HOPE foundation

PYTHON PROGRAMMING

Contents

I. Computer

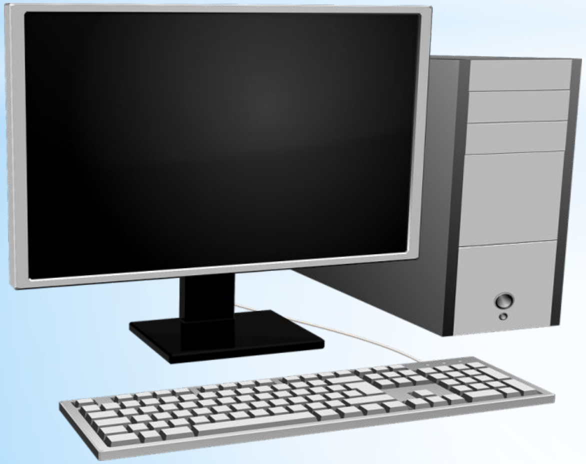
II. Computer Languages

III. Programming

IV. Python Language

Computer

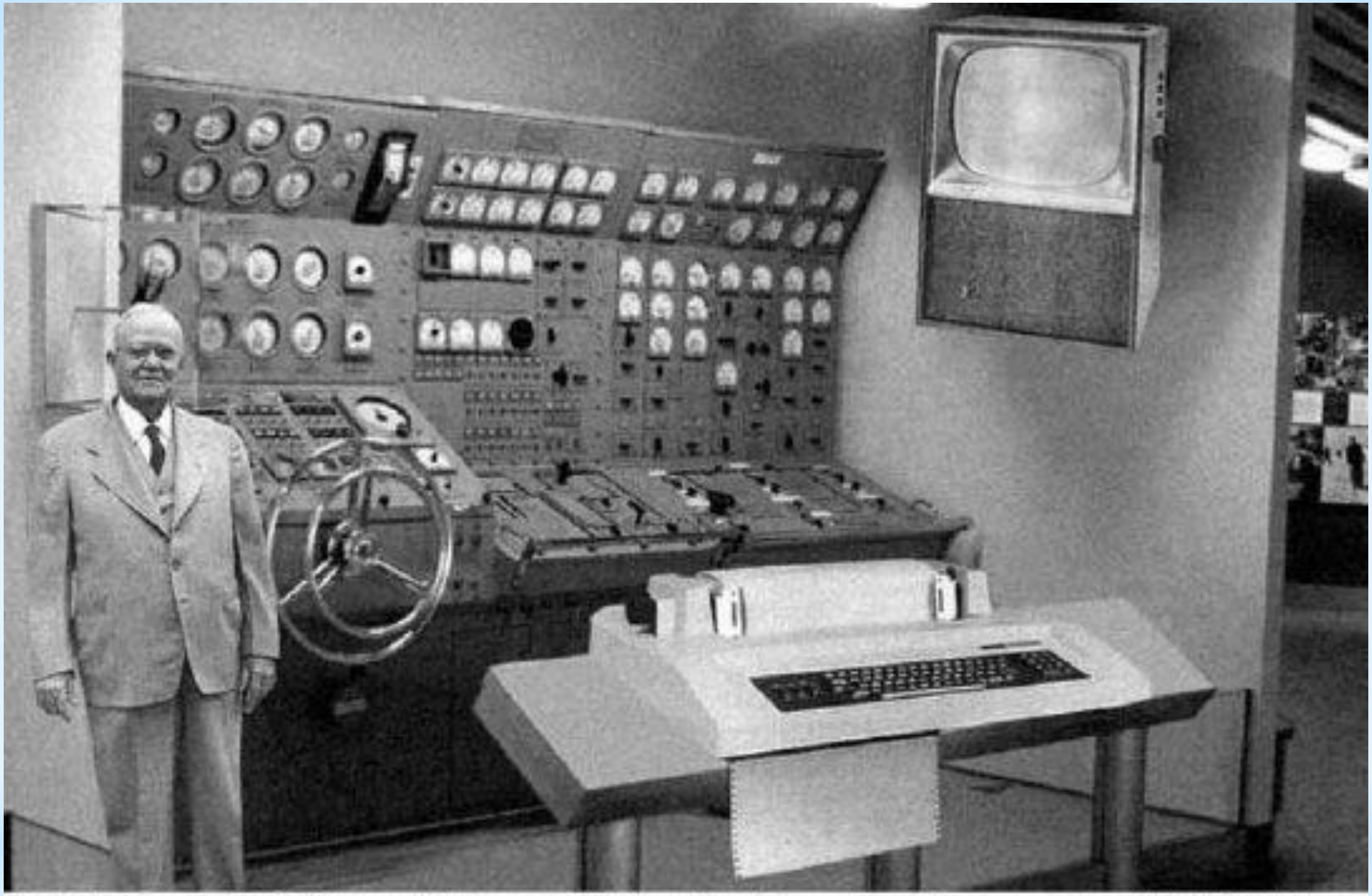
A computer is a very powerful electronic device which accepts instructions from input devices, process them and finally outputs the results using output devices.



