Chapter One
Introduction

By Kibret Zewdu

Department of Computer Science
College o Informatics

University of Gondar

Basic Programming Concepts

- introduces basic concepts such as:
  - instructions,
  - data types,
  - variables, and
  - some other related terminology.

What is a program?

- Computers only know what you tell them.
- The way to tell them to do something is by a **program**.
- A program is a set of ordered instructions designed to command the computer to do something.

What is Python

- **Python** is a general purpose, dynamic, high-level and interpreted programming language.
- It supports **Object Oriented programming** approach to develop applications.
- Python **is *easy to learn*** yet powerful and versatile scripting language, which makes it attractive for Application Development.
- Python's **syntax** and ***dynamic typing*** with its interpreted nature make it an ideal language for scripting and rapid application development.
- Python supports *multiple programming pattern*, including object-oriented, imperative, and functional or procedural programming styles.

Python 2 and Python 3 are different

Python 2

- uses **print** as a statement and used as print "something" to print some string on the console.
- uses the function **raw_input()** to accept the user's input.
  - It returns the string representing the value, which is typed by the user.
  - To convert it into the integer, use the **int()** function in Python.
- The implicit string type is ASCII

Python 3

- uses **print** as a function and used as print("something") to print something on the console.
- uses **input()** function which automatically interpreted the type of input entered by the user.
  - However, we can cast this value to any type by using primitive functions (int(), str(), etc.
- The implicit string type is Unicode

Python History

- Python was invented by **Guido van Rossum** in 1991 at CI in Netherland.
  - The idea of Python programming language has taken from the **ABC** programming language or we can say that ABC is a predecessor of Python language.
- Guido van Rossum was a fan of the popular BBC comedy show of that time, **"Monty Python's Flying Circus"**.
- Python has the vast community across the world and releases its version within the short period.

Python Features

- **Readability**: Python is designed to be a highly readable language
  The use of English keywords, and the use of spaces to limit code blocks and its internal logic (indentation), contribute to this end.

```
print("Hello world!")
```

- **Built-in features**:
  It has a rich and versatile standard library that is immediately available.
  With few lines, you can read and write XML and JSON files, parse and generate email messages, extract files from a zip archive, open a URL as if were a file, and many other possibilities that in other languages, it would require a third-party library.
- **Availability of third-party modules** for a broad spectrum of activities.
  Data Visualization and plotting, PDF generation, bioinformatics analysis, image processing, machine learning, game development, interface with popular databases, and application software are only handful examples of modules that can be installed to extend python functionalities.

