BCA SEMESTER - II

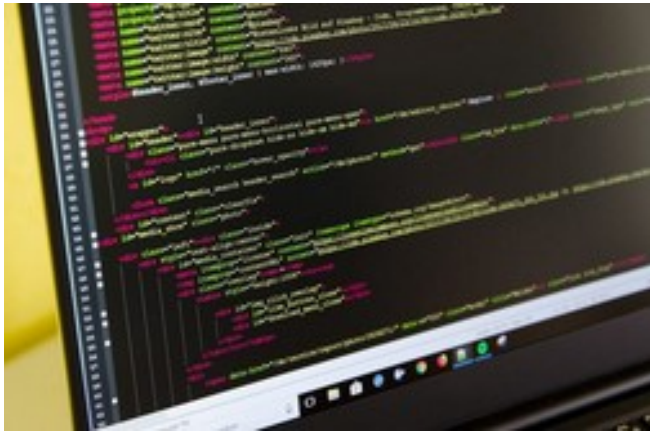
0302203
HISTORY OF COMPUTING

UNIT – 3
HISTORY OF PROGRAMMING LANGUAGES

- Dr. Disha Shah

Index

- **History of Programming Languages**
  - Ada Lovelace’s machine algorithm
  - Machine Language
  - Symbolic Programming Language
  - Lower Level Languages
  - Higher Level Languages
  - FORTRAN
  - ALGOL (Algorithmic Language)
  - LISP (List Processor)
  - COBOL (Common Business Oriented Language)



Index

- **Mid 1900s (Any 3)**
  - BASIC
  - PASCAL
  - Smalltalk,
  - C,
  - PROLOG
  - Ada
  - C++
  - Python
  - Ruby
  - Java,
  - PHP
  - Java Script
- **Mid 2000s (Any 3)**
  - Scala
  - Go
  - Dart
  - Swift
  - AlphaGo
  - Rust
  - Kotlin
  - Flutter
  - NLP

Python

- Python is a high-level, interpreted, interactive and object-oriented scripting 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.
- It was created by Guido van Rossum, and released in 1991.
- It is used for:
  - web development (server-side)
  - software development
  - Mathematics
  - system scripting

