Assignment (1)

1) Explain programming and python in details
- Defination and purpose of programming
- characteristics and applications of python.
- Types of comments in python with syntax.

-Answers:-

Defination of programming :-

programming is the process of giving instructions to the computers. It is a step-by-step process which gives instructions as code to perform a specific task. As computer can only understand binary code. Then the computer converts the instructions into Machine-level language to process the specific task and give the result as output.

purpose of programming :-

programming language is used to make human life better and easier.

By using programming we can develop mobile applications. create web applications, web development.

programming is used to as it can do multiple task at once at any number of times.

It gives perfect and correct information if we give correct instructions.

Characteristics of python:-

python is a high-level, object-oriented, interpreted programming language which is used to develop software Web development. Game development, creating websites

It is very easy and simple to use as its syntax

are very easy to learn. As well python is case-sensitive.

-Easy to learn and use.

python syntax are simple and easy to learn as it is often uses english words. It is beginner friendly and easy to learn compared to other programming languages like C++, Java, etc.

Interpreted language:-

python is a interpreted language which is known as it is a step-by-step process - the code (program) is executed line-by-line instead of executing the whole program at once. This process helps us to debug the errors easily at runtime.

High level language:-

python is a High-level, object-oriented programming language so it converts to human-readable language and sends the result as output

Cross platform:-

python is a cross-platform mean an independent programming language. it can on any operating systems. like MacOS, Linux, Windows etc.

Applications of python:-

Web Development.

python is used for web development, creating websites, webpages, for developing Games and web applications.

Data analysis:-

python helps us in analyzing data and finding the data patterns.

Game development :-

python is used for creating and developing games and also animations which are developed using python

Education :-

python is used in schools, colleges and research fields as it is beginner-friendly which is very simple and easy to learn and its syntax are easy to learn.

Types of comments in python with examples :-

In python comments are used to explain the code (program). There are three types of comments in python. They are:

- single line comment
- multi line comment
- Documentary string.

single line comment :-

single line comment is used to comment only one line

syntax :- # This is a single-line comment

example :-

print ("Hello World"); # printing a message

multiline comment :-

It used to comment multiple lines

syntax

```
# line 1
# line 2
# line 3.
```

Example :-

```
a = 10
b = 20 # This program calculates.
print (a+b) # sum of the two numbers
```

Inline comment :-

It is written at the end of the statement and is used to explain a specific part

syntax :-

" This is a
multi line comment
in python.

Importance of python in modern software :-

Simple syntax :-

python syntax are close to english and its syntax are very easy & simple to learn.

python is a beginner friendly and faster to learn it has less code.

supports AI and ML

python is mostly prefered language for AI and ML language because it support libraries like pandas, numpy, Tensorflow.

platform independent

python is platform independent

used By Big companies :-

python is used by companies like google netflix, and nasa

2) Describe data types and operators in python with suitable examples.

Built in data types in python (Numeric, set, sequences, Mapping examples.

- Built in data types in python.
- Type identification using type().
- various python operators (Arithmetic, comparison, logical, Membership concatenation, identity)
- Real-world usage of operators.

Data types:-

Data types tells what kind of value is stored in the variable such as integer, float, character, etc.
python has different data types

Numeric data type:-

Numeric data types are used to store numbers. They are in numeric form.

Ex:-

```
int = 2
float = 2.14
complex = 3+5j
```

Boolean data types:-

Boolean data-types are used in conditions as comparision It only store the true false values.

Ex:-

```
print(10>5) # True
print(10<5) # false.
```

Set data type:-

It stores only unordered data items duplicates are not allowed set data types uses {}

Ex:- {20, Harshi, 26.14}

Mapping Data types:-
Mapping connects one value to another value. It is used to store data in the key-value pairs
-ex:- student: {"name": "Harshi"
"Grade": "Bsc computers"
"age": 20}

Sequence data type :-
A sequence data types the collection of items in an ordered manner.
List:- List is stored in ordered manner and it is mutable which can be changed & allows duplicates these are written in between [ ]
ex:-
fruits: ["apple", "Banana", "mango"],
Tuple:-
It is in ordered manner and immutable which cannot be changed and Allows duplicates
ex:-
String:- colors = ("red", "Blue", "pink")
string is a sequence characters which are store the values in an ordered manner string is immutable.
name = "Madhu"
The value of data type must be written in " ".
Type identification using type():
In python type() is used to identify the data type of a variable or value it is used when you are debugging code.
Syntax:- type(value)
Numeric type
Ex:- x = 10
print (type(x)) # <class 'int'>

501

Float type:-

Ex:- y=3.14
print (type (y)) # <class 'float'>

String type:-

Ex:- name = "Madhu"
print (type (name)) < class' str'>

Boolean type:-

is - valid = True.
print (type (is - valid)) # class 'bool'>.

List :-

nums = [1,2,3]
print (type (nums)) # class <'list'>.

Tuple:-

t = (1,2,3)
print (type (t)) # class <'tuple'>

Set

S = {1,2,3}
print (types (s)) # class <'set'>.

