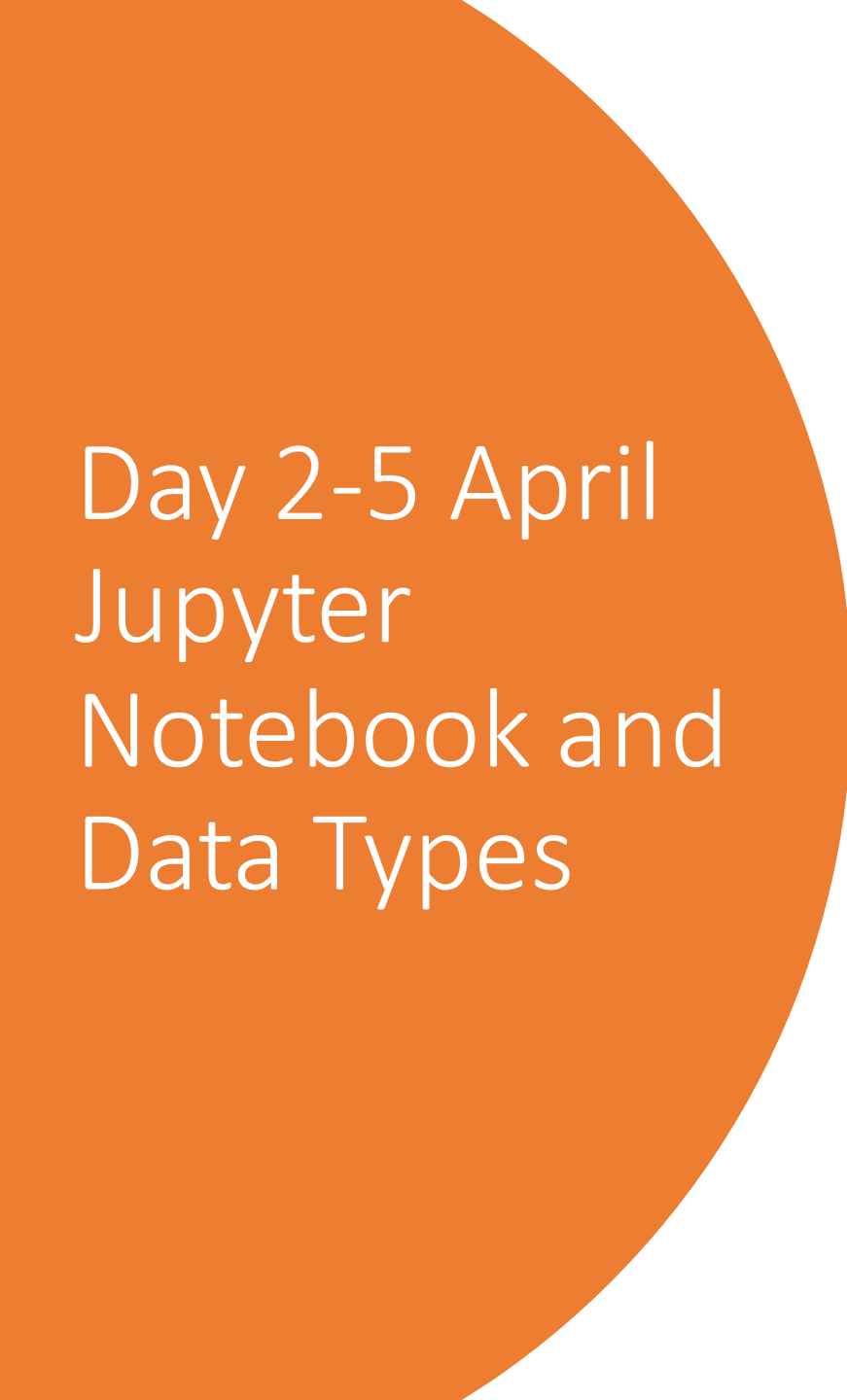


EdYoda Digital University


Python-21 March 2022

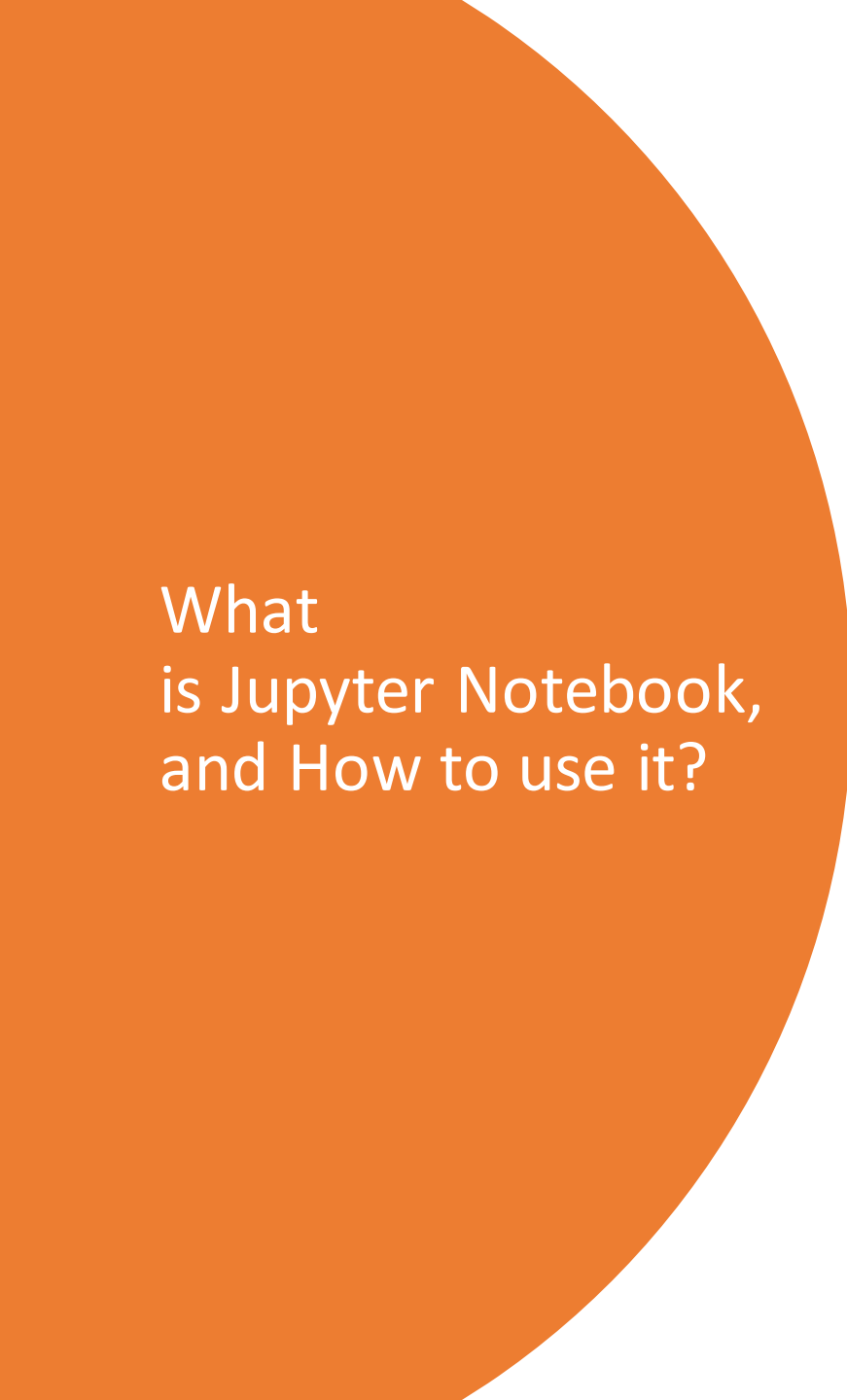
Batch-DS250322

Sagar Sarkar


A large orange circle is positioned on the left side of the slide, partially overlapping the text.

Day 2-5 April Jupyter Notebook and Data Types

1. What is Jupyter Notebook, and How to use it?
 2. Why do we need Data Types?
 3. Builtin Data Types.
 4. What are the different uses of Data Types.
 5. Range Function
 6. How to take Input
 7. Comments
- 
- A series of four yellow dashed line segments are arranged in a curved, upward-pointing arc in the bottom right corner of the slide.

A large orange circle is positioned on the left side of the slide, partially cut off by the edge.

What is Jupyter Notebook, and How to use it?

- The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.
 - Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.
 - Python is a requirement (Python 3.3 or greater, or Python 2.7) for installing the Jupyter Notebook itself.
- 
- A series of four yellow curved dashes are located in the bottom right corner of the slide.

Why do we need Data Types?

- Data types are the classification or categorization of data items.
- It represents the kind of value that tells what operations can be performed on a particular data.
- Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

Built In Data Types

- In Python, numeric data type represent the data which has numeric value. Numeric value can be integer, floating number or even complex numbers. These values are defined as int, float and complex class in Python.
- **Integers** – This value is represented by int class. It contains positive or negative whole numbers (without fraction or decimal). In Python there is no limit to how long an integer value can be.
- **Float** – This value is represented by float class. It is a real number with floating point representation. It is specified by a decimal point. Optionally, the character e or E followed by a positive or negative integer may be appended to specify scientific notation.
- **Complex Numbers** – Complex number is represented by complex class. It is specified as *(real part) + (imaginary part)j*. For example – 2+3j

Sequence data types

1) String

- In Python, a string is a collection of one or more characters put in a single quote, double-quote or triple quote. In python there is no character data type, a character is a string of length one. It is represented by str class.

2) List

- Lists are just like the arrays, declared in other languages which is a ordered collection of data. It is very flexible as the items in a list do not need to be of the same type

Sequence data types

3) Tuple

- Just like list, tuple is also an ordered collection of Python objects. The only difference between tuple and list is that tuples are immutable i.e. tuples cannot be modified after it is created. It is represented by `tuple` class.