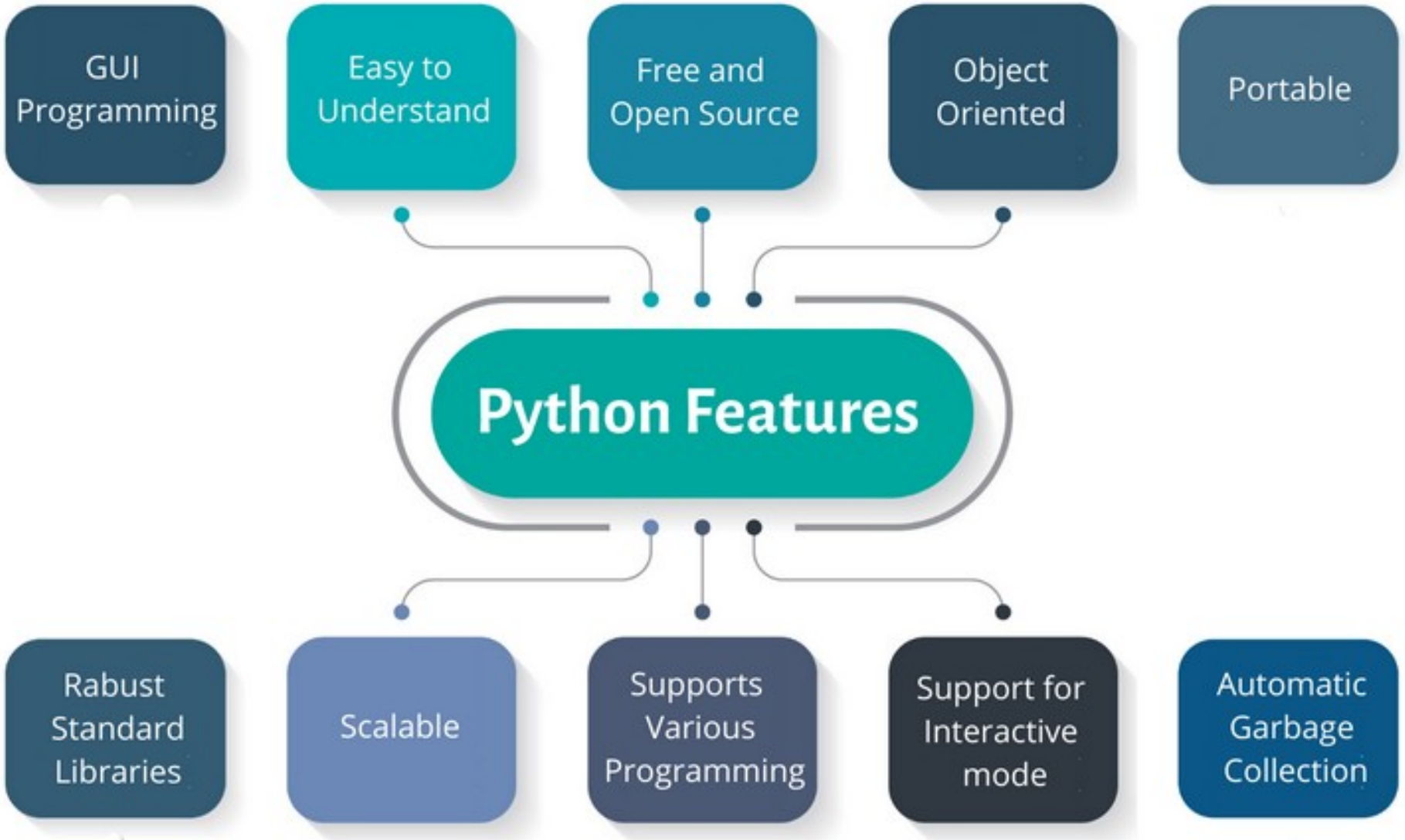
Python

- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

- Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
- Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.
- Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).
- Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features



Python Features

- Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- Easy-to-read − Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain − Python's source code is fairly easy-to-maintain.
- A broad standard library − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

Python Features

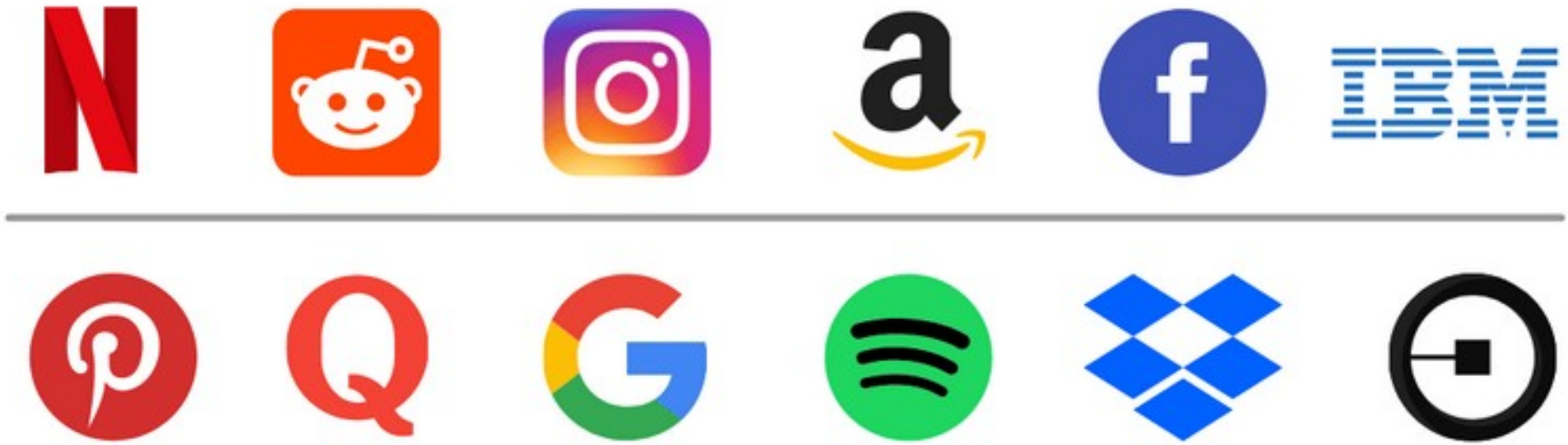
- Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- Databases − Python provides interfaces to all major commercial databases.
- GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- Scalable − Python provides a better structure and support for large programs than shell scripting.

What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Python Applications

Companies using Python



Python Editors

- Python code editors are designed for the developers to code and debug program easily. Using these Python IDEs(Integrated Development Environment), you can manage a large codebase and achieve quick deployment.
- 1) PyCharm
  - PyCharm is a cross-platform IDE used for Python programming. It is one of the best Python IDE editor that can be used on Windows, macOS, and Linux. This software contains API that can be used by the developers to write their own Python plugins so that they can extend the basic functionalities.