Desktop

Laptop

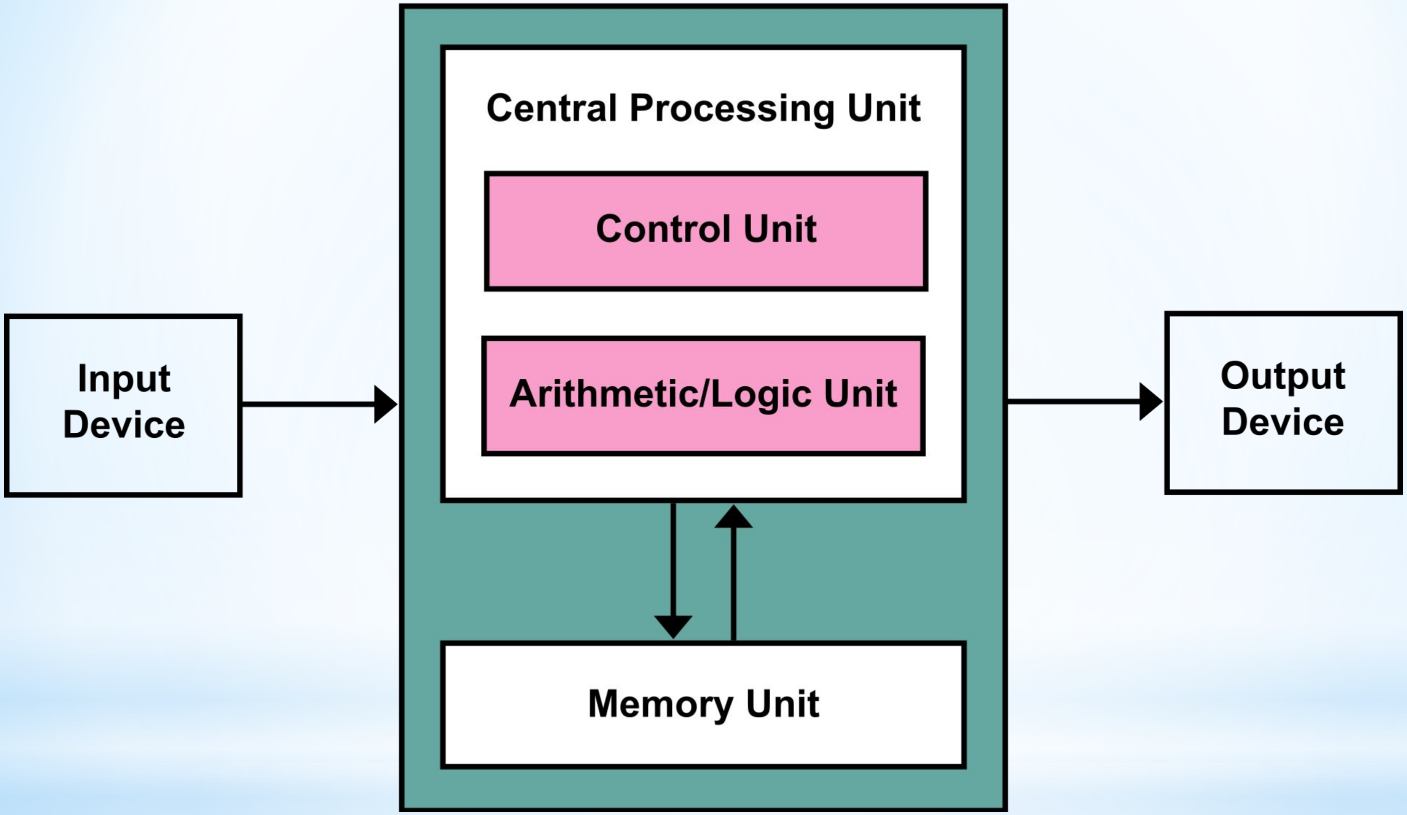
Handheld Computer



Scientists from the RAND Corporation have created this model to illustrate how a "home computer" could look like in the year 2004. However the needed technology will not be economically feasible for the average home. Also the scientists readily admit that the computer will require not yet invented technology to actually work, but 50 years from now scientific progress is expected to solve these problems. With teletype interface and the Fortran language, the computer will be easy to use.

First model of Home Computer

BLOCK DIAGRAM OF A COMPUTER



Input Devices

An input device accepts instructions from the user and transfers to processing devices.

Keyboard and Mouse

Joy Stick

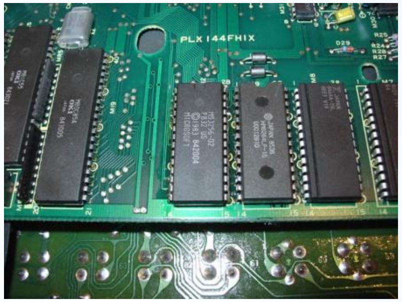
Track Ball

Light Pen

Processing Devices

Processing devices fetch instructions from input devices and process them accordingly, the results are then transferred to output devices.

RAM



ROM

Processor Chip

Hard Disk

CMOS

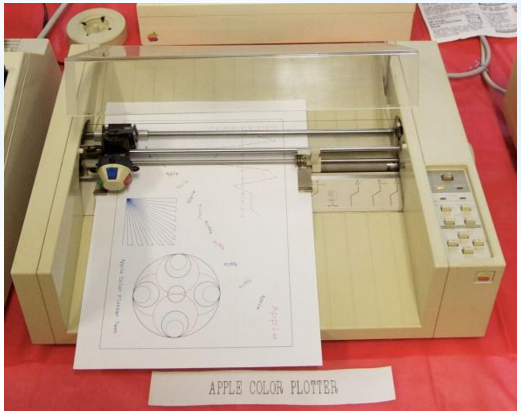
Output Devices

Output devices receives processed data and displays the result through various devices like Printer, Monitor, Plotter etc.,.

