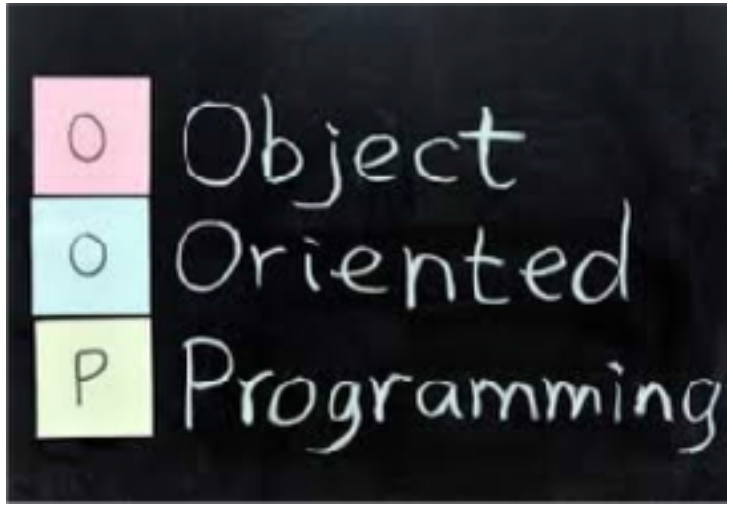
PYTHON

WHAT IS PYTHON?

IT'S AN INTERPRETED LANGUAGE WHICH PHILOSOPHY IS TO FOCUS IN A SYNTAX THAT FAVORS A READABLE CODE.

-OBJECT ORIENTED
-IMPERATIVE PROGRAMMING
-DYNAMIC PROGRAMMING
-MULTIPLATFORM



WHAT IS PYTHON?

IT IS VERY ATTRACTIVE FOR RAPID APPLICATION DEVELOPMENT.

PYTHON'S SIMPLE, EASY TO LEARN SYNTAX EMPHASIZES READABILITY AND THEREFORE REDUCES THE COST OF PROGRAM MAINTENANCE.

OPEN SOURCE AND MANAGED BY THE PYTHON SOFTWARE FOUNDATION.

CONCEIVED IN LATE 80'S BY GUIDO VAN ROSSUM IN THE NETHERLANDS .

WHY PYTHON?

-THE MOST POPULAR INTRODUCTORY LANGUAGE IN U.S. UNIVERSITIES

-THE FOURTH MOST POPULAR LANGUAGE ACCORDING TO AN IEEE SURVEY BEHIND OLD CLASSICS JAVA, C AND C++.

