

"Wet your feet..."



- What is Python?
- Reasons for Popularity
- What sets Python apart?
- Where is Python Used?
- Who uses Python today?
- Programming Paradigms

- Functional Programming Model
- Procedural Programming Model
- Object-oriented Programming Model
- Event-driven Programming Model
- Exercises



## What is Python?

- Python is a high-level programming language created by Guido Van Rossum - fondly known as Benevolent Dictator For Life.
- Python was first released in 1991. Today Python interpreters are available for many Operating Systems including Windows and Linux.
- Python programmers are often called Pythonists or Pythonistas.

# **Reasons for Popularity**

• There are several reasons for Python's popularity. These include:

#### (a) Free:

- Python is free to use and distribute and is supported by community.
- Python interpreter is available for every major platform.

#### (b) Software quality:

- Better than traditional and scripting languages.
- Readable code, hence reusable and maintainable.
- Support for advance reuse mechanisms.

## (c) Developer productivity:

- Much better than statically typed languages.
- Much smaller code.
- Less to type, debug and maintain.
- No lengthy compile and link steps.

# (d) Program portability:

- Python programs run unchanged on most platforms.
- Python runs on every major platform currently in use.
- Porting program to a new platform usually need only cut and paste. This is true even for GUI, DB access, Web programming, OS interfacing, Directory access, etc.

# (e) Support libraries:

- Strong library support from Text pattern matching to networking.
- Vast collection of third party libraries.
- Libraries for Web site construction, Numeric programming,
   Game development, Machine Learning etc.

#### (f) Component integration:

- Can invoke C, C++ libraries and Java components.
- Can communicate with frameworks such as COM, .NET.
- Can interact over networks with interfaces like SOAP, XML-RPC, CORBA.
- With appropriate glue code, Python can subclass C++, Java, C#.
   classes, thereby extending the reach of the program.
- Popularly used for product customization and extension.

#### (g) Enjoyment:

- Fase of use.
- Built-in toolset.
- Programming becomes pleasure than work.

# What sets Python apart?

#### (a) Powerful:

- Dynamic typing.
- No variable declaration.
- Automatic allocation and Garbage Collection.
- Supports classes, modules and exceptions.
- Permits componentization and reuse.
- Powerful containers Lists, Dictionaries, Tuples, etc.

# (b) Ready-made stuff:

- Support for operations like joining, slicing, sorting, mapping, etc.
- Powerful library.
- Large collection of third-party utilities.

# (c) Ease of use:

- Type and run.
- No compile and link steps.
- Interactive programming experience.
- Rapid turnaround.
- Programs are simpler, smaller and more flexible.

# Where is Python used?

- Python is used for multiple purposes. These include:
- (a) System programming
- (b) Building GUI applications
- (c) Internet scripting

- (d) Component integration
- (e) Database programming
- (f) Rapid prototyping
- (g) Numeric and Scientific programming
- (h) Game programming
- (i) Robotics programming

## Who uses Python today?

- Many organizations use Python for varied purposes. These include:
- (a) Google In web search system
- (b) YouTube Video Sharing service
- (c) Bit-torrent Peer to Peer file sharing system
- (d) Intel, HP, Seagate, IBM, Qualcomm Hardware testing
- (e) Pixar, Industrial Light and Magic Movie animation
- (f) JP Morgan, Chase, UBS Financial market forecasting
- (g) NASA, FermiLab Scientific programming
- (h) iRobot Commercial robot vacuum cleaners
- (i) NSA Cryptographic and Intelligence analysis
- (j) IronPort Email Servers

## **Programming Paradigms**

- Paradigm means organization principle. It is also known as model.
- Programming paradigm/model is a style of building the structure and elements of computer programs.
- There exist many programming models like Functional, Procedural, Object-oriented, Event-driven, etc.
- Many languages facilitate programming in one or more paradigms.
   For example, Python supports Functional, Procedural, Object-oriented and Event-driven programming models.

- There are situations when Functional programming is the obvious choice, and other situations were Procedural programming is the better choice.
- Paradigms are not meant to be mutually exclusive. A single program may use multiple paradigms.