Printer

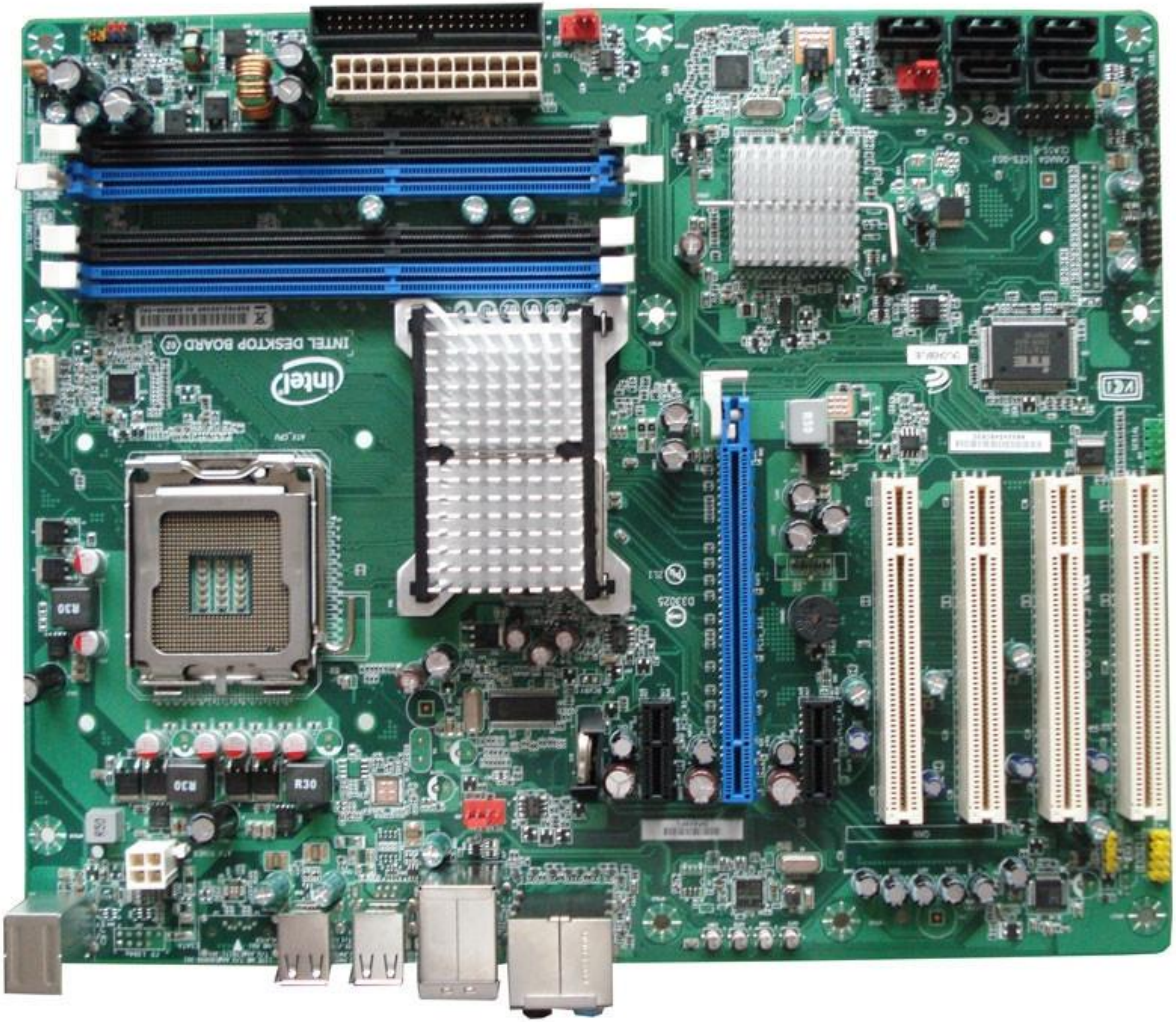
Monitor Screen

Multifunctional Device



Plotter

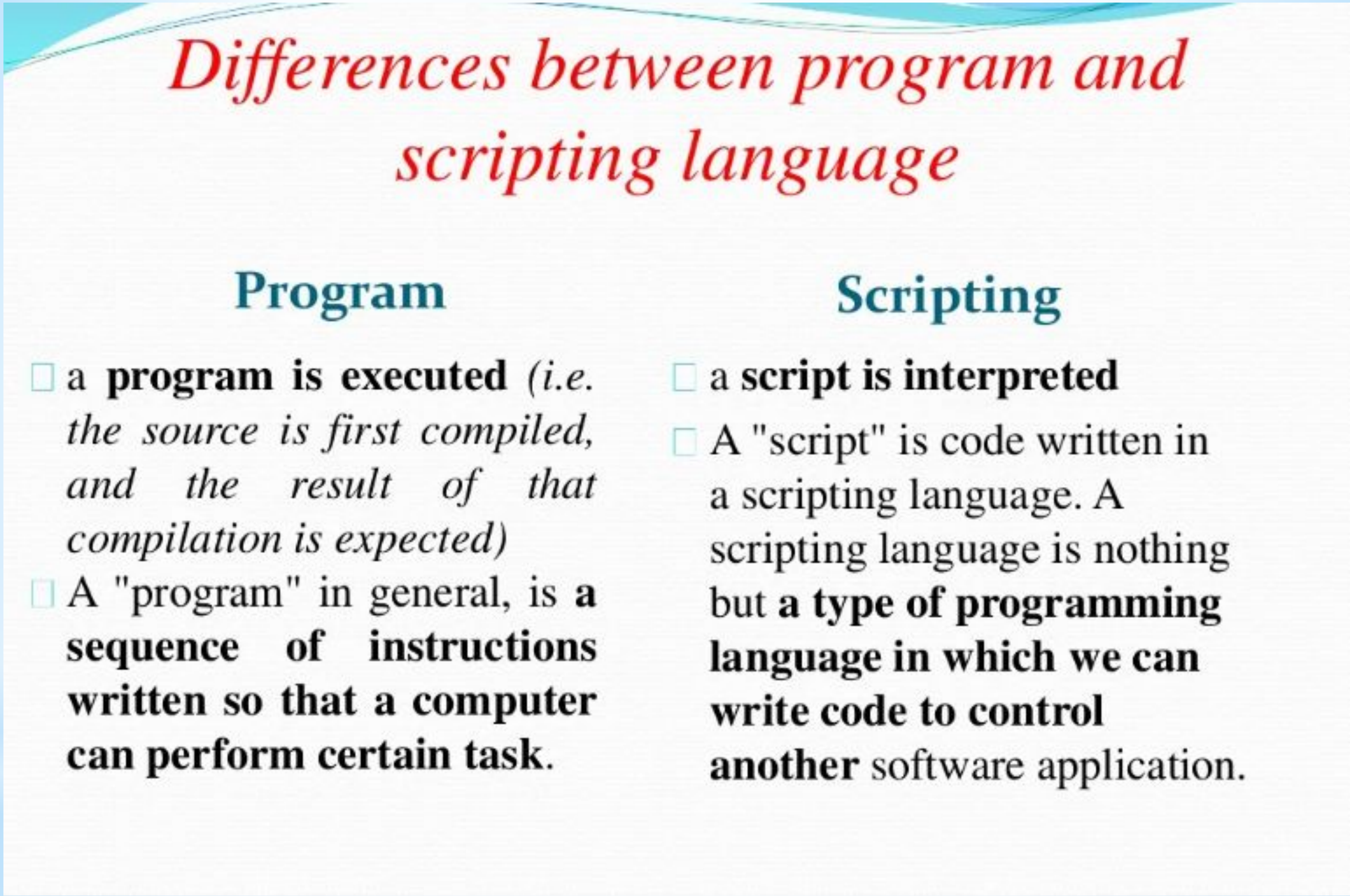
MOTHER BOARD



Computer Languages

Computer programming languages allow us to give instructions to a computer in a language the computer understands. there are many computer programming languages that programmers can use to communicate with a computer.

- Machine Language



Differences between program and scripting language

Program

□ a **program is executed** *(i.e. the source is first compiled, and the result of that compilation is expected)*

□ A "program" in general, is **a sequence of instructions written so that a computer can perform certain task**.

Scripting

□ a **script is interpreted**

□ A "script" is code written in a scripting language. A scripting language is nothing but **a type of programming language in which we can write code to control another** software application.

PROGRAMMING

Program is a set of instructions that are written in a language that the computer can understand.

- C, C++
- Java
- Python
- PHP
- PEARL
- Ruby

COMPILER

A **compiler** is a computer program that translates computer code written in one programming language (the source language) into another language (the target language).

INTERPRETER

An **interpreter** is a computer program that directly executes instructions written in a programming or scripting language, without requiring them previously to have been compiled into a machine language program.

PYTHON LANGUAGE

Python is a high-level, interpreted, interactive and object-oriented scripting language. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Guido van Rossum is the creator of the Python programming language. He developed It in mid 1985-1990.

Brief History of Python

- Invented in the Netherlands, early 90s by Guido van Rossum
- Named after Monty Python
- Open sourced from the beginning
- Considered a scripting language, but it is also, scalable, object oriented and functional from the beginning
- Used by Google from the beginning
- Increasingly popular

Applications of Python

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.

Applications of Python

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.

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.

Applications of Python

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.

PYTHON PROGRAMMING (Basics)

TOPIC: 1.2

Python & PyCharm Installation

I. Real World Applications of Python

II. Python Installation

III. Working on Python IDLE

IV. Run Python Script using command prompt

V. PyCharm Installation

1. Real World Applications of Python

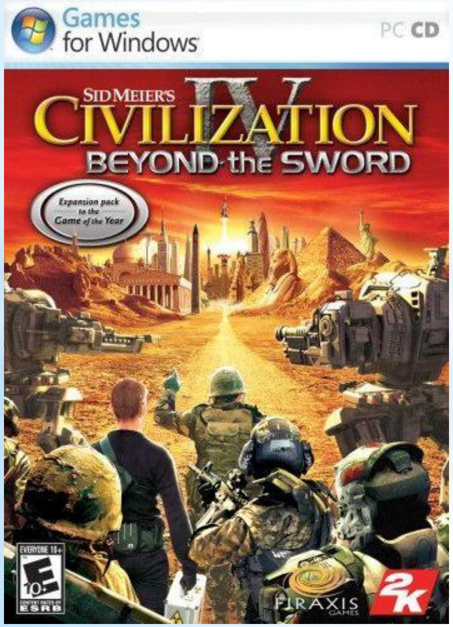
1.1 Web Development:

Python can be used to make web-applications at a rapid rate. Python frameworks like Django, Flask, Pyramid used to develop these applications.