4) Mapping Data Type-Dictionary

- Dictionary in Python is an unordered collection of data values, used to store data values like a map, which unlike other Data Types that hold only single value as an element, Dictionary holds `key:value` pair. Key-value is provided in the dictionary to make it more optimized. Each key-value pair in a Dictionary is separated by a colon `:`, whereas each key is separated by a 'comma'.

Range Function

- The `range()` method returns an immutable sequence of numbers between the given start integer to the stop integer.
- `range()` constructor has two forms of definition:
 1. `range(stop)`
 2. `range(start, stop[, step])`
- `range()` takes mainly three arguments having the same use in both definitions:
 1. **start** - integer starting from which the sequence of integers is to be returned
 2. **stop** - integer before which the sequence of integers is to be returned.
The range of integers ends at `stop - 1`.
 3. **step (Optional)** - integer value which determines the increment between each integer in the sequence

How to take Input

- The `input()` function takes input from the user and returns it.
- The `input()` function takes a single optional argument:

prompt (Optional) - a string that is written to standard output (usually screen) without trailing newline

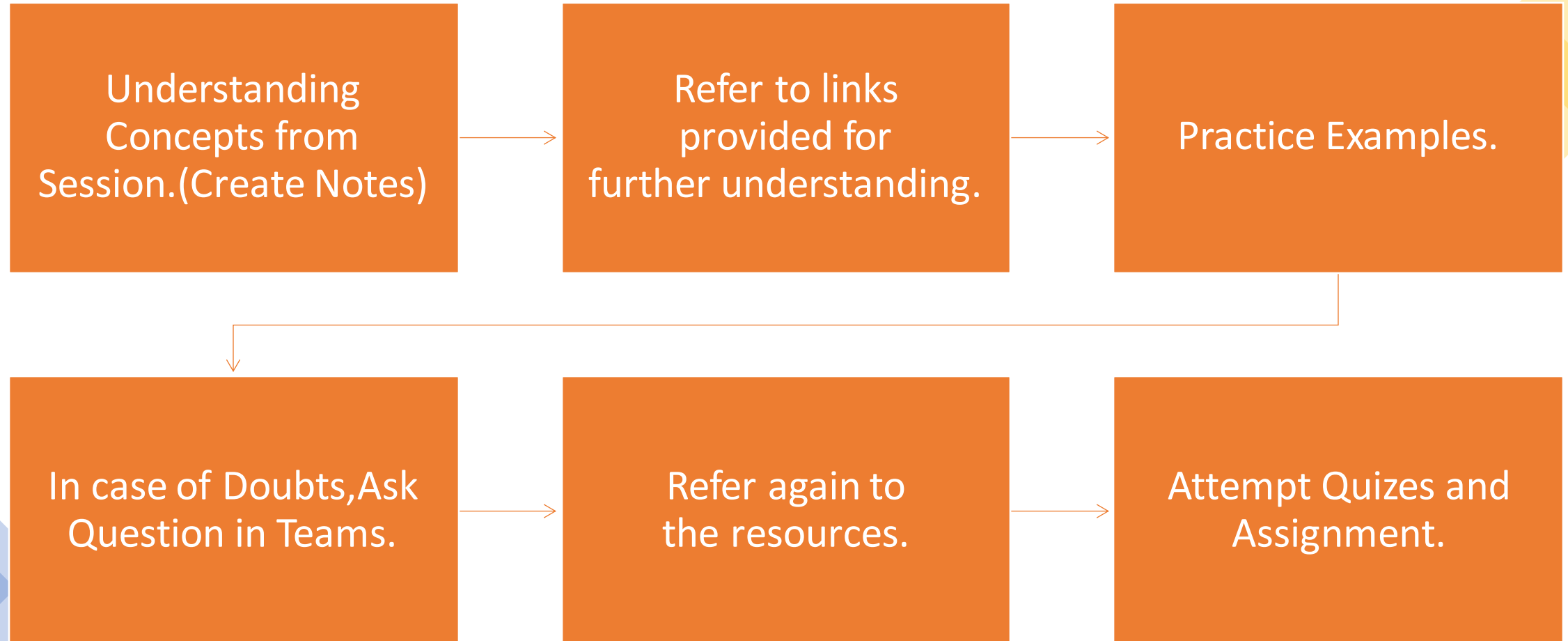
- The `input()` function reads a line from the input converts the line into a string by removing the trailing newline, and returns it.

Comments

- Comments are descriptions that help programmers better understand the intent and functionality of the program.
- They are completely ignored by the Python interpreter.
- Using comments in programs makes our code more understandable. It makes the program more readable which helps us remember why certain blocks of code were written.
- Other than that, comments can also be used to ignore some code while testing other blocks of code. This offers a simple way to prevent the execution of some lines or write a quick pseudo-code for the program.

- In Python, we use the hash symbol # to write a single-line comment.
- Even though there is no unique way to write multiline comments in Python, we know that the Python interpreter ignores the string literals that are not assigned to a variable.
- The multiline string isn't assigned to any variable, so it is ignored by the interpreter. Even though it is not technically a multiline comment, it can be used as one.

Approach to learning Python



Anyone ??