Python Features

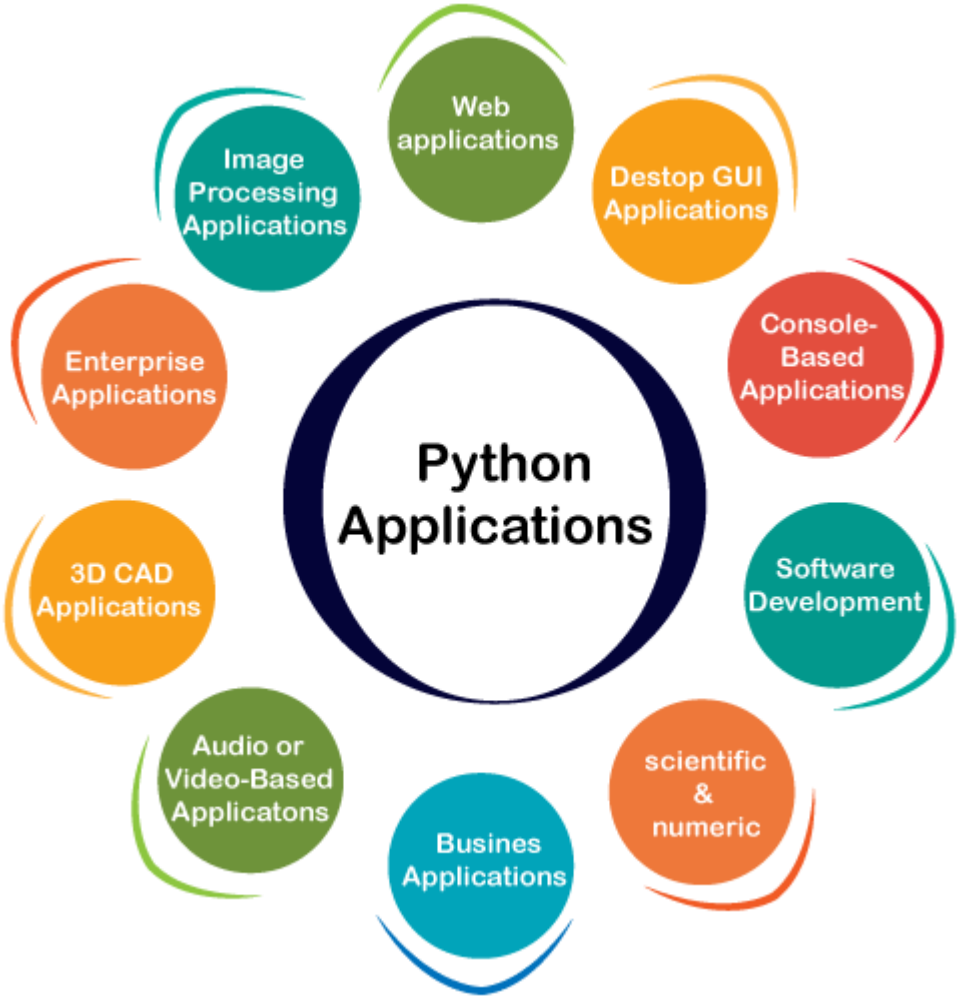
- **High-level built-in data structures**: Dictionaries, sets, lists, tuples, and others. These are very useful to model real-world data.
  Third-party modules such as NumPy and SciPy can also extend the structures to kd-trees, n-dimensional arrays, matrix operations, time-series, image objects, and more.
- **Multiparadigm**: Python can be used as a “classic” procedural language or as “modern” object-oriented programming (OOP) language

Python Features

- **Open source**: Python has a liberal open source license that makes it freely usable and distributable, even for commercial use.
- **Cross platform**: A program made in Python can be run under any computer that has a Python interpreter.
- **Thriving community**: Python is nowadays the programming language to use for scientists and researchers

Python Applications

- **Python** is known for its general-purpose nature that makes it applicable in almost every domain of software development.
- **Python** makes its presence in every emerging field.
- It is the fastest-growing programming language and can develop any application.



Python Applications

- Python has a wide range of applications.
  - From cell phones to web servers, there are thousands of Python applications in the most diverse fields.
  - There is Python code powering Wikipedia robots, helping design next generation special effects at Industrial Light & Magic, and
  - it is the scripting language of the **OpenOffice** suite embedded in D-link modems and routers.
- Python **desktop applications** run with a native look and feel on multiple platforms.
  - include the **BitTorrent** p2p client/server, **Calibre**, an **Ebook** manager, Sage Math (a mathematics software system), the Dropbox client, and more.

Python Applications

- **For building web applications**
  - Python can be found in high traffic sites like Reddit, NationalGeographic, Instagram, and NASA.
  - There are specialized software for building web sites (called webframeworks) in Python like **Django**, **Web2Py**, **Pyramid**, **Flask**, and **Bottle**.
- From **system administration to data analysis**
  - Python provides a broad range of tools such as generic operating system services, file and directory access, data compression and archiving, Interprocess communication and networking, internet, String services

Python Applications

- **Python** is gaining momentum as the default computer language for the scientific community
- There are several libraries oriented toward scientific users, such as **SciPy** and **Anaconda**.
- Both distributions integrate modules for linear algebra, signal processing, optimization, statistics, genetic algorithms, interpolation, ODE solvers, special functions, etc.
- **Python** has support for **parallel programming** with **pyMPI** and 2D/3D scientific data plotting.
- **Python** is known to be used in wide and diverse fields like engineering, electronics, astronomy, biology, paleomagnetism, geography, and many more.

Installing Python

- **Python** is pre-installed in macOS and most Linux distributions.
- In **Windows**, download the Windows x86-64 web-based installer from the Python download page (https://www.python.org/downloads/windows/).
- **Installation** is pretty straightforward
  - Double-click the installer file and run the Python Install Wizard.
  - Accept the default settings and you will have Python installed in a few minutes without hassle.

