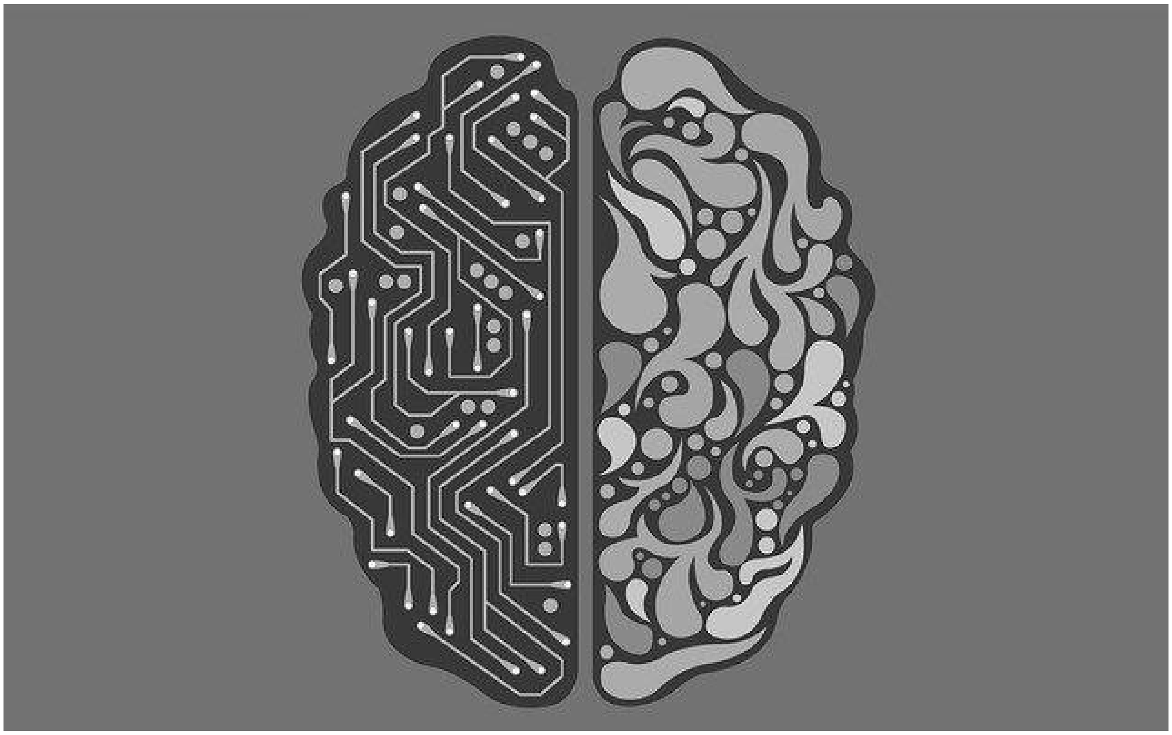
The above line graph clearly shows the performance of different machines the company is using. Following the diagram above, you can eventually deduce and make a conclusion on which machines the company can keep using to get the maximum yield. On most occasions, this evaluation method by the help of Seaborn library will enable you to predict the exact abilities of your different inputs. Again, this information can actually help for future reference in the case of purchasing more machines. Seaborn library also has the power to detect the performance of other variable inputs within the company. For example, the number of workers within the company can be easily identified with their corresponding working rate.

NumPy

This is a very widely used Python library. Its features enable it to perform multidimensional array processing. Also, it helps in the matrix processing. However, these are only possible with the help of an extensive collection of mathematical functions. It is important to note that this Python library is highly useful in solving the most significant computations within the scientific sector. Again, NumPy is also applicable in areas such as linear algebra, derivation of random number abilities used within industries, and more so Fourier transformation. NumPy is also used by other high-end Python libraries such as TensorFlow for Tensors manipulation. In short, NumPy is mainly for calculations and data storage. You can also export or load data to Python since it has those features that enable it to perform these functions. It is also good to note that this Python library is also known as numerical Python.

CHAPTER 11:

How Can Python Work With Machine Learning



When it comes to completing your own modules and algorithms in Machine Learning, you will find that there is just one language that can help you get it all done. Sure, other programmers may spend time talking about how another language is the best, but Python has the libraries and extensions, the features, the ease of use, and all of the power that you are looking for all in one place. There are so many reasons why you should choose to work with Python when you want to undertake a project that has Machine Learning, and we are going to explore some of those benefits here.