Dictionary :-

d = {name": "Madhu", "age" :21}
print(type(d) # class <'dict'>.

Various operators :-

operator :- It is a symbol that tells computer to perform certain Mathematical and logical operations. (operations).

operator	Meaning	Example.	Result.
+	Addition	10+5	15
-	subtraction	10-5	5
*	Multiplication	10*5	50
/	Division	10/5	2.0
//	floor division	10//3	3.

%	Modules	10 % 3	1
**	Exponent	2 ** 3	8

Comparison operators:-

used to compare values. Returns True or False.

operators	Meaning	Example	Result
==	equal to	5 == 5	True
!=	not equal to	5 != 10	True
>	Greater than	10 > 5	True
>=	Greater than equal to	10 >= 0	True
<	less than	3 < 10	True
<=	less than or equal to	3 <= 15	False.

Assignment operators :-

It is used to assign the values to variables.

operators	Meaning	Example	Result.
=	Assign	x = 5	x = 5
+=	Add & Assign	x += 3	x = x + 3
-=	sub & Assign	x = 3	x = x * 3
*=	Multiply & Assign	x *= 3	x = x / 3
/=	Divide & Assign	x /= 3	x = x // 3
//=	floor & Assign	x //= 3	x = x % 3
%=	Modulus & Assign	x %= 3	x = x ** 3.
**=	exponent & Assign.	x **= 3	

Logical operators :-

it is used to combine the conditional statements.

3) Explain python input & output operations in detail

Include the following in your explanation:-

- Input() function & it's default data type.
- Type conversion while taking input
- Taking multiple inputs.
- formatted output using print(), separators, and format specifiers.

Input() function:-

Input() function are used to us take input from the user

syntax:- var = input("Message")

It always returns data as strings by default.

default datatypes:-

Data taken using input() is always stored as str.

Type conversion:

The convert input to other data types.

=) convert to integer

num = int(input("enter a number"))

=) convert to float:

price = float(input("enter price:"))

=) convert to boolean.

Type conversion while taking input:

→ input() always takes data as string (str) by default

→ To use number (for Math), we must convert the input

⇒ common conversions:

int() → converts input to integer

float() → converts input to decimal number.

Example:

```
a = int(input("Enter age:"))
b = int float(input("enter price:"))
```

=) Without conversion:

operator	Meaning	Example	output
And	True if both true	True and false	false
or	True if any true	True & false	True
not	Reverse result	not true	false.

Membership operator:-

it is used to check weather the value is present in sequence

operator	Meaning	Example	output
in	True if value present	"a" in apple	True
not in	True if not present	"b" is not in apple	True.

Identity operator:-

it is used to compare memory location of two objects.

operator	Meaning	Example	Result
is	True if same object	x is y	True/false
is not	True if not same object	x is not y	True/false.

Ex:-

```
x = [1,2,3]
y = x
z = [1,2,3]
print(x is y) # True.
print(x is z) # false.
```

x = int (input ("5")) -> 5 (integer)
Y = input ("3") -> "3" (string)
x+y = "153" (string concatination)

With conversion:-

x = int (input ("5")) -> 5 (integer)
y = float (input (5.8)) -> 5.8 (decimal).

Taking multiple inputs:-

Sometimes we need more than one input from the user

Method:

a, b = input().split()

- user enters values in one line.
- default type is string

example:- a, b = input ("enter two numbers : ").split()

*) Type conversion with split():

If we need integer values:

a, b = map (int, input().split())

if we need float values:

x, y = map (float, input().split()).

- formatted output using print(), separators and formal specifiers.

=) print() is used to display output

=) sep changes the separator b/w values

ex:- print (1, 2, 3, sep = "=") ++ 1=2=3

=) "end" changes the ending of the line

Ex :- print ("Hello", end = " ")

=) format is used to insert values in a formatted way.

ex:-

Name: {}", format (name).

f - strings are used modern formatting

ex:- f "Name : {name}"

K501

4) discuss control statements and decision Making statements in python.

Your answer should cover:-

- Meaning and Importance of control statements.
- Types of control statements.
- Decision Making statements:
  if, if-else and if-elif-else.
- Syntax flow and execution control with examples.

Meaning:-

control statements are special instructions in a programming language that control the flow of execution of a program they decide which part of the code runs, how many times it runs and under what conditions it runs.

Importance of control statements:-

Decision Making:-

control statements allow program to Make decisions and choose different actions based on conditions

Reduce code length:-

using loops and conditions reduce duplicate code making programming shorter & cleaner.

Repetition of tasks:-

loops help execute a block of code Multiple times without writing in repeatedly.

Types of control statements:-

In python there are mainly three types of control statements. They are.

Conditional:- used to check conditions & decide what to do
- if
- if else
- elif.

Looping statements:-
Used to repeat a block of code automatically until it meets a specified task
- While loop
- for loop.

Jumping statements:-
Used to change the normal flow inside loops.
• break • continue • Return

Decision Making statements:-

if statement:-
used when you want to check one condition if the condition is true then the code runs.