Python Editors

- 2) Kite

  Kite is IDE for Python that automatically completes multiple line codes. This editor supports more than 16 languages. It helps you to code faster with no hassle.

- 3) Spyder

  Spyder is a scientific integrated development environment written in Python. This software is designed for and by scientists who can integrate with Matplotlib, SciPy, NumPy, Pandas, Cython, IPython, SymPy, and other open-source software. Spyder is available through Anaconda (open-source distribution system) distribution on Windows, macOS, and Linux

- 4) IDLE

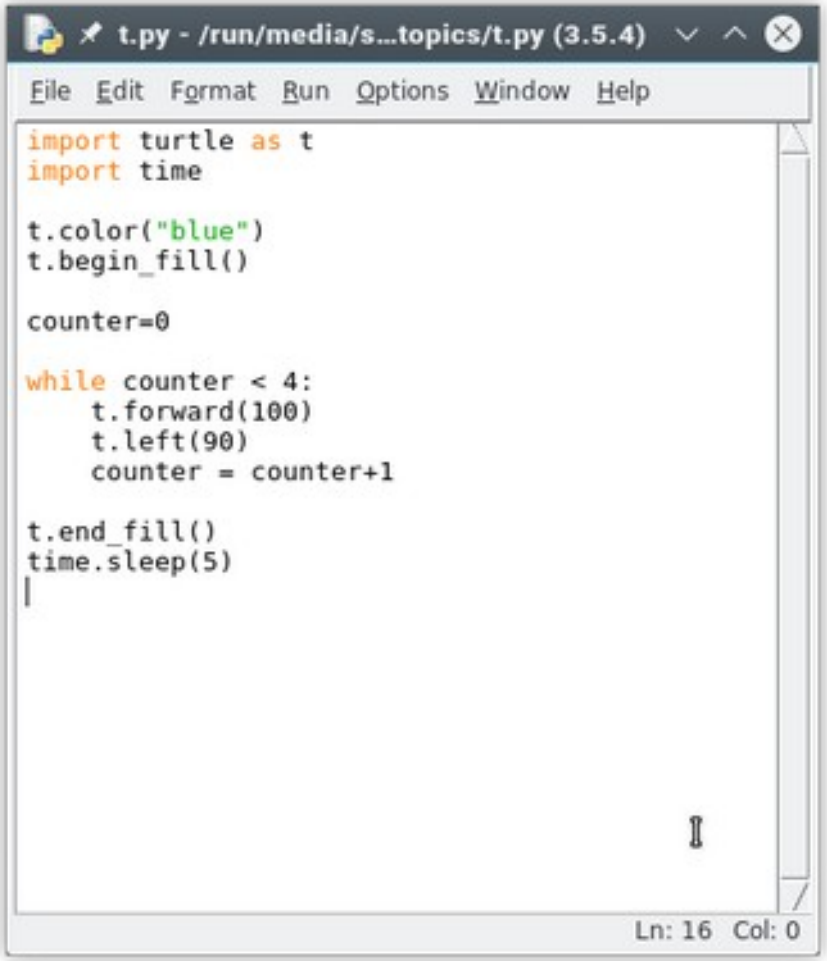
  IDLE (Integrated Development and Learning Environment) is a default editor that comes with Python. It is one of the best Python IDE software which helps a beginner to learn Python easily. IDLE software package is optional for many Linux distributions. The tool can be used on Windows, macOS, and Unix.

Python Editors

- 5) Sublime Text 3

  Sublime Text 3 is a code editor which supports many languages including Python. It is one of the best Python editor that has basic built-in support for Python. Customization of Sublime Text 3 is available for creating create a full-fledged Python programming environment. The editor supports OS X, Windows, and Linux operating systems.


Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('Hello, World!')
Hello, World!
>>>
Ln: 5 Col: 4


t.py - /run/media/s...topics/t.py (3.5.4)
File Edit Format Run Options Window Help
import turtle as t
import time
t.color("blue")
t.begin_fill()
counter=0
while counter < 4:
t.forward(100)
t.left(90)
counter = counter+1
t.end_fill()
time.sleep(5)
Ln: 16 Col: 0

Summary

- Python is an interpreted, object-oriented, high-level programming language with dynamic semantics.
- Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.
- Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.
- Python supports modules and packages, which encourages program modularity and code reuse.
- The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.