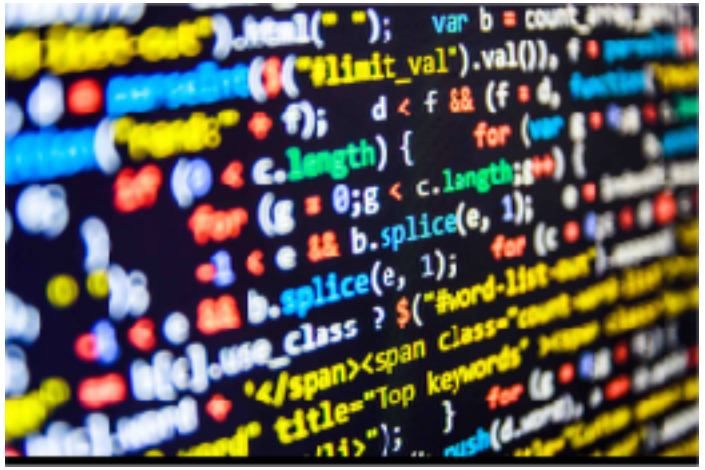
-INCREASED PRODUCTIVITY

-NO COMPILATION STEP

-EDIT-TEST-DEBUG CYCLE IS FAST

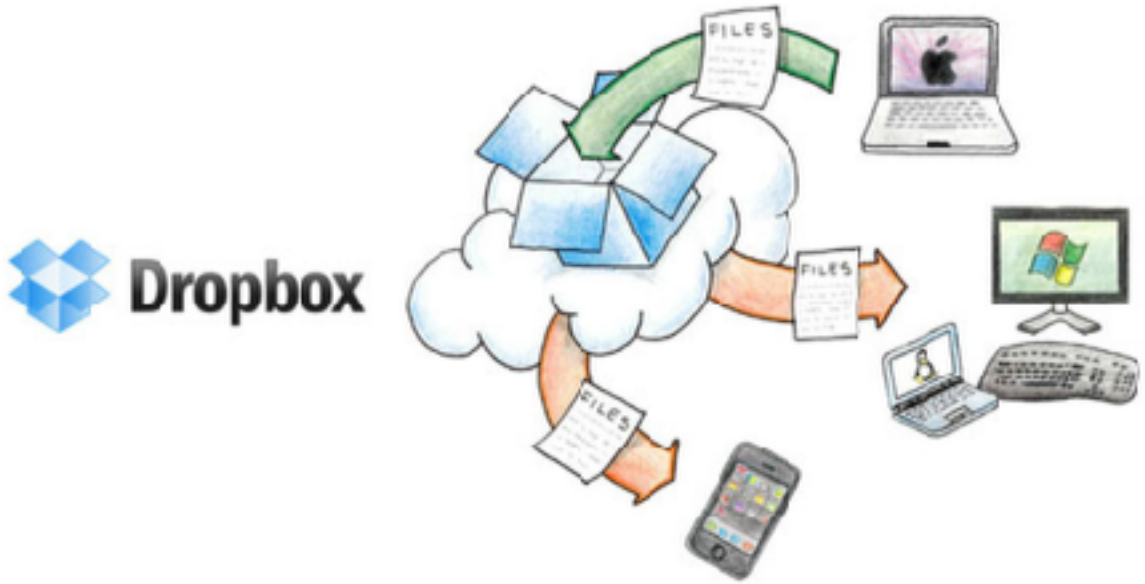
WHO USES IT AND HOW?

DON'T THINK THAT BECAUSE PYTHON IS EASY TO USE IT'S A WEAK LANGUAGE. PYTHON IS INCREDIBLY POWERFUL - THERE'S A REASON COMPANIES LIKE GOOGLE, DROPBOX, SPOTIFY AND NETFLIX USE IT.



WHO USES IT AND HOW?

DROPBOX DESKTOP CLIENT IS WRITTEN ENTIRELY IN PYTHON, THE SERVER-SIDE CODE IS IN PYTHON AS WELL MAKING IT THE MAJORITY LANGUAGE USED AT THE COMPANY.



WHO USES IT AND HOW?

GOOGLE USES A MIX OF LANGUAGES, WITH JAVA, C++ AND PYTHON. EARLY ON AT GOOGLE, THERE WAS AN ENGINEERING DECISION TO USE “PYTHON WHERE WE CAN, C++ WHERE WE MUST.” IT WAS USED FOR PARTS THAT REQUIRED RAPID DELIVERY AND MAINTENANCE.

WHO USES IT AND HOW?

SPOTIFY USES JAVA HEAVILY, BUT USES PYTHON FOR THINGS LIKE THE WEB API AND THEIR INTERACTIVE API CONSOLE, WHICH LETS DEVELOPERS EXPLORE ENDPOINTS WITH AN EASY-TO-USE INTERFACE. ALSO USES IT FOR DATA ANALYTICS AND OTHER NON- CUSTOMER FACING PROCESSES.

WHO USES IT AND HOW?

NETFLIX USES A MIX OF JAVA, SCALA, AND PYTHON, AND GIVES DEVELOPERS AUTONOMY WHEN CHOOSING
WHICH LANGUAGE FITS THE PROBLEM BEST. THEY HEAVILY USE PYTHON IN THEIR REAL-TIME ANALYTICS GROUP.

COMPARISON WITH OTHER LANGUAGES

PYTHON VS JAVA

-PYTHON SNIPPET TAKES VERY LESS MEMORY IN COMPARISON TO JAVA.

-PYTHON SNIPPET FOR A PARTICULAR FUNCTIONALITY REQUIRES LESS NUMBER OF LINES.