Ex:- age = 18
if age == 18:
print ("you are adult").

if else statement :- used when you have two possible outcomes

Ex:- marks = 38
if marks >= 35:
print ("pass")
else:
print ("pass")

if-elif-else statement.
used when you have multiple conditions.

Ex:- Marks = 75
if marks >= 90:
print ("Grade A")
elif marks >= 75:
print ("Grade B")
else:
print ("Grade c").

flow and execution control:-

the program checks the condition: if the condition is true then the statements inside are executed if not block is skipped.

ex:-

```
num=5
if num>0:
  print("positive number").
```

if else statement:-

syntax:-

```
if condition:
    statements
else
    statements.
```

flow and execution control:

condition is checked.

If true, if block runs.

If false, else block runs.

ex:-

```
marks=30
if marks>=35:
    print("pass")
else:
    print("fail").
```

5) Write an essay on python programming fundamentals.

Explain:-

- Role of programming in problem solving.
- python syntax simplicity and readability.
- use of comments for code documentation.
- Data types, operators and input/output operations.
- control flow using decision-making statements.

Role of programming in problem solving:-

programming plays an important role in problem solving. because it helps us to tell the computer what to do in clear and step-by-step way.

Breaking problems into steps:-

Big problems can be divided into smaller and easier steps so it helps us to tell the computer what to do in in clear & step-by-step way.

Breaking problems into steps:-

Big problems can be divided into smaller & easier steps so it becomes simple to solve.

Giving fast & correct results:-

computers follows instructions exactly. so the result are faster & more accurate than the humans.

Improving logical thinking:-

While writing a program we need to think logically and plan properly which improves our problem solving skills in real life too.

python syntax simplicity & readability:-

python is known for having a simple and clean syntax

- easy to understand.
- easy to write.

Simplicity in syntax:-
python focuses on writing less code with clean meaning.
for ex:- in python we dont need : or {} for basic structure
Redability:-
Means code is equal to read understand, even after many
Mainly python code looks clean and neat like plain english.
- developers can find errors easily.
- Learns can works better.

use of codes comments for code documentation:-
comments help other understand the purpose of code.
Helps in debugging:-
By commenting out certain lines temporaily we can test
Data types:-
data types defines the type of data a variable can hold. They help the computer understand to store and use data.
common data types in python:
int - store whole numbers.
float - stores decimal numbers.
str - stores text characters.
boolean - True/false is stored.
list - collection of items.
Tuple - similar to list but cannot be changed.
operators:-
operators are symbols that perform operators on data.

input/output operators:

These operations allow communication b/w user & program.

input:-

Input() function is used to take data from the user

```
Name = input("Enter your Name")
```

① **Program on movie ticket pricing.**

```
age = int(input("enter your age:"))
is3D = int(input("Enter if you are Watching a 3D
                  Movie else enter 0:"))
if is3D == 1:
    if age<13:
        print("your ticket price ₹200")
    elif age >=12 and age >=59:
        print("your ticket price is ₹ 300")
    else:
        print("your ticket price is ₹ 250")
    elif age <13:
        print("your ticket price is ₹ 150")
    elif age >=13 and age<=59:
        print("your ticket price is ₹ 250")
    ebe
        print("your ticket price is ₹ 200")
```

output:-

Enter your age: "

Enter, if you are working a 3D movie else enter 0:"

your ticket price is ₹ 200.

2 College Attendence Rule:

```
atten=int(input("Enter your attendence percentage :"))
medic=int(input("Enter if you have medical certificate
                if-you-hav else 0 :"))
if atten>=75 (or) (atten>=60 and medic==1):
   print("Allowed to write exam")
else:
   print("Not allowed to write exam")
```

output :-

Enter your attendance percentage :70
Enter 1 if you have medical certificate else 0:1
You are not allowed to write the exam.

program on E-Commerce Discount:

```
billamount=int(input("Enter your bill Amount:"))
is prime=int(input("Enter 1 if you have prime
                    else 0:"))
if is prime==1:
    if billamount>=5000:
      dis=0.25*billamount
  elif 2000<=billamount<=4999:
      dis=0.15*billamount
  else:
      dis=0
  else
   if billamount>=5000:
    dis=0.2*billamount
   elif 2000<=billamount<=4999:
      dis=0.10*billamount.
```

```
else
 dis=0
amount=billamount-dis
print("final amount to be paid is", amount).
```

output:-

```
Enter your bill amount: 4500
enter 1 if you have prime member else 0:0
final amount to be paid is 4050.0.
```

4) smartphone Battery warning.

```
battery = int(input("Enter your battery percentage:"))
ischar = int(input("Enter your 1 if phone is charging else 0"))
if ischar==1:
   print("charging")
elif battery<=20:
   print("low Battery")
elif battery<=80:
   print("Normal").
else:
   print("Enter 0 or 1").
```

output:-

```
Enter your battery percentage :87.
Enter 1 if phone