Python has the libraries that you need to handle your Machine Learning algorithms and get things done. We will take a look at some of the libraries that you are able to work with when it comes to the Python language and Machine Learning or data science in a bit. But there are quite a few that you can focus your attention on. This is great news for someone who is just getting started with this kind of process because you are certain to find the library that works the best for your needs. You can find Python libraries that work for all parts of the data analysis and Machine Learning process, from gathering up the data you need all the way to show the visuals at the end.

Python has a lot of power behind it that will ensure that we can get some of our data analysis modules done and ready to go. While we have spent some time talking about how easy Python is to learn and how all programmers can figure this one out, whether you are someone who is just getting started with coding or you have been doing it for a long time with other languages. Even though this is a simple language that we are able to learn and that we can catch onto quickly, that doesn't mean that we will have to give up on some of the power that is necessary for some of the coding we want to complete.

As you can imagine, Machine Learning is going to require some power in order to be successful. If you go in with a wimpy language that is not able to handle complex coding and more, you are going to end up with a lot of issues along the way. This is not a problem that you need to worry about with Python. Despite the ease of using and learning, you will find that this language still has all of the power and strength that you are looking for in order to get all of the different algorithms and modules that you want with Machine Learning done.

The Python language is easy enough to work with that even a beginner is able to learn even some of the more complicated parts that come with Machine Learning. If you have never done coding and this is the first step that you have taken to work on Machine Learning and making it work for you, it may seem like this is all too complicated to handle. But as you go through some of the coding that we have talked about already in this guidebook, and use that to help with Machine Learning and creating some of your own codes in the process, then you will quickly see that it is not as hard or as scary as you may have thought. Python will make all of this easier to handle.

Many of the algorithms that we will use to handle our data and make it do what we would like in the process work the best with the simple and easy to understand codes of Python. We already know that Python is going to come with a really easy to use and read code, and this can be beneficial when we are working with some of the more complicated parts that come with Machine Learning. It can be reassuring to see these recognizable codes in

Python when we have to handle our own data analysis, and learning some of the simplicity that comes with Python can help prepare us for some of the work that we need to do with Machine Learning.

Python is also able to work with a lot of the other coding languages out there. With some of our data analysis libraries, we will find that it is best to execute the work with another language. But because Python is able to handle doing some of the work, and it is compatible with other languages, we can still use the Python language in order to help out with writing the codes, but then switching it over, with the help of the Python library that we choose like TensorFlow, in order to have the stronger language finish things up.

The good news here is that the Python language is able to handle a lot of the work and the algorithms that you want to do with Machine Learning. And many times, you will not need to go through and make any adjustments or add Python to another language to get the work done. This makes it so much easier for us to write out the modules we want, while also training and testing them, to ensure that we can work with the data and really understand what it is trying to tell us.

Python is able to handle all of the different parts that come with Machine Learning and good data analysis. As we go through the idea of data analysis, you will find that there are actually a number of steps that need to be followed in order to figure out what is inside of all that data. For example, you will need to gather and sort through the data, you will need to clean it up and deal with all of the different parts that are there, and you will need to consider which algorithms to use and then train and test it.

All of this needs to be done before we are even able to put the data through and do the analysis. Then, when we get to the analysis, we may need to trial and error again to make sure that it is working the way that we want, and then we can use Python again in order to create some of the algorithms that we want with our analysis. All of these steps are important to the process of data analysis and will ensure that you are able to get the information that you need, and you can definitely work with Python in order to make this happen.

There may be other choices that we can make when it is time to handle Machine Learning and all that it entails, but you will find that one of the best languages out there that are able to handle all of the work, from finding and gathering the data to cleaning it off to getting the module done, and so much more, you will find that the Python language is able to take on the work.

CHAPTER 12:

Machine Learning and Data Science

What is a Data Scientist?

The funny thing is that this great value of the data contrasts with that precisely the data is the most abundant resource on the planet (it is estimated that 2.5 trillion bytes of new information is created per day). They don't seem easy to make things compatible. How is it possible that something so abundant is so valuable? Even if it was pure supply and demand, accumulating data should be trivial. And the complex thing is to process them.

Until relatively recently, we simply couldn't do it. At the end of the 90s, the field of Machine Learning began to take on an autonomous entity, our ability to work with immense amounts of data was reduced and the social irruption of the internet did the rest. For a few years, we have faced the first great 'democratization' of these techniques. And with that, the boom of data scientists: nobody wants to have an untapped gold mine.

In Search of a Data Scientist

The problem is that, suddenly, there has been a great demand for a profile that, until now, practically did not exist. Remember that you need statistical knowledge that a programmer does not usually have, and computer knowledge that a statistician does not usually even imagine.

