Contents

[1. Python Basics 2](#_Toc33431911)

[Rules 2](#_Toc33431912)

[Data Types 3](#_Toc33431913)

[variable declaration 3](#_Toc33431914)

[Dynamic Typing 3](#_Toc33431915)

[Strong Typing 3](#_Toc33431916)

[Operator 3](#_Toc33431917)

[Numeric Operator 3](#_Toc33431918)

[Arithmetic Operator 3](#_Toc33431919)

[Logical operator 3](#_Toc33431920)

[Print format 4](#_Toc33431921)

[Conditional Loop 4](#_Toc33431922)

[String Objects 4](#_Toc33431923)

[String operation method 4](#_Toc33431924)

[Tuple 4](#_Toc33431925)

[Sets 4](#_Toc33431926)

[Dictionary 4](#_Toc33431927)

[Map 4](#_Toc33431928)

[Reduce 4](#_Toc33431929)

[Filter 4](#_Toc33431930)

[Conversion function 4](#_Toc33431931)

[Python Advances 5](#_Toc33431932)

[Modules 5](#_Toc33431933)

[Exception Handling 5](#_Toc33431934)

[Database programming 5](#_Toc33431935)

[Numpy 5](#_Toc33431936)

[Pandas 5](#_Toc33431937)

[Data Visualization(Matplotlib, seaborn, Plotly, Cuflinks) 5](#_Toc33431938)

[Projects 5](#_Toc33431939)

[Web Scrapping 5](#_Toc33431940)

[Image scrapping with deployment using Flask 5](#_Toc33431941)

[Machine Learning 5](#_Toc33431942)

[Linear regression 5](#_Toc33431943)

[Logistics regression 5](#_Toc33431944)

[Neutral networks 5](#_Toc33431945)

[CNN 5](#_Toc33431946)

[YOLO 5](#_Toc33431947)

[Hands on above topics 5](#_Toc33431948)

[Example Program 5](#_Toc33431949)

[Input from end user 5](#_Toc33431950)

# Python Basics

## Rules

1. There are not semi colons
2. There are no curly braces for making start /end of a module/if/loop
3. We do not declare variables
4. For single line comment we use “#” symbol for a single line comment
5. Does not support multi line comment
6. A newline terminates each individual statement
7. You can also use semicolon to separate statement example principal =1000 ; rate = 0.05
8. printing to console/reading from keyboard the datatypes which is associated with the value is always a string

## Data Types

* number - int, float
* string - a collection of chars
* Boolean - true/false

### How to declare variable

*# No need to declare data types with variable*var = 32 *# treated as integer*print (var)  
print (type(var)) *# prints data type*dbl = 1.5 *# float variable*print (dbl)  
print (type(dbl)) *# prints data type*str = **"hello"**print (str) *# String data*print (type(str)) *# prints data type*isTrue = **False**print (isTrue) *# boolean data*print (type(isTrue)) *# prints data type*

print(id(a)) – id keyword prints memory address

## Example Program

### How to take Input from end user

var = input("Enter the number:")

print(type(var))

Q1. To calculate area of circle where radius is provided by the end user

- ask user to enter radius

- store it into var rad

- calculate area using the formula area = 3.141\*rad\*rad

- print a read

Q2. Write down a program which takes user input for principle, rate and time, based on user input calculate the simple interest the user has to pay

principal-int

rate-double

time-int

SI = (P\*R\*T)/100

### Dynamic Typing

### Strong Typing

### PPS model in Python

# Operator

## Numeric Operator

The following operations can be applied to all numeric types -

* x + y addition
* x – y subtraction
* x \* y multiplication
* x / y division
* x \*\* y power of x
* x % y Modulo (x mod y)
* --x unary minus
* +x unary plus

Build in functions on all numeric types

* Abs(x) – absolute value
* divmod(x ,y) - return (int ( x / y), x % y)
* pow (x , y [, modulo] – returns (x \*\* y) % modulo
* round (x, [n]) – returns to the nearest integer (floating point number only)

## Comparison Operator

* x < y – less than
* x > y – greater than
* x == y – equal to
* x != y – not equal to
* x >= y – greater than or equal to
* x <= y – less than or eaual to

## Arithmetic Operator

## Logical operator

X or y – Logical or - if x is false, return Y, otherwise return x

X and y – Logical and - if x is false, return x, otherwise return y

Not x - logical negation : if x is false, return 1, otherwise 0

## Operations on Sequence

The following operators can be applied to sequence types including strings, lists and tuples :

* s + r – concatenation
* s \* n – makes n copies of s where n is an integer
* s % d – String formatting (strings only)
* s[i] – indexing
* s [ I : j ] – slicing only
* x in s, x not in s – membership
* for x in s: - iteration
* len(s) – length
* min(s) – Minimum item
* max(s) – maximum item

## Print format

# Conditional Loop

## If condition

a=10  
b=20  
  
*####################################################  
#Simple If test  
####################################################***if** a<b:  
 z=b  
**else**: *# else is optional* z=a  
print (z)  
  
*####################################################  
#To create an empty clause, use the pass statement  
####################################################*a=11  
b=12  
t = 0  
*# Simple if test***if** a<b:  
 **pass** *#To create an empty clause, use the pass statement***else**:  
 t=a  
print (t)  
  
*####################################################  
# Form Boolean expression using or, and and not keywords  
####################################################*c = 5  
**if** b >= a **and** b <=c:  
 print (**"b is between a and c"**)  
**if not** (b < a **or** b > c):  
 print (**"b is still between a and c"**)  
  
*####################################################  
# Use elif statment to handle multiple tests  
####################################################*a = **"\*"  
if** a == **'+'**:  
 op = **"PLUS"  
elif** a == **'-'**:  
 op = **"MINUS"  
elif** a == **'\*'**:  
 op = **"MULTIPLY"  
else**:  
 **raise** RuntimeError(**"Unknow operator"**)  
print (op)

If condition with else

For loop

For I in s :

Statements

For loop with else

Break

Continue

While

Range function

## String Objects

### String operation method

List objects

# Tuple

# Sets

# Dictionary

# Map

# Reduce

# Filter

OOPS concept and working with files

# Conversion function

this method convert one data type to another

* int() - convert string to int
* float() - convert string to float
* str() - converts number/bool to string

Example :

var = int(input("enter the number : ")) - first input will be read from the keyboard and then converted to integer value and assigned to the var

# Python Advances

## Modules

## Exception Handling

## Database programming

# Numpy

# Pandas

# Data Visualization(Matplotlib, seaborn, Plotly, Cuflinks)

# Projects

## Web Scrapping

## Image scrapping with deployment using Flask

# Machine Learning

## Linear regression

## Logistics regression

## Neutral networks

## CNN

## YOLO

# Hands on above topics