**UNIT – 2**

**DATA,​ ​EXPRESSIONS,​ ​STATEMENTS**

***Question Bank***

1. **Give a short note on python.**

Python is an interpreted high-level programming language for general-purpose programming. It is dynamically object oriented programming. It is and open source software.

1. **What are all the data types supported by python?**

The following data types are supported by the python language.

* Integer [Binary, Octal, Hexadecimal]
* Float
* String
* List
* Tuple

1. **Define list in python.**

In python list is an ordered sequence of items. Items can be any type and it is separated by comas. List is flexible data type. We can add and remove items any time. List are inerrable it is access based on index value. Lists are mutable.

Eg: a\_list = [1,3,2,5,6]

1. **Define tuple in python.**

In python tuple is an ordered unchangeable sequence of items. Items cam be any types and each one separated by comas. The accessing of elements is based on index and it is inerrable like list. Tuple’s are immutable. Once it is created we can’t change the elements by number’s or values.

Eg: a\_tupple = (1,2,3,5,4,7)

1. **What is variable in python?**

The name that mentions the value is called variable. It is a specific location in the memory where certain value or data stored. Variable names should be unique. Each variables holds separate memory location.

***Rules for usage of variable names:***

1. Name may contain letters, numbers and underscore.
2. It must start with letter.
3. Special characters (@,$,#...etc) not allowed.
4. Variable names are case sensitive.

1. **What is identifier in python?**

A valid string that can be used as name in programming languages it is known as identifiers. Variable name, class names, function names are all known as identifiers**.**

***Rules for usage of identifiers:***

1. Name may contain letters, numbers and underscore.
2. It must start with letter.
3. Keyword should not be used as identifiers.
4. Special characters (@,$,#...etc) not allowed.
5. **What is the difference between (and) operator and (&) operator?**

Both are recognized as operators in python with operands.

* ***and*** – is logical operator that checks for Boolean values of both operands. If both values are true then and expression evaluation is true. Eg True ***and*** True – True , True ***and*** False – False
* ***& -*** Is a bitwise operator it receives two integer operands and performs bitwise comparison of the operands binary values. Eg: 4&5 – 4 [100 & 101 - 100] .

1. **Write a program that receives two numbers from the user and find which one is great?**

a = int(input(“enter number 1”)

b = int(input(“enter number 1”)

if (a>b):

print(“number 1 is great”)

else

print(“number 2 is great”)

1. **List the features of python language.**

* Simple
* Open source
* High-level
* Portable
* Object Oriented
* Interpreted
* Easy to maintain
* Scalable

1. **List the operators supported by the python language.**

* Arithmetic Operators
* Relational Operators
* Logical/Boolean Operators
* Bitwise Operator
* Assignment Operators
* Membership Operators
* Identity Operators

1. **What is bit wise operator?**

Bitwise operators are operators that works on binary values of an integer numbers. It receives integer operands and perform bitwise logic on the binary representation of an integer numbers. The following operators are knows as bitwise operators.

**|** [OR], **&** [AND], ~ [Negotiation], **^** [XOR], **>>** [Right shift], **<<** [Left Shift].

1. **What is use of membership operator in python?**

It is special type of operators it works on sequences. It checks for an item is present or not present in a sequence. The sequence can be anything lists, tuples, strings. There are two membership operators available.

***in*** = it checks for item is present in the sequence. If an item present it returns true else it returns false.

***not in*** = it checks for an item not present in the sequence. If not returns true else it returns false.

**Eg:**

alist = [1,4,5,6]

print( 4 in alist ) – True

print ( 10 not in alist ) – True

1. **Evaluate expressions based on operator precedence.**
   1. 45+5\*7/4-2 [ Answer = 51.75 ]
   2. 4\*\*2/4+3-1 [ Answer = 6.0 ]
2. **Define functions.**

Functions are the block of reusable code that can be used to perform one or more related operations. It ensures the modularity and reusability of the program. Functions can be used anywhere and anytime within the program.

There are two types of functions [1. Built-in functions, User defined functions].

1. **List four in-built in functions with its descriptions.**

abs() – It returns an absolute value of number.

chr() – It returns character from the integer.

bin() – It converts integer into binary.

min() – It finds minimum in a sequence.

1. **List down the types of functions.**

* ***Based on Parameters***
  + Functions without parameters.
  + Function with parameters.
* ***Based on arguments***
  + Default arguments.
  + Keyword arguments.
  + Arbitrary arguments.

1. **What are modules in python?**

The file in which the python code and function definitions are saved is referred to as modules. Files are saved with extension (.py). Modules can be imported in scripts.

1. **Write a program that helps to convert Celsius to Fahrenheit.**

temp\_c = float(input(“Enter temperature in Celsius : ”))

temp\_f = (temp\_c \* (9/5))+32

print (“Temperature in Fahrenheit :”,temp\_f)

1. **Write a program that swaps the values of two variables.**

a = 10

b = 20

print (“Before Swapping”)

print (“A is”, a, ”B is”, b)

temp = a

a = b

b = temp

print (“After Swapping”)

print (“A is”, a, ”B is”, b)

1. **Write a function that helps to find the factorial of a number.**

def fact\_finder(number):

result = 1

while (number > 0):

result = result \* number

number = number – 1

**PART – B**

1. Explain about data types supported by python language with simple examples.
2. Explain about arithmetic and comparison operators supported by python with simple examples.
3. Explain about assignment operators and identify operators with examples.
4. Explain about bitwise and logical operators with example and distinguish them.
5. Explain about types of functions based on arguments with simple examples.
6. Write a program that helps to find the given year is leap year or not. if it is leap year find that year is sum of digits of the year. [Eg – 2012 is leap sum of digits is 5].
7. Write a program that helps to find the square root of the given number using newton’s iterative method.
8. Write a program that helps to find the e power x with a help of user defined functions.
9. Create a module for simple calculator operations and import them into another program.
10. Write a program that helps to calculate the grade of the student based on mark obtained in five subjects.