Most of the time, it has been solved with self-taught training that completes the basic skills that the training program should have but does not have. That is why, today, we can find a great diversity of professional profiles in the world of data science. According to Burtch Works, 32% of active data scientists come from the world of mathematics and statistics, 19% from computer engineering, and 16% from other engineering.

How to Train

Degrees

Today, there are some double degrees in computer engineering and mathematics (Autonomous University of Madrid, Granada, Polytechnic University of Madrid, Polytechnic University of Catalonia, and Complutense, Murcia Autonomous University of Barcelona) or in computer science and statistics (University of Valladolid) that seem the best option if we consider this specialization.

Postgraduate

The postgraduate is a very diverse world. We can find postgraduate, masters, or specialization courses in almost all universities and a truly excessive private offer.

To give some examples, we have postgraduate degrees at the UGR, the UAB, the UAM, the UPM, or the Pompeu Fabra. However, in postgraduate course, it is more difficult to recommend a specific course.

What we can find in the postgraduate training that we cannot find in the previous training is the ‘business orientation’ component.

We must not forget that most of the work of data scientists is in companies that seek to make their databases profitable, because what market orientation is highly recommended.

In fact, many of the masters in ‘big data’ are offered by business schools such as OEI or Institute Empress.

MOOCs

One of the most interesting resources you can find are the MOOCs (you know, the Massive Open Online Courses). In fact, recently, we saw that this self-training option could have a lot of future.

Starting with the specialization program in Big Data of Coursera, we can find online courses from the best universities in the world, all this without mentioning the numerous tools to learn languages like Python or R.

What Languages Should Be Learned?

In reality, as any initiate knows, in programming the choice of one language or another is always complicated. In this election, they intervene from technical or formative factors to simple personal preferences. What is clear is that there are some languages more popular than others.

Although common sense tells us that each language is better for certain things, in practice there is a certain rivalry. Personally, I use R, but I usually recommend Python. Not only because it is prettier, but because it is multipurpose and that is always an advantage.

Other tools

A fireproof

- **Excel:** It is not a language and usually does not like those who work with professional data. Or so they say when asked why polls say otherwise: 59% percent of respondents routinely use excel. So, finally, the application of Office spreadsheets is still a lot of war.

The corporate brother and other languages and programs

- Some languages or environments enjoy some success driven by corporate inertia: it is the case of the classic MATLAB but progressively, it is losing weight and use up to only 6%.
- If we examine the surveys, we can find many more languages that obey more particular needs of the practice of data scientists (or the programs they use): Scala (17%), Slack (10%), Perl (12%), and C # (6%), Mahout (3%), Apache Hadoop (13%) or Java (23%).
- Also, although it is possible that we should talk about them separately, there are many specific programs (free or proprietary) that are used in data science with different uses. For example, we could talk about Tableau, Rapid Miner, or Weka.

The labor market: salaries and opportunities

Salaries, as in general in the world of software development, change a lot depending on the place, the functions, and the employer. However, right now, it is a well-paid expertise. On a general level and according to the annual KdNuggets survey, salaries/incomes average $ 141,000 for freelancers, 107,000 for employees, 90,000 for government workers, or in the non-profit sector; 70,000 dollars for work in universities.

However, these average salaries must be taken with great caution. While the average salary in the United States is between $ 103,000 and $ 131,000, in Western Europe it is between $ 54,000 and $ 82,000. In Spain, we are in similar numbers because, despite our (increasingly smaller) deficit of product companies, we have large companies (especially banks) that have turned in this field.

What differentiates data science from the rest of the development world is perhaps the shortage of professionals. This phenomenon makes salaries relatively inflated and, as more dater profiles appear, they adjust. Therefore, it can be said that it is time to get on the wave of data science. Within a couple of years, the market will have matured, and the opportunities will be elsewhere.

CHAPTER 13:

Decision Trees

Decision trees are built similarly to support vector machines, meaning they are a category of supervised Machine Learning algorithms that are capable of solving both regression and classification problems. They are powerful and used when working with a great deal of data.

You need to learn beyond the barebones basics so that you can process large and complex datasets. Furthermore, decision trees are used in creating random forests, which is arguably the most powerful learning algorithm. In this phase, we are going to exclusively focus on decision trees explicitly because of their popular use and efficiency.