Installing Anaconda

- **Anaconda** offers the easiest way to perform Python/R data science and machine learning on a single machine.
- Has thousands of open-source packages and libraries.
- Download the latest anaconda installer for windows from https://repo.anaconda.com/archive/Anaconda3-2022.10-Windows-x86_64.exe

Python Use

- Two ways to use Python:
  - **Interactive** mode
  - **Batch** mode
- **Interactive** mode allows the programmer to get an immediate answer to each instruction.
- In **batch** mode, instructions are stored in one or more files and then executed.
- **Interactive** mode is used mostly for small tests while most programs are run in batch mode.
- **Interactive** mode can be invoked by executing python or within some Python editors like Spyder, PyCharm, IDLE and others

Interactive Mode

To command python interpreter to print **Hello World!** string

```
>>> print('Hello World!')
Hello World!
```

- The three greater-than characters (>>>); this is the Python prompt of the interactive mode.
- It is already there, you don't need to type it

Basic Input and Output
Output:

- From Python 3, **print** is a function.
- A function is a reusable code that can perform a specific task.
- Each function may receive one or more values called parameters.
- In the case of `print("Hello World!")`, the name of the function is print and the parameter is the string `"Hello World!"`.

Basic Input and Output
Output:

- The print function can receive several elements:
```
>>> print('Hello', 'World!')
Hello World!
```
- By default it prints all string separated with a whitespace, but you can change the separator with a parameter named sep:
```
>>> print('Hello', 'World!', sep=';')
Hello;World!
```
- Redirect the output to a file:
```
>>> print("Hello","World!", sep=",", file=filehandle)
```
- To change the end on the output, use parameter end.
```
>>> print("Hello", "World!", sep=";", end='\n\n')
Hello;World!
```

Basic Input and Output
Input:

- To input data in a running program you can use input.
- The following command takes a string of data from the user and returns it to a variable called name.
- In the following code, after typing the string, the variable is entered and the content of the variable is displayed:

```
>>> name = input("Enter your name: ")
Enter your name: Aschalew
>>> name
’Aschalew’
```

More on Interactive Mode

- Interactive mode can be used as a calculator:

```
>>> 1 + 1
2
```

- When ‘+’ is used on strings, it returns a **concatenation**:

```
>>> ’1’+’1’
’11’
>>> "A string of " + ’characters’
’A string of characters’
```

More on Interactive Mode

- Different data types can’t be added:

```
>>> ’The answer is ’ + 42
Traceback (most recent call last):
   File "<stdin>", line 1, in <module>
TypeError: must be str, not int
```

- To convert this into a sum of strings, the number must be converted into a string; this is done with the str() function:

```
>>> ’The answer is ’ + str(42)
’The answer is 42’
```

- The same result can be archived with “String Formatting Operations”:

```
>>> ’The answer is {0}’.format(42)
’The answer is 42’
```

More on Interactive Mode

- Assign names to any Python element, and then refer to them later:

```
>>> number = 42
>>> 'The answer is {0}'.format(number)
'The answer is 42'
```

- **Names** should contain only letters, numbers, and underscores (_), but they can't start with numbers.
- In other programming languages **names** are called **variables**.

Arithmetic-Style Operators

Symbol	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
**	Exponentiation
%	Modulus(Remainder)

- The operator precedence is the same as used in math.

Exit from Python Shell

- To exit from the Python shell, in MacOS or Linux, use CRTL-D (that is, press Control and D simultaneously).
- In Windows, press CTRL-Z and Enter. Another alternative, that works in any operating system, is to use the exit() function:

```
$ python
Python 3.5.1 |Anaconda 2.4.1 (64-bit)| (Dec 7 2015,
11:16:01)
[GCC 4.4.7 20120313 (Red Hat 4.4.7-1)] on linux
Type "help", "copyright", "credits" or "license" for more
information.
>>> exit()
```

Batch Mode

- The code used in an interactive session can be accessed only when the session is active. Each time that an interactive session is closed, all typed code is gone.
- In order to have code persistence, programs are stored in text files.
- When a program is executed from such a text file, rather than line by line in an interactive interpreter, it is called batch mode.
  - These are regular text files usually with the ".py" extension.
- To run python text files,
  `python filename.py`