# Data types

Ch-3

# How to write comments

- # This is single line comment.

- ‘’’
- This line is part of multiline comment.
- This line is part of multiline comment.
- This line is part of multiline comment.
- ‘’’

# Docstrings

- Written as first statements in a module/function/class/method..
- E.g.

```
def add(x,y):
        “””
        This function takes two numbers and find their sum.
        It displays the sum as result.
        “””
        print(“Sum = “, (x+y))
##
C:\<path>\python –m pydoc –w filename.py
```

# How python sees variables

- Do not declare variables
- Start using them
- In python, a variable is seen as a tag (or name) that is tied to some value.
  - x = 7
  - Means, the value 7 is created first in memory and then a tag by the name ‘x’ is created.
  - x ------> 7
  - Python considers the values (i.e. 7 or 9 or 1 or 3) as ‘objects’
  - …

# How python sees variables

- In python, a variable is seen as a tag (or name) that is tied to some value.
  - …
  - If, we change the value of ‘x’ to a new value 9:
    - x = 9
    - Then, the tag is simply changed to the new value (or object) as-
      - x --------> 9  7
      - Since, the value ‘7’ becomes unreferenced object, it is removed by the garbage collector.
  - Assigning, one variable to another variable makes a new tag bound to the same value.
    - y = x
      - New tag ‘y’ will be created that refers to ‘2’ as –  x --------> 9
      - ∴  y

# Data types

- 5 built-in types:
  - None type
  - Numeric type
  - Sequences
  - Sets
  - Mappings

# None type

- It represents an object that does not contain any value.
  - Same is ‘null’ object in java.
- In a python program, only one ‘None’ is provided.
- Usage:
  - It is used inside function as a default value of the arguments.
  - In Boolean expressions, None datatype represents ‘False’.

# Numeric types

- int
- float
- Complex
  - datatypes.ipynb

# Representing binary, octal & hex numbers

- Converting datatypes explicitly
  - x = 7.9
  - int(x)

- datatype.ipynb

# Bool datatype

- True (internally represented as 1)
- False (internally represented as 0)

- datatype.ipynb

# Sequences in python

- Sequence
  - Represents a group of elements or items.
  - str
  - bytes
  - bytearray
  - list
  - tuple
  - range

# str datatype

- str=‘ddu’
- str=“ddu”
- str=“””This is multiline string.
  Dharmsinh Desai University.
  Nadiad.”””
- str=‘’’This is multiline string.
  Dharmsinh Desai University.
  Nadiad.’’’

# Bytes datatype

- Represents a group of byte numbers just like an array does.
- A byte number is any positive integer from 0 to 255 (inclusive).
- Can’t store negative numbers.
- arr = [10,20,30,40,50]
- barr = bytes(arr)

# bytearray datatype

- Similar to bytes datatype.
- Difference:
  - the bytes type array cannot be modified
  - but bytearray type array can be modified.
    - i.e. any element or all elements of the bytearray can be modified.
- Example:

```
arr = [10,20,30,40,50]
barr = bytearray(arr)
print(barr[0]
```

# list, tuple & set : datatypes

- Will be discussed in later unit.

# User-defined datatypes

- Array
- Class
- Module

# identifiers

- It is a name that is given to a variable or function or class etc.
- Can include:
  - Letters
  - Numbers
  - Underscore character (_)
- Should always start with a nonnumeric character.
- Special symbols not allowed: ?, #, %, $, @
- Python is case sensitive.

# Reserved words

- Keywords
- 35

# Naming conventions

- Packages:
- Modules:
- Classes:
- Global variables or module-level variables:
- Instance variables:
- Functions:
- Methods:
- Method arguments:
- Constants:
- Non accessible entities