An Overview on Decision Trees

Decision trees are essentially a tool that supports a decision that will influence all the other decisions that will be made. This means that everything from the predicted outcomes to consequences and resource usage will be influenced in some way. Take note that decision trees are usually represented in a graph, which can be described as some kind of chart where the training tests appear as a node. For instance, the node can be the toss of a coin, which can have two different results. Furthermore, branches sprout to individually represent the results, and they also have leaves, which are the class labels. Now you see why this algorithm is called a decision tree. The structure resembles an actual tree. As you probably guessed, random forests are exactly what they sound like. They are collections of decision trees, but enough about them.

Decision trees are one of the most powerful supervised learning methods you can use, especially as a beginner. Unlike other more complex algorithms, they are fairly easy to implement and they have a lot to offer. A decision tree can perform any common data science task, and the results you obtain at the end of the training process are highly accurate. With that in mind, let's analyze a few other advantages, as well as disadvantages, to gain a better understanding of their use and implementation.

Let's begin with the positives:

1. Decision trees are simple in design and, therefore, easy to implement even if you are a beginner without a formal education in data science or Machine Learning. The concept behind this algorithm can be summarized with a sort of a formula that follows a common type of programming statement: If this, then that, else that. Furthermore, the results you will obtain are very easy to interpret, especially due to the graphic representation.
2. The second advantage is that a decision tree is one of the most efficient methods in exploring and determining the most important variables, as well as discovering the connection between then. Also, you can build new features easily to gain better measurements and predictions. Don't forget that data exploration is one of the most important stages in working with data, especially when there is a large number of variables involved. You need to be able to detect the most valuable ones in order to avoid a time-consuming process, and decision trees excel at this.
3. Another benefit of implementing decision trees is the fact that they are excellent at clearing up some of the outliers in your data. Don't forget that outliers are noise that reduces the accuracy of your predictions. In addition, decision trees aren't that strongly affected by noise. In many cases, outliers have such a small impact on this algorithm that you can even choose to ignore them if you don't need to maximize the accuracy scores.

Finally, there's the fact that decision trees can work with both numerical as well as categorical variables. Decision trees, on the other hand, are proven to be versatile and handle a much more varied set of tasks.

As you can see, decision trees are powerful, versatile, and easy to implement, so why should we ever bother using anything else? As usual, nothing is perfect, so let’s discuss the negative side of working with this type of algorithm:

1. One of the biggest issues encountered during a decision tree implementation is overfitting. Take note that this algorithm tends to sometimes create very complicated decision trees that will have issues generalizing data due to their complexity. This is known as over fitting, and it is encountered when implementing other learning algorithms as well, however, not to the same degree. Fortunately, this doesn’t mean you should stay away from using decision trees. All you need to do is invest some time to implement certain parameter limitations to reduce the impact of overfitting.
2. Decision trees can have issues with continuous variables. When continuous numerical variables are involved, the decision trees lose a certain amount of information. This problem occurs when the variables are categorized. If you aren’t familiar with these variables, a continuous variable can be a value that is set to be within a range of numbers. For example, if people between ages 18 and 26 are considered of student age, then this numerical range becomes a continuous variable because it can hold any value between the declared minimum and maximum.

While some disadvantages can add to additional work in the implementation of decision trees, the advantages still outweigh them by far.

Classification and Regression Trees

Decision trees need to be divided into classification and regression trees. They handle different problems; however, they are similar in some ways since they are both types of decision trees.

Take note that classification decision trees are implemented when there’s a categorical dependent variable. On the other side, a regression tree is only implemented in the case of a continuous dependent variable. Furthermore, in the case of a classification tree, the result from the training data is the mode of the total relevant observations. This means that any observations that we cannot define will be predicted based on this value, which represents the observation which we identify most frequently.

Regression trees, on the other hand, work slightly differently. The value that results from the training stage is not the mode value, but the mean of the total observations. This way, the unidentified observations are declared with the mean value, which results from the known observations.

Both types of decision trees undergo a binary split, however, going from the top to bottom. This means that the observations in one area will spawn two branches that are then divided inside the predictor space. This is also known as a greedy approach because the learning algorithm is seeking the most relevant variable in the split while ignoring the future splits that could lead to the development of an even more powerful and accurate decision tree.

As you can see, there are some differences as well as similarities between the two. However, what you should note from all of this is that the splitting is what has the most effect on the accuracy scores of the decision tree implementation. Decision tree nodes are divided into sub-nodes, no matter the type of tree. This tree split is performed to lead to a more uniform set of nodes.
Now that you understand the fundamentals behind decision trees, let’s dig a bit deeper into the problem of overfitting.

