UNIT-1

PAGE NO.:
DATE: / /

INTRODUCTION TO PYTHON

* Python is a high-level, interpreted, interactive and object-oriented scripting language.

Python is High-level programming Language:-

Python is designed to be highly readable. It uses English keywords Frequently. Where as other languages use punctuation and it has Fewer Syntactical constructions than other languages.

Python is Interpreted:-

Python is processed at runtime by the interpreter. You don't need to compile your program before executing it.

Python is Interactive:-

You can actually sit at a python prompt and interact with the interpreter drictly to write your programs.

PAGE NO.:
DATE: / /

Python is object-oriented:-

Python supports object-oriented style or technique of programming that encapsulates code within objects.

HISTORY OF PYTHON

* Python was developed by Guido van Rossum in the late 1980's and early 1990's at the National Research Intitute For Mathematics and computer Science in the Netherlands.

* Python is derived From many other languages, including ABC, modula-3, C, C++, Algol-68, SmallTalk, Unix shell and other scripting languages.

→ NEED OF PYTHON PROGRAMMING

1. Software Quality

* Python Focus on readability, coherence and software quality in General sets it apart From other tools in the scripting world.

PAGE NO.:
DATE: / /

* Python has deep support for more advanced software reuse mechanisms, such as object-oriented programming [OOP].

2. Developer Productivity

* Python code is typically one-third to one-fifth the size of equivalent C++ or Java code. That means there is less to type, less to debug and less to maintain.

* Python ~~Prg~~ Programs also run immediately without the lengthy compile and link steps required by some other tools, further boosting programmer speed.

3. Support Libraries :-

* Python comes with a large collection of prebuilt and portable functionality, known as the standard library. This library supports an array of application level programming tasks from text pattern matching to Network Scripting.

* Python can be extended with both homegrown libraries and a vast collection of third-party application support software.

Python's third-party domain offers tools for website construction, numeric programming, serial port access, game development and much more.

4. Easy to Understand :-

* Being a very high-level language, Python reads like English, which takes a lot of Syntax-learning stress off coding beginners.

* Python handles a lot of complexity for you, so it is very beginner-friendly in that it allows beginners to focus on learning programming concepts and not have to worry about to much details.

5. Very Flexible :-

* As a dynamically typed language, Python is really flexible. This means there are no hard rules on how to build features and you'll have more flexibility. solving problems using different methods.

PAGE NO.:
DATE: / /

* Python is also more forgiving of errors, so you'll still be able to compile and run your program until you hit the problematic part.

→ APPLICATIONS OF PYTHON

* The following are the application of Python in a wide range of areas :-

1. Web Application
2. Desktop Application
3. Database Application
4. Web Scraping
5. Web Mapping
6. Data Analysis
7. Interactive Web Visualization.
8. Computer Vision For image and video processing
9. Object oriented programming.

PAGE NO.:
DATE: / /

→ PYTHON IDENTIFIERS:-

* A Python identifier is a name used to identify a variable, Function, class, module or other objects.

* An identifier starts with a letter A-Z or a-z or an underscore (_) Followed by zero or more letters underscore and digits (0 to 9)

* The Following are naming conventions For Python identifiers:-

i) Class names start with an uppercase letter. All other identifiers start with a lower-case letters.

ii) Starting an identifier with a single leading underscore indicates that the identifier is private.

iii) Starting an identifier with two leading underscores indicates a strongly private identifier.

iv) If the identifier also ends with two trailing underscores, the identifier is a language-defined special name.

PAGE NO.:
DATE: / /

PYTHON KEYWORDS

* Keywords are reserved words and you cannot use them as constant or variable or any other identifier names.

* All the Python keywords contain lowercase letters only.

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

PAGE NO.:
DATE: / /

VARIABLES

* Variables are nothing but reserved memory locations to store values. this means when you create a variable, you reserve some space in memory.

* Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory.

Assigning values to variables

* Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable the equal sign (=) is used to assign value to variables.

```
Eg: a = 15      # An Integer assignment
    b = 3.12    # A Float
    c = "RISE"  # A string.
```

```
Eg: a = b = c = 2   # Assign single value to several variables.
```

```
Eg  a, b, c = 15, 3.12, "RISE"
    # Assign multiple objects to multiple variables.
```