Python for Beginners

Author: Jared Viazanko

Python

Python is a very strong and easy to learn/read programming language used majorly for scripting, or task automation. Learning Python is a great way to get into programming and is extremely useful. In this tutorial you will learn how to install python along with and IDE, and the basic variables along with writing your first simple program.

Learning Objective

1. Basic understanding of Python
2. Able to read and comprehend basic code
3. How to install and setup python along with an IDE

Prerequisites

The "*" is a bulleted list. If this can only be done on a Windows computer, using Packet Tracer, or any other technology, list those prerequisites here.

- Windows
- Internet connection for downloads

Instructions

Guide the person through the steps. Starting a line with a period is an automatically numbered list.

1. install python from: https://www.python.org/downloads/ , the latest version will be what we are using. If you already have python installed, just upgrade from current version, if not the default values and paths for a fresh install is perfectly fine.
2. next we will be installing PyCharm from JetBrains from: https://www.jetbrains.com/pycharm/download/#section=windows , the community version is the one we will be using. Again the default settings are fine, changing the installation path is up to you incase you want it installed on a different drive.
3. next we will getting into "variables" since Python is a dynamically typed programming language there is not variable definition, the system just knows what kind of variable it is. for the first variable we will discuss strings:

```
string = "Hello"  # A string must always be accompanied by quotation marks, this is
how the system knows its the string variable.

number = 100  # defining numbers is just as easy as strings, you don't have to
define if it is a float, or a double, or integer python just knows.

Time for some expressions, we will get more into string manipulation later in the
guide when we get to if statements and for loops.

starting out, just type these expressions in;

x = 100  # defining the variables
y = 150

a = x+y  # all of these expression work the same as you would think, except module
which is the last one
print(a)
b = x-y
print(b)
c = x*y
print(c)
d = x/y
print(d)
e = x%y  # we wont be using this expression at all it just shoots the remainder
between the two
print(e)

make sure you run the script by either pressing "alt+shift+f10", or simply going to
the run tab and clicking run.
```

4. Now lets dabble into IF statements

If statements just like in most programming language are used as a logic statement to have the computer make a decision such as:

~~~~
 ```
a = 33
b= 200
~~~~

if b > a: print("b is greater than a") # since we know that 200 is greater then 33, and so does the computer, we print that its true # there is also something called and else if and else statement that pairs with if statements a that allows for an alternate output if something is or isn’t true.

the else if statement

a = 33 b = 33 if b > a: print("b is greater than a") elif a == b: # the elif is and else if statement that allows for another computation and output if the first one fails. print("a and b are equal")

the if else statement

a = 200 b = 33 if b > a: print("b is greater than a") elif a == b: print("a and b are equal") else: # as you can see the else statement is used properly and the if else statements prints everything correctly. print("a is greater than b")

That’s about it for if and else if statements. Just play around with your own variables and logic to get a further understanding, its pretty straight forward until you get into more complicated programming.

~~~
 ```
 ```
. Now For loops
~~~

A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). With a for loop we can execute a set of statements for each item in a list, or data set.

Eaxmples:

~~~
 ```
# we will print every super hero from the list of super hero's
superheros = ["batman", "robinhood", "superman"]
for x in superheros:
  print(x)
~~~

looping through a strings, this prints every character in a string.

for x in "Aquaman": print(x) # Looping can be used in conjunction with if and else if statements in order to create more logic then need and continue to run through a list, array, or data set. This can be done in many different ways including using special commands such as "break, continue, and range" # quick examples:

break statements does what you’d expect, it’ll break your for loop

```
superheros = ["batman", "robinhood", "superman"] for x in superheros: if x == "robinhood": break print(x)
```

continue statements does as the name implies, it continues looping once a condition is met, but allows you yo another action such as printing to console

```
superheros = ["batman", "robinhood", "superman"] for x in superheros: if x == "robinhood": print(x + " sucks") continue print(x)
```

range is used to print numbers from within a range pretty simple

```
for x in range(6): print(x)
```

using Else in a For Looping

```
for x in range(6): print(x) else: print("Finally finished!")
```

you can also get fancy with it a use nested for loops in order to manipulate data further, but this gets a little complicated for new programmers

adj = ["red", "big", "tasty"] fruits = ["apple", "banana", "cherry"]

for x in adj: for y in fruits: print(x, y)

that example just prints red in front of all the fruits and continues through the same concept with big and tasty.

~~~
```
~~~

Thats all the basic knowledge one should have of python to begin programming, the best way to learn python like any other thing is life is to just get out there and do it. With this basic knowledge you should be able to jump into bigger problems and work them out with your own logic, or at least be able to understand and read what’s going on.

Challenge

create a simple calculator using your new found knowledge.

Reflection

Provide some thought questions that help the learner make sense of how the tutorial fits in the bigger picture.

asciidoctor-pdf tutorial-template.adoc