The Overfitting Problem

You learned earlier that overfitting is one of the main problems when working with decision trees, and sometimes it can have a severe impact on the results. Decision trees can lead to a 100% accuracy score for the training set if we do not impose any limits. However, the major downside here is that overfitting creeps in when the algorithm seeks to eliminate the training errors, but by doing so, it actually increases the testing errors. This imbalance, despite the score, leads to terrible prediction accuracy in the result. Why does this happen? In this case, the decision trees grow many branches, and that’s the cause of overfitting. To solve this use, you need to impose limitations on how much the decision tree can develop and how many branches it can spawn. Furthermore, you can also prune the tree to keep it under control, much like how you would do with a real tree in order to make sure it produces plenty of fruit.

To limit the size of the decision tree, you need to determine new parameters during the definition of the tree. Let's analyze these parameters:

1. **min_samples_split:** The first thing you can do is change this parameter to specify how many observations a node will require to be able to perform the splitting. You can declare anything with a range of one sample to maximum samples. Just keep in mind that to limit the training model from determining the connections that are very common to a particular decision tree, you need to increase the value. In other words, you can limit the decision tree with higher values.
2. **min_samples_leaf:** This is the parameter you need to tweak to determine how many observations are required by a node, or in other words, a leaf. The overfitting control mechanism works the same way as for the samples split parameter.
3. **max_features:** Adjust this parameter in order to control the features that are selected randomly. These features are the ones that are used to perform the best split. To determine the most efficient value, you should calculate the square root of the total features. Just keep in mind that in this case, the higher value tends to lead to the overfitting problem we are trying to fix. Therefore, you should experiment with the value you set. Furthermore, not all cases are the same. Sometimes a higher value will work without resulting in overfitting.
4. **max_depth:** Finally, we have the depth parameter, which consists of the depth value of the decision tree. To limit the overfitting problem, however, we are only interested in the maximum depth value. Take note that a high value translates to a high number of splits, therefore a high amount of information. By tweaking this value, you will have control over how the training model learns the connections in a sample.

Modifying these parameters is only one aspect of gaining control of our decision trees in order to reduce overfitting and boost performance and accuracy. The next step after applying these limits is to prune the trees.

Pruning

This technique might sound too silly to be real; however, it is a legitimate Machine Learning concept that is used to improve your decision tree by nearly eliminating the overfitting issue. As with real trees, what pruning does is reduce the size of the trees in order to focus the resources on providing highly accurate results. However, you should keep in mind that the segments that are pruned are not entirely randomly selected, which is a good thing. The sections that are eliminated are those that don’t help with the classification process and don’t lead to any performance boosts. Less complex decision trees lead to a better-optimized model.

Conclusion

This is the end of the guidebook. The next milestone is to make the best use of your new-found wisdom of Python programming, Data Science, Data Analysis, and Machine Learning that have resulted in the birth of the powerhouse, which is the "Silicon Valley." So many companies, that span a lot of different industries, are able to benefit when they work with data analysis. This allows them to get a lot of the power and control that they want for their respective industries and will ensure that they will be able to really impress their customers and get some good results in the process. Learning how to use a data analysis is going to change the game in how you do business, as long as it is used in the proper manner.

This guidebook has been organized well to explore what data analysis is all about, and how we are able to use this for our benefits as well. There are a lot of business tools out there, but data analysis is designed to help us focus on finding the hidden patterns and insights that are in our data, making it easier to base our decisions on data, rather than intuition and guessing as we did in the past. And when it comes to making sure that we complete the data analysis in the right manner, nothing is better than working with the Python coding language to get things done.

There are so many aspects that need to come into play when we are working with our own data analysis, and it is important that we take the time to learn how these works, and how to put it all together. And that is exactly what we will do in this guidebook. When you are ready to learn more about Python data analysis, and all of the different parts that come together to help us with understanding our data and how to run our business, make sure to recheck this guide to help you.

Now that you have finished reading this book and mastered the use of Python programming, you are all set to start developing your own Python-based Machine Learning model as well as performing big data analysis using all the open sources readily available and explicitly described in this book. You can position yourself to use your deep knowledge and understanding of all the cutting edge technologies obtained from this book to contribute to the growth of any company and land yourself a new high paying and rewarding job!

Remember, however, – and wherever – you are teaching, foster a good learning environment. Trial and error is all part of learning when it comes to coding. Children like it when adults make mistakes, so when you know that they have a good understanding of what you are teaching them, then you can make mistakes too. Does your child notice this and pick up on it?

If you are working at home with your child, then work together rather than leaving them to play by themselves. If possible, involve other family members, too. Coding doesn't have to be a boring, dry, academic subject. Foster that creativity in your children by being creative yourself. Have fun and enjoy bonding with your child!