Example:

Snippet | Python

```
Print ("GoodBye World!")
```

Snippet | JAVA

```
public class GoodByeWorld {

    public static void main(String[] args)

    {

        System.out.println("GoodBye World!");

    }

}
```

Output-

```
GoodBye World!
>>>
```

COMPARISON WITH OTHER LANGUAGES

PYTHON VS C++

-PYTHON PROVIDES MUCH FLEXIBILITY IN CALLING FUNCTIONS AND RETURNING VALUES THAN C++

-PYTHON USES GARBAGE COLLECTOR WHEREAS C++ DOESN'T

Example:

Snippet | Python

```
Print ("GoodBye World!")
```

Snippet | C++

```
#include <iostream>
void main()
{
        cout << "GoodBye World!";
}
```

Output-

```
GoodBye World!
>>>
```

COMPARISON WITH OTHER LANGUAGES

PYTHON VS PHP

-PYTHON IS MUCH MORE MAINTAINABLE THAN PHP.

-PYTHON IS MUCH MORE STABLE AND UPWARD COMPATIBLE THAN PHP.

-PYTHON SNIPPET EXECUTES MUCH FASTER THAN PHP. (95% FASTER)

Example:

Snippet | Python

```
Print ("GoodBye World!")
```

Snippet | PHP

```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
 <?php echo '<p> GoodBye World!</p>'; ?>
 </body>
</html>
```

Output-

```
GoodBye World!
>>>
```

SYNTAX

INTERACTIVE MODE PROGRAMMING

INVOKING THE INTERPRETER WITHOUT PASSING A SCRIPT FILE AS A PARAMETER.

SYNTAX

SCRIPT MODE PROGRAMMING

INVOKING THE INTERPRETER WITH A SCRIPT PARAMETER BEGINS EXECUTION OF THE SCRIPT AND CONTINUES UNTIL THE SCRIPT IS FINISHED. WHEN THE SCRIPT IS FINISHED, THE INTERPRETER IS NO LONGER ACTIVE.

SYNTAX

PYTHON IDENTIFIERS

A PYTHON IDENTIFIER IS A NAME USED TO IDENTIFY A VARIABLE, FUNCTION, CLASS, MODULE OR OTHER OBJECT. AN IDENTIFIER STARTS WITH A LETTER A TO Z OR a to z OR AN UNDERSCORE (_) FOLLOWED BY ZERO OR MORE LETTERS, UNDERSCORES AND DIGITS (0 TO 9).

PYTHON DOES NOT ALLOW PUNCTUATION CHARACTERS SUCH AS @, $, AND % WITHIN IDENTIFIERS. PYTHON IS A CASE SENSITIVE PROGRAMMING LANGUAGE. THUS, MANPOWER AND manpower ARE TWO DIFFERENT IDENTIFIERS IN PYTHON.

SYNTAX

PYTHON IDENTIFIERS

HERE ARE NAMING CONVENTIONS FOR PYTHON IDENTIFIERS:

-CLASS NAMES START WITH AN UPPERCASE LETTER. ALL OTHER IDENTIFIERS START WITH A LOWERCASE LETTER.

-STARTING AN IDENTIFIER WITH A SINGLE LEADING UNDERSCORE INDICATES THAT THE IDENTIFIER IS PRIVATE.

-STARTING AN IDENTIFIER WITH TWO LEADING UNDERSCORES INDICATES A STRONGLY PRIVATE IDENTIFIER.

-IF THE IDENTIFIER ALSO ENDS WITH TWO TRAILING UNDERSCORES, THE IDENTIFIER IS A LANGUAGE-DEFINED SPECIAL NAME.

SYNTAX

RESERVED WORDS

and	exec	not
assert	finally	or
break	for	pass
class	from	print
continue	global	raise
def	if	return
del	import	try
elif	in	while
else	is	with
except	lambda	yield

SYNTAX

LINES AND INDENTATION

PYTHON PROVIDES NO BRACES TO INDICATE BLOCKS OF CODE FOR CLASS AND FUNCTION DEFINITIONS OR FLOW CONTROL. BLOCKS OF CODE ARE DENOTED BY LINE INDENTATION, WHICH IS RIGIDLY ENFORCED. THE NUMBER OF SPACES IN THE INDENTATION IS VARIABLE, BUT ALL STATEMENTS WITHIN THE BLOCK MUST BE INDENTED THE SAME AMOUNT.

```
if True:
    print "True"
else:
  print "False"
```

However, the following block generates an error –

```
if True:
    print "Answer"
    print "True"
else:
    print "Answer"
  print "False"
```

SYNTAX

MULTI-LINE STATEMENTS

STATEMENTS IN PYTHON TYPICALLY END WITH A NEW LINE. PYTHON DOES, HOWEVER, ALLOW THE USE OF THE LINE CONTINUATION CHARACTER (\) TO DENOTE THAT THE LINE SHOULD CONTINUE.