1. Real World Applications of Python

1.2 Game Development:

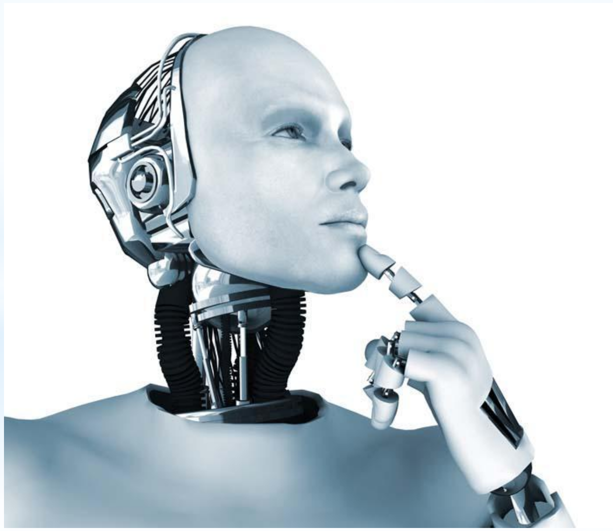
Python can be used to develop interactive games. There are libraries such as PySoy which is a 3D game engine supporting Python 3, PyGame which provides functionality and a library for game development.

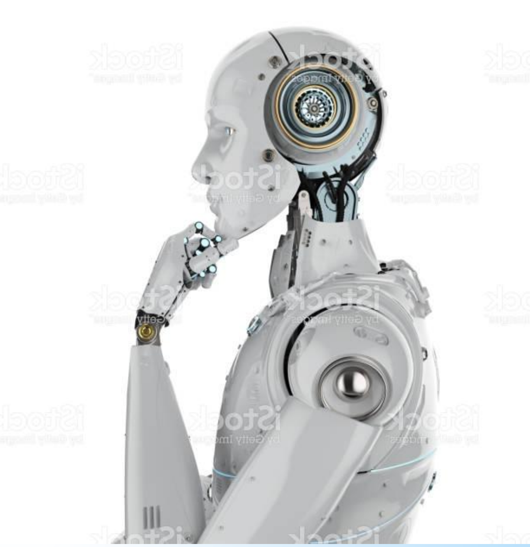


1. Real World Applications of Python

1.3 Machine Learning & Artificial Intelligence:

Machine Learning and Artificial Intelligence are the talks of the town as they yield most promising careers for the future. Libraries such as Pandas, Scikit-learn, Numpy support machine learning and artificial intelligence applications.





1. Real World Applications of Python

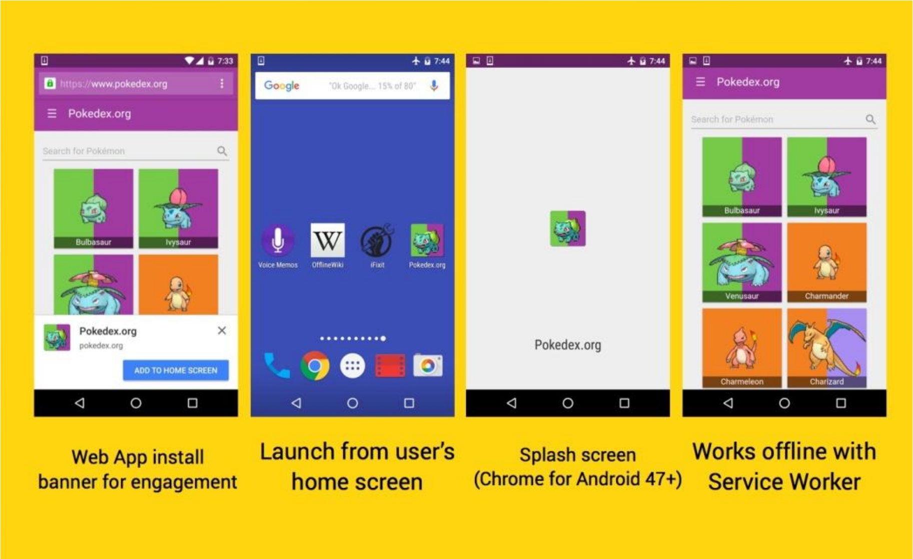
1.4 Data Science & Data Visualization :

Data Science is extracting information and analysing it to take appropriate decisions in organizations. Libraries such as Pandas, NumPy helps in extracting information. You can even visualize the data libraries such as Matplotlib, Seaborn, which are helpful in **plotting graphs** and much more. .

1. Real World Applications of Python

1.5 Desktop GUI :

Python can be used to program **desktop applications.** It provides the Tkinter library that can be used to develop user interfaces. There are some other useful toolkits such as the wxWidgets, Kivy, PYQT that can be used to create applications on several platforms.



Web App install banner for engagement

Launch from user's home screen

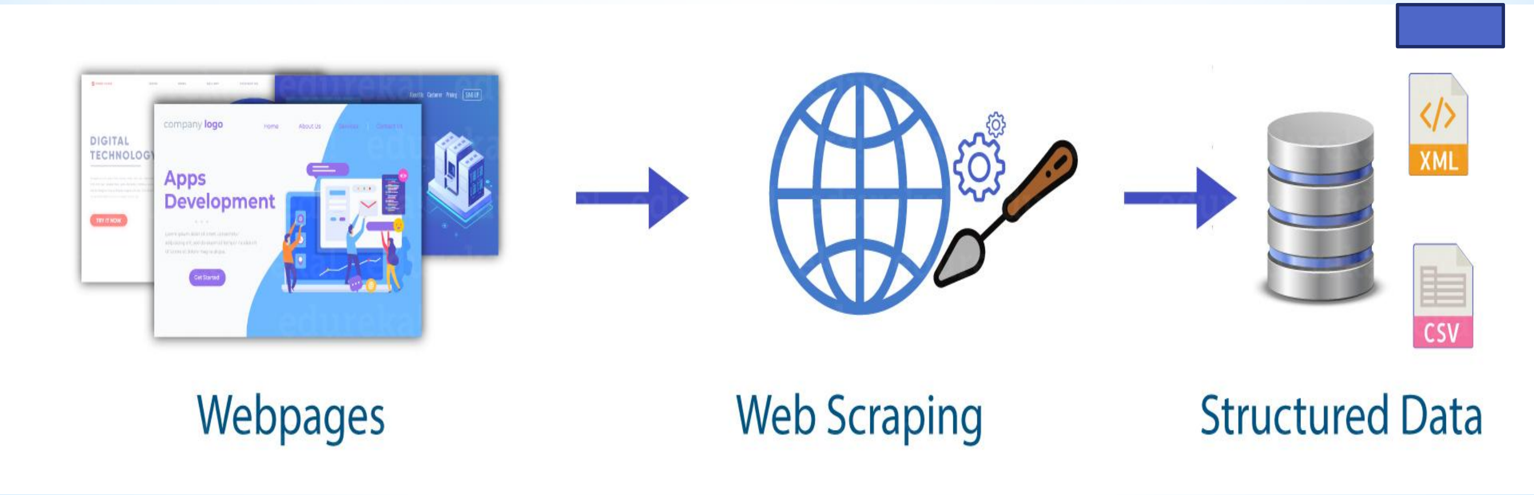
Splash screen (Chrome for Android 47+)

Works offline with Service Worker

1. Real World Applications of Python

1. 6 Web Scarping Applications:

Python can be used to pull a large amount of data from websites which can then be helpful in various real-world processes such as price comparison, job listings, research and development and much more. Python has a library called **BeautifulSoup** which can be used to pull such data.