## **Functional Programming Model**

- Functional programming decomposes a problem into a set of functions. These functions provide the main source of logic in the program.
- Functions take input parameters and produce outputs. Python provides functional programming techniques like lambda, map, reduce and filter. These are discussed in Chapter 15.
- In this model computation is treated as evaluation of mathematical functions. For example, to get factorial value of a number, or n<sup>th</sup> Fibonacci number we can use the following functions:

```
factorial(n) = 1 if n == 0

= n * factorial(n - 1) if n > 0

fibo(n) = 0 if n = 0

= 1 if n = 1

= fibo(n - 2) + fibo(n - 1) if n > 1
```

- The output value of a function depends only on its arguments, so calling a function with the same value for an argument always produces the same result. As a result, it is a good fit for parallel execution.
- No function can have side effects on other variables (state remains unaltered).
- Functional programming model is often called a 'Declarative' programming paradigm as programming is done with expressions or declarations instead of statements.

# **Procedural Programming Model**

 Procedural programming solves the problem by implementing one statement (a procedure) at a time. Thus it contains explicit steps that are executed in a specific order.

- It also uses functions, but these are not mathematical functions like the ones used in functional programming. Functional programming focuses on expressions, whereas Procedural programming focuses on statements.
- The statements don't have values and instead modify the state of some conceptual machine.
- Same language expression can result in different values at different times depending on the global state of the executing program. Also, the functions may change a program's state.
- Procedural programming model is often called 'Imperative' programming as it changes state with an explicit sequence of statements.

## **Object-oriented Programming Model**

- This model mimics the real world by creating inside the computer a mini-world of objects.
- In a University system objects can be VC, Professors, Non-teaching staff, students, courses, semesters, examinations, etc.
- Each object has a state (values) and behavior (interface/methods).
   Objects get state and behavior based on the class from which it created.
- Objects interact with one another by sending messages to each other, i.e. by calling each other's interface methods.

## **Event-driven Programming Model**

- This model is popularly used for programming GUI applications containing elements like windows, check boxes, buttons, comboboxes, scroll bars, menus, etc.
- When we interact with these elements (like clicking a button, or moving the scrollbar or selecting a menu item) events occur and these elements emit messages. There are listener methods which are registered with these GUI elements which react to these events.
- Since there is no guaranteed sequence in which events may occur (based on how we interact with GUI elements), the listeners should be able to handle them in asynchronous manner.



#### [A] Answer the following:

- (a) Mention 5 fields in which Python is popularly used.
- (b) Where is event-driven programming popularly used?
- (c) Why Python is called portable language?
- (d) What is the single most important feature of different programming models discussed in this chapter?
- (e) Which of the following is not a feature of Python?
  - Static typing
  - Dynamic typing
  - Run-time error handling through error numbers
  - Library support for containers like Lists, Dictionaries, Tuples
- (f) Give an example application of each of the following programming models:
  - Functional model
  - Procedural model
  - Object-oriented model
  - Event-driven model
- [B] State whether the following statements are True or False:
- (a) Python is free to use and distribute.
- (b) Same Python program can work on different OS microprocessor combinations.
- (c) It is possible to use C++ or Java libraries in a Python program.
- (d) In Python type of the variable is decided based on its usage.
- (e) Python cannot be used for building GUI applications.
- (f) Python supports functional, procedural, object-oriented and event-driven programming models.
- (g) GUI applications are based on event-driven programming model.

(h)	Functional programming model coobjects.	ons	ists	of i	nteractio	on of	multiple
[C]	Match the following pairs:						
	<ul><li>a. Functional programming</li><li>b. Event-driven programming</li><li>c. Procedural programming</li><li>d. OOP</li></ul>	<ol> <li>GUI element based interaction</li> <li>Interaction of objects</li> <li>Statements</li> <li>Maths-like functions</li> </ol>					
[D]	Fill in the blanks:						
(a)	Functional programming paradig programming model.	m	is	also	known	as	
(b)	Procedural programming paradig programming model.	ŗm	is	also	known	as	
(c)	Python was created by						

(d) Python programmers are often called \_\_\_\_\_\_.