STATEMENTS CONTAINED WITHIN THE [], {}, OR () BRACKETS DO NOT NEED TO USE THE LINE CONTINUATION CHARACTER.

```
total = item_one + \
        item_two + \
        item_three
```

```
days = ['Monday', 'Tuesday', 'Wednesday',
        'Thursday', 'Friday']
```

SYNTAX

QUOTATION IN PYTHON

PYTHON ACCEPTS SINGLE (') , DOUBLE (") AND TRIPLE (''' OR """) QUOTES TO DENOTE STRING LITERALS, AS LONG AS THE SAME TYPE OF QUOTE STARTS AND ENDS THE STRING.

THE TRIPLE QUOTES ARE USED TO SPAN THE STRING ACROSS MULTIPLE LINES.

```
word = 'word'
sentence = "This is a sentence."
paragraph = """This is a paragraph. It is
made up of multiple lines and sentences."""
```

SYNTAX

COMMENTS IN PYTHON

A HASH SIGN (#) THAT IS NOT INSIDE A STRING LITERAL BEGINS A COMMENT. ALL CHARACTERS AFTER THE # AND UP TO THE END OF THE PHYSICAL LINE ARE PART OF THE COMMENT AND THE PYTHON INTERPRETER IGNORES THEM.

```
# This is a comment.
# This is a comment, too.
# This is a comment, too.
# I said that already.
```

SYNTAX

MULTIPLE STATEMENT GROUPS AS SUITES

A GROUP OF INDIVIDUAL STATEMENTS, WHICH MAKE A SINGLE CODE BLOCK ARE CALLED SUITES IN PYTHON. COMPOUND OR COMPLEX STATEMENTS, SUCH AS IF, WHILE, DEF, AND CLASS REQUIRE A HEADER LINE AND A SUITE.

HEADER LINES BEGIN THE STATEMENT (WITH THE KEYWORD) AND TERMINATE WITH A COLON (:) AND ARE FOLLOWED BY ONE OR MORE LINES WHICH MAKE UP THE SUITE.

```
if expression :
   suite
elif expression :
   suite
else :
   suite
```

SOME PYTHON LIBRARIES

1.WXPYTHON A GUI TOOLKIT FOR PYTHON. YOU WILL REALLY LOVE IT.

2. PILLOW PYTHON IMAGING LIBRARY IS A MUST HAVE FOR ANYONE WHO WORKS WITH IMAGES.

3. NUMPY IT PROVIDES SOME ADVANCED MATH FUNCTIONALITIES TO PYTHON.

4.SCIPY WHEN WE TALK ABOUT NUMPY THEN WE HAVE TO TALK ABOUT SCIPY. IT IS A LIBRARY OF ALGORITHMS AND MATHEMATICAL TOOLS FOR PYTHON AND HAS CAUSED MANY SCIENTISTS TO SWITCH FROM RUBY TO PYTHON.

5.MATPLOTLIB A NUMERICAL PLOTTING LIBRARY. IT IS VERY USEFUL FOR ANY DATA SCIENTIST OR ANY DATA ANALYZER.

PIP

PIP IS A PACKAGE MANAGEMENT SYSTEM USED TO INSTALL AND MANAGE SOFTWARE PACKAGES WRITTEN IN PYTHON.

PYTHON 2.7.9 AND LATER, AND PYTHON 3.4 AND LATER INCLUDE PIP BY DEFAULT.

PIP IS A RECURSIVE ACRONYM THAT CAN STAND FOR EITHER "PIP INSTALLS PACKAGES" OR "PIP INSTALLS PYTHON"