1. Real World Applications of Python

1.7 Business Applications:

Business Applications are different than our normal applications covering domains such as e-commerce, ERP and many more. Platforms such as **Tryton** can be used to develop such business applications.

1. Real World Applications of Python

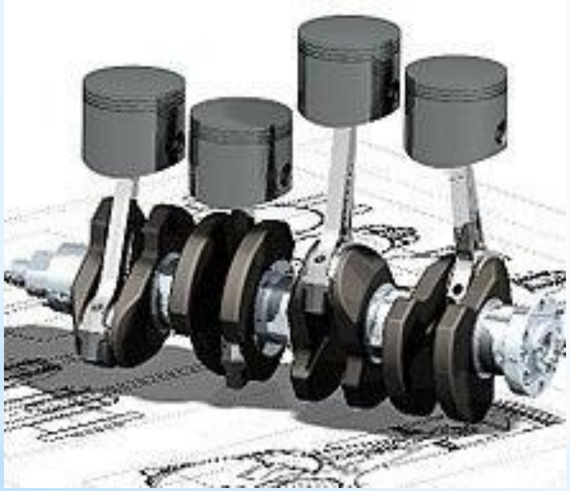
1. 8 Audio and Video Applications:

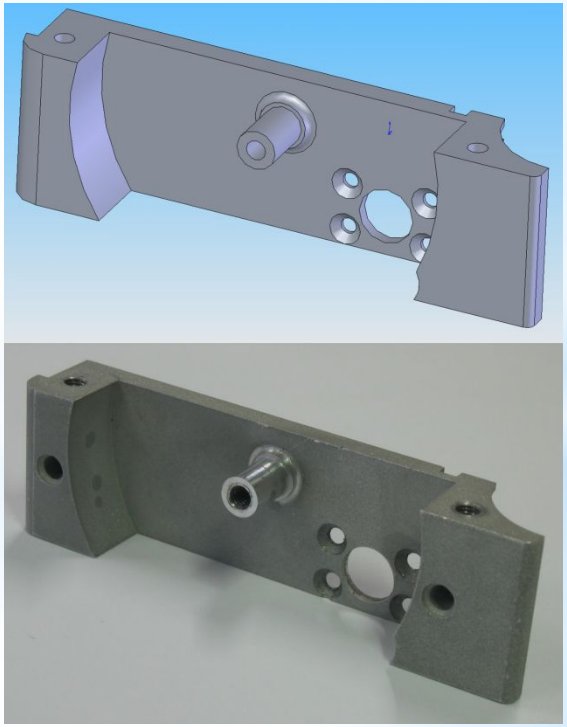
Python can be used to develop applications that can multi-task and also output media. Video and audio applications such as **TimPlayer**, **Cplay** have been developed using Python libraries and they provide better stability and performance compared to other media players.

1. Real World Applications of Python

1. 9 Computer Aided Design Applications:

Fandango is a Python application which makes complicated designing look easy.





1. Real World Applications of Python

1.10 Embedded Applications:

Python can be used for creating Embedded C Software for embedded applications. This helps to perform higher-level applications on smaller devices.

I. Real World Applications of Python ✓

II. Python Installation

III. Working on Python IDLE

IV. Run Python Script using command prompt

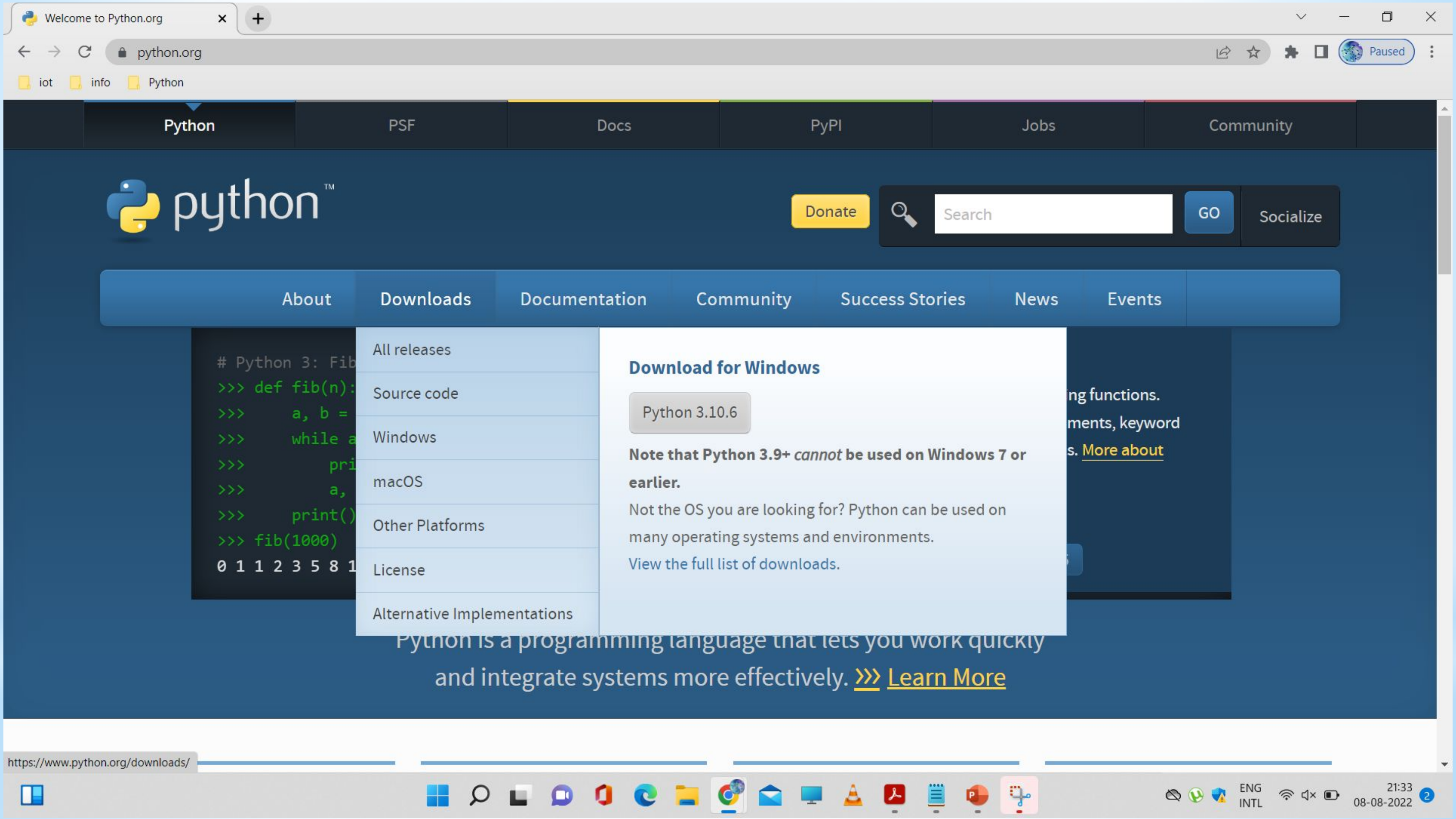
V. PyCharm Installation

I. Python Installation ✓

II. Working on Python IDLE

III. Run Python Script using command prompt

IV. PyCharm Installation



```
# Python 3: Fib
>>> def fib(n):
>>>     a, b =
>>>     while a
>>>         pri
>>>         a,
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 1
```

Download for Windows

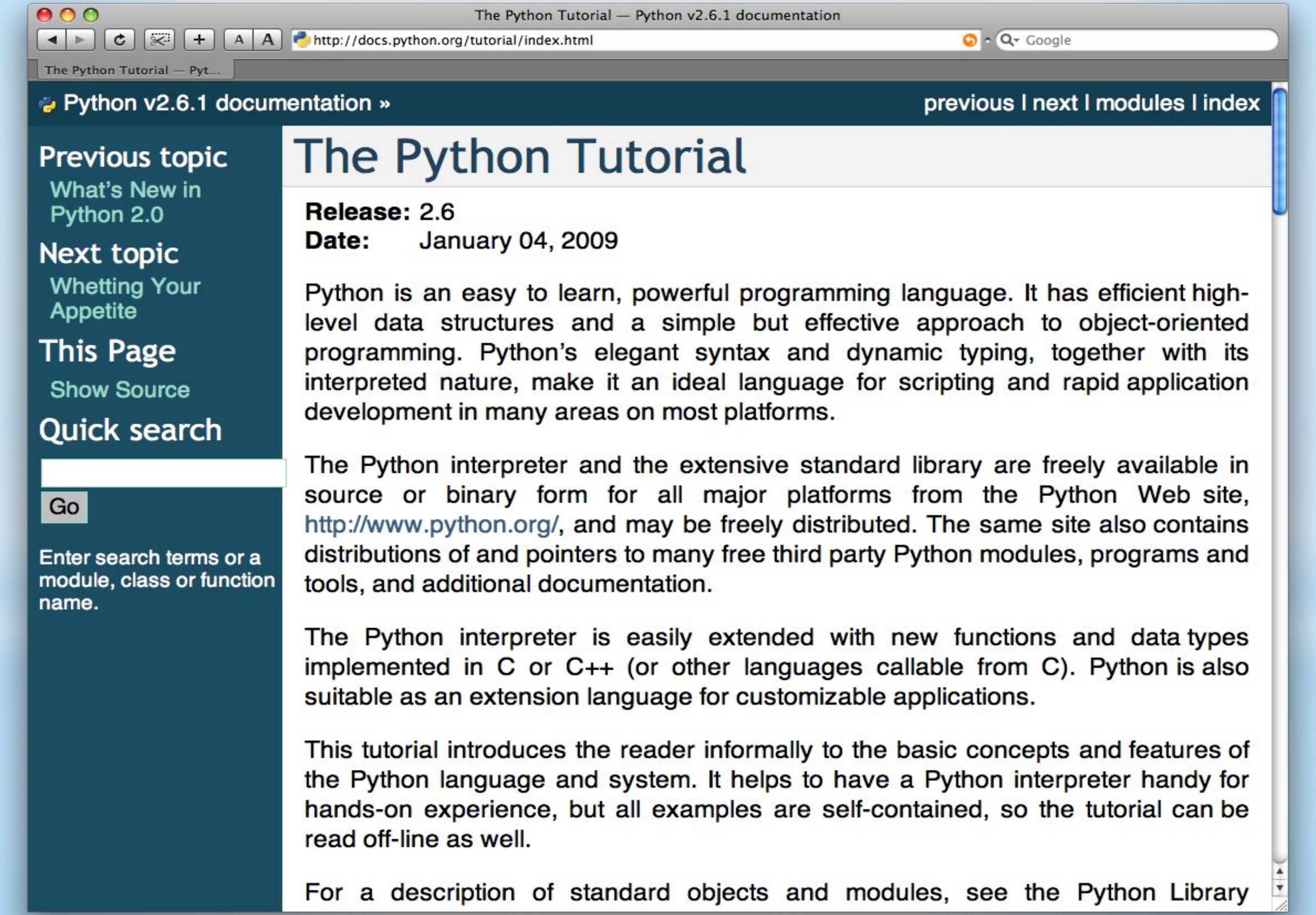
Note that Python 3.9+ *cannot* be used on Windows 7 or earlier.

Not the OS you are looking for? Python can be used on many operating systems and environments.

View the full list of downloads.

ng functions.
ments, keyword
s. More about

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More


The Python Tutorial — Python v2.6.1 documentation
http://docs.python.org/tutorial/index.html
Google
The Python Tutorial — Pyt...
Python v2.6.1 documentation »
previous | next | modules | index
Previous topic
What's New in Python 2.0
Next topic
Whetting Your Appetite
This Page
Show Source
Quick search
Go
Enter search terms or a module, class or function name.

The Python Tutorial

Release: 2.6
Date: January 04, 2009

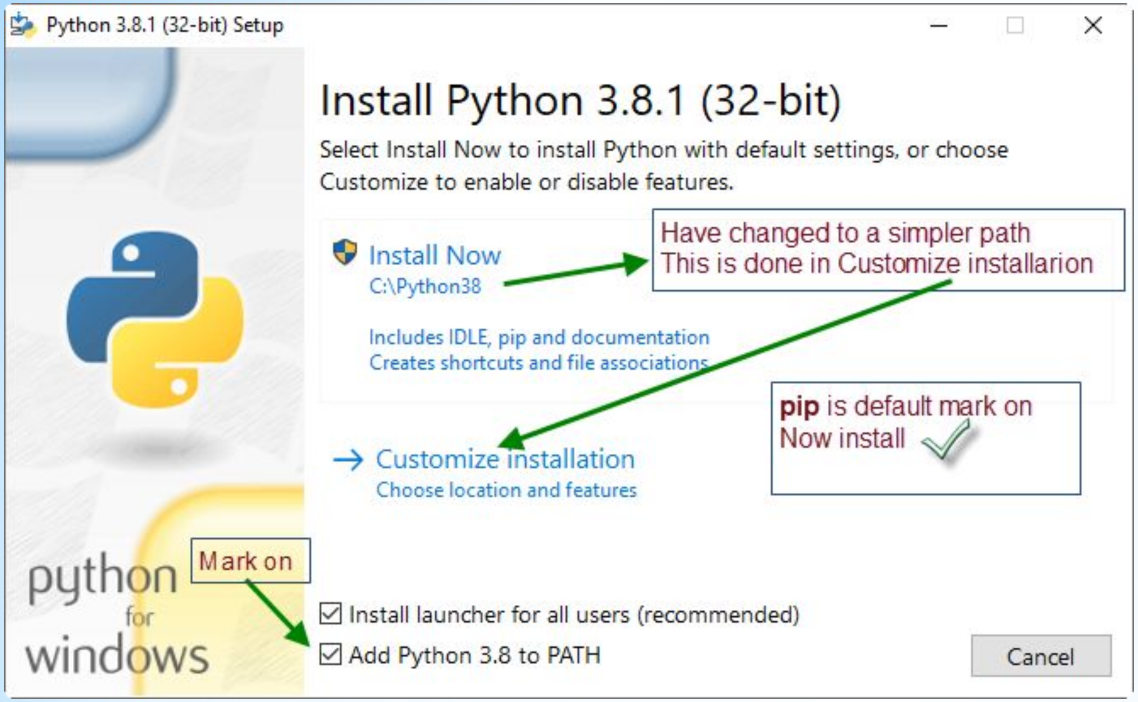
Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python Web site, http://www.python.org/, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation.

The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

This tutorial introduces the reader informally to the basic concepts and features of the Python language and system. It helps to have a Python interpreter handy for hands-on experience, but all examples are self-contained, so the tutorial can be read off-line as well.

For a description of standard objects and modules, see the Python Library



Python 3.8.1 (32-bit) Setup

Install Python 3.8.1 (32-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Python38

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

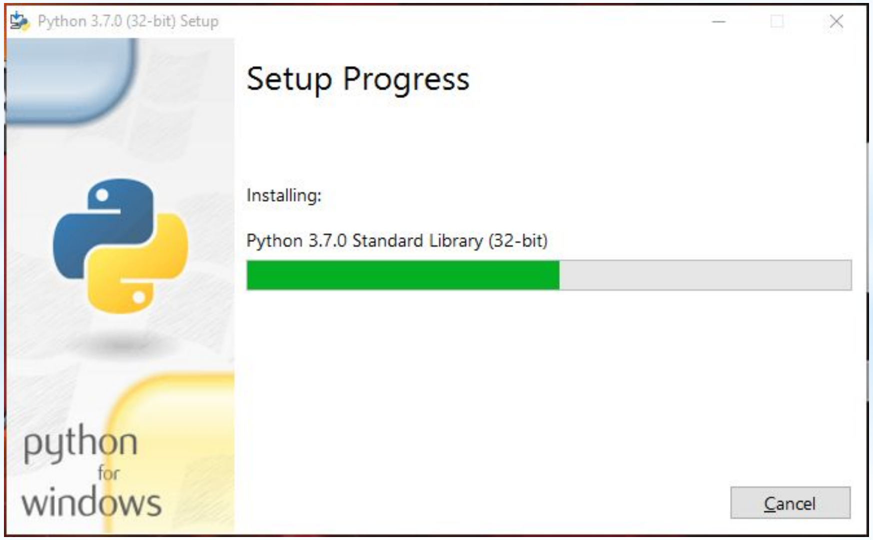
☑ Install launcher for all users (recommended)

☑ Add Python 3.8 to PATH

Have changed to a simpler path
This is done in Customize installarion

pip is default mark on
Now install ✓

Mark on



Python 3.7.0 (32-bit) Setup

Setup Progress

Installing:

Python 3.7.0 Standard Library (32-bit)

I. Python Installation ✓

II. Working on Python IDLE ✓

III. Run Python Script using command prompt

IV. PyCharm Installation

```
x = 34 - 23              # A comment.
y = “Hello”              # Another one.
z = 3.45
if z == 3.45 or y == “Hello”:
    x = x + 1
    y = y + “ World”   # String concat.
print x
print y
```

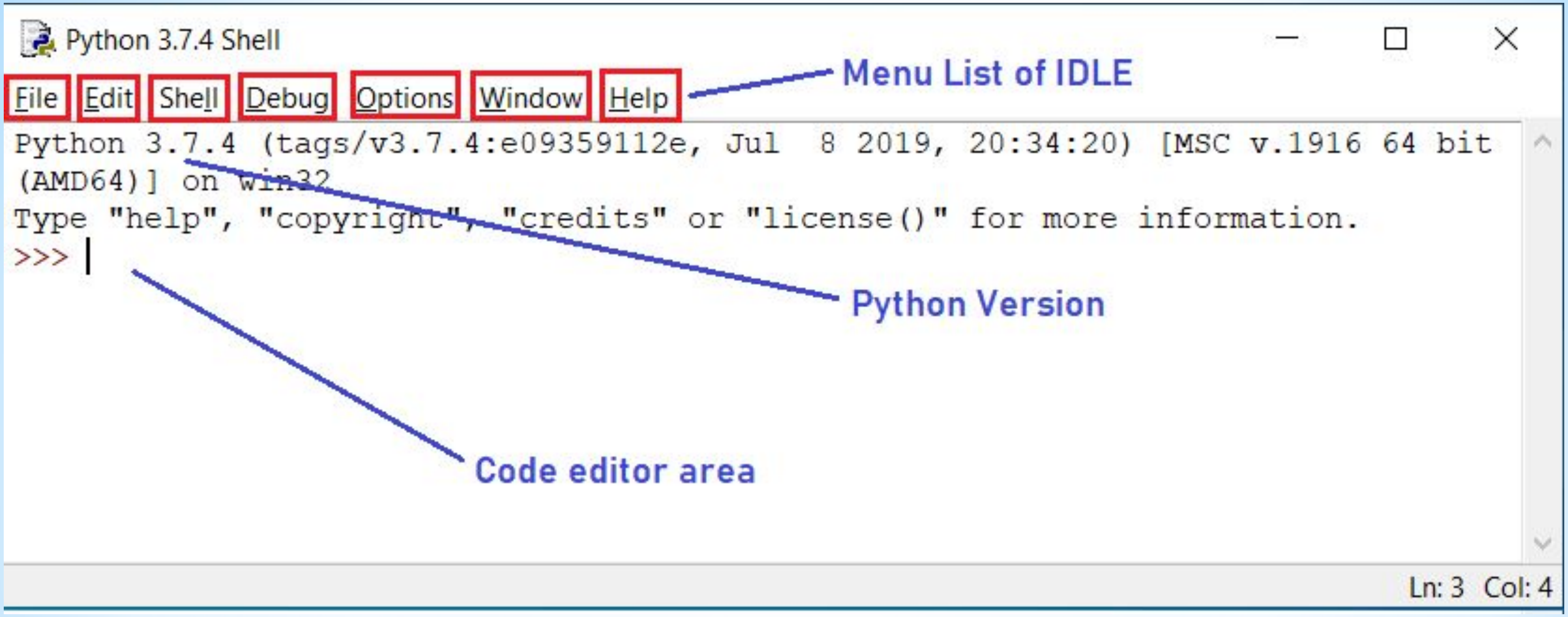
*A Code Sample (in IDLE)

I. **Python Installation** ✓

II. **Working on Python IDLE** ✓

III. **Run Python Script using command prompt** ✓

IV. **PyCharm Installation**

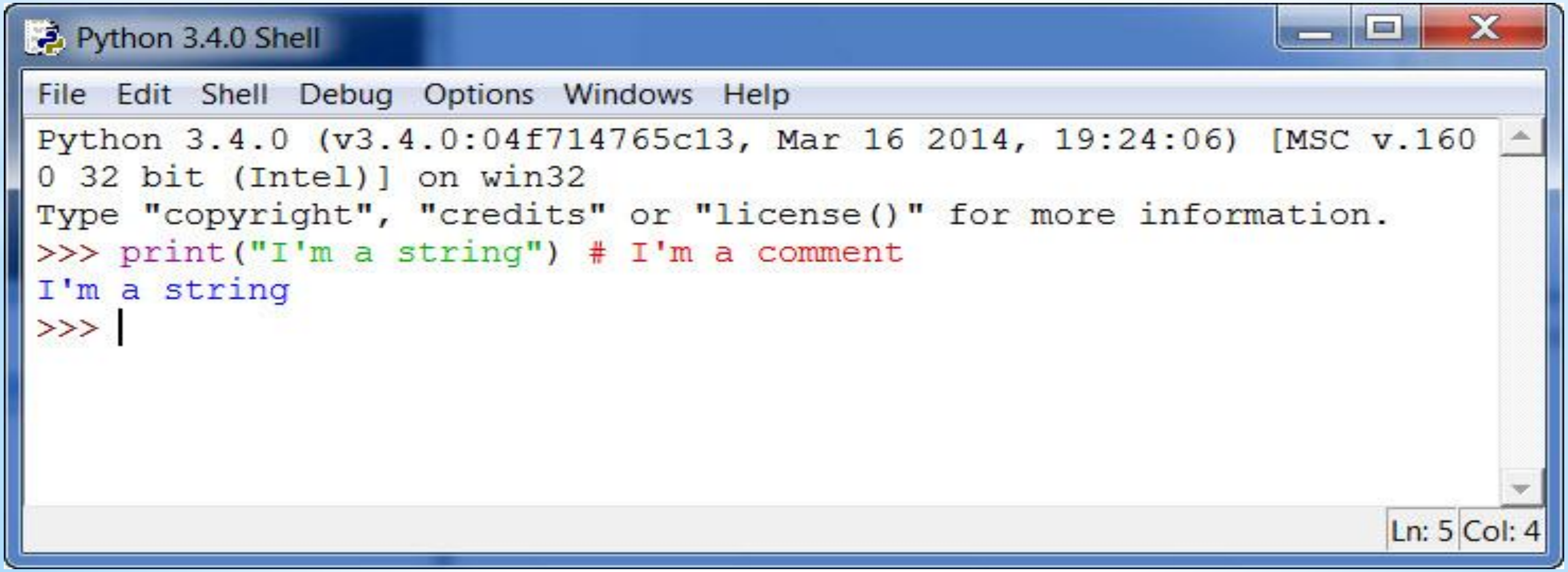


Menu List of IDLE

Python Version

Code editor area

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
```



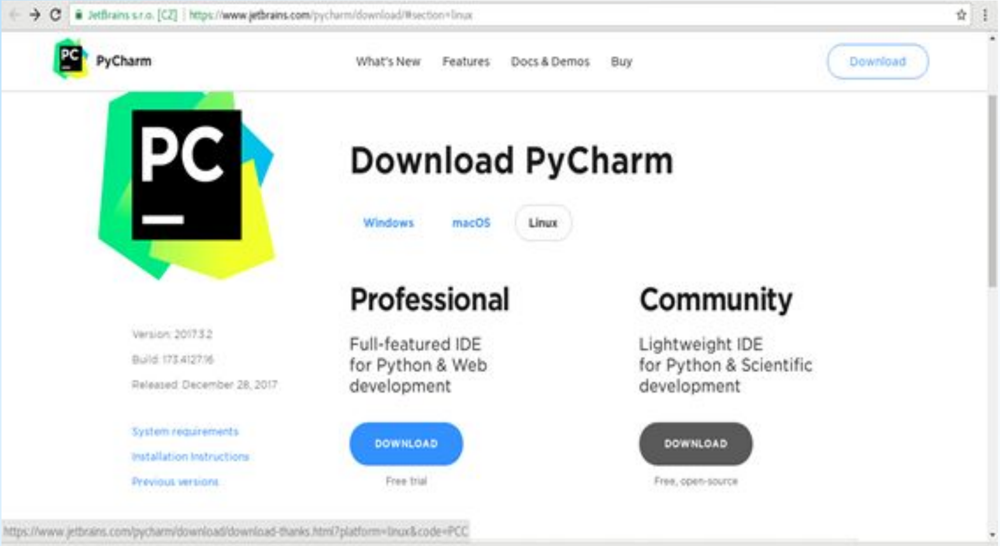
```
Python 3.4.0 (v3.4.0:04f714765c13, Mar 16 2014, 19:24:06) [MSC v.160
0 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("I'm a string") # I'm a comment
I'm a string
>>> 
```

I. Python Installation ✓

II. Working on Python IDLE ✓

III. Run Python Script using command prompt ✓

IV. PyCharm Installation ✓


JetBrains s.r.o. [CZ] | https://www.jetbrains.com/pycharm/download/#section=linux
PyCharm
What's New
Features
Docs & Demos
Buy
Download
Windows
macOS
Linux
DOWNLOAD
DOWNLOAD
https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=linux&code=PCC

Download PyCharm

Version: 2017.3.2
Build: 173.4127.16
Released: December 28, 2017

System requirements
Installation Instructions
Previous versions

Professional

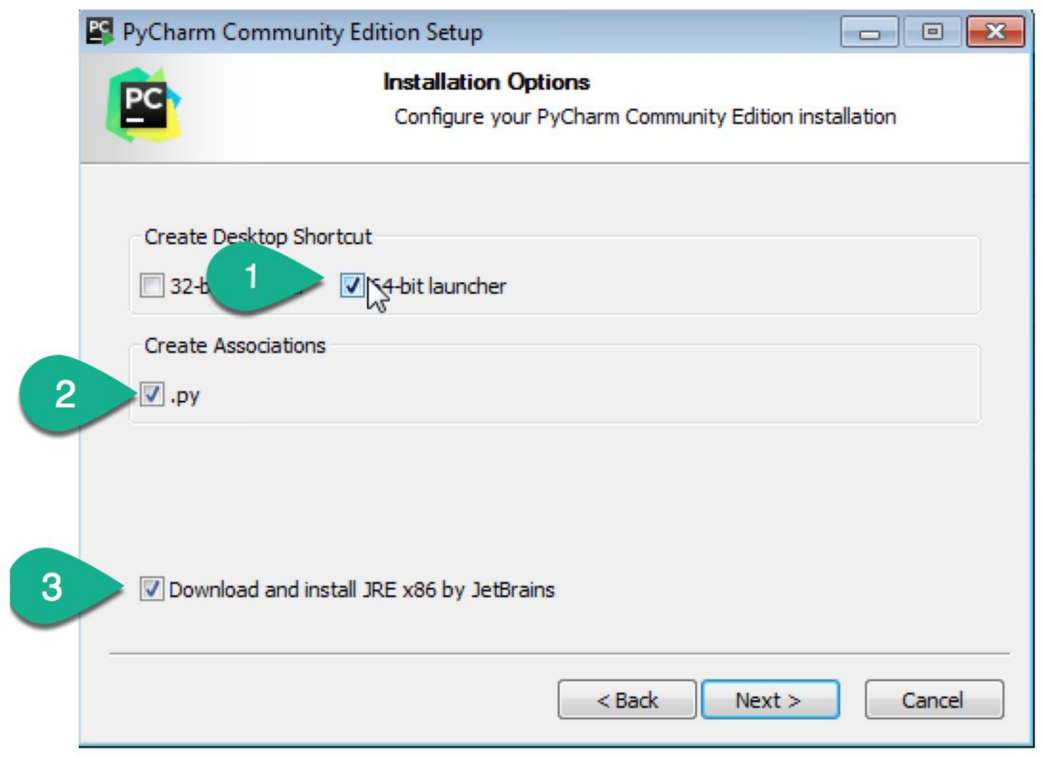
Full-featured IDE for Python & Web development

Free trial

Community

Lightweight IDE for Python & Scientific development

Free, open-source



Installation Options

Configure your PyCharm Community Edition installation

Create Desktop Shortcut

1. □ 32-b ☑ 64-bit launcher

Create Associations

2. ☑ .py

3. ☑ Download and install JRE x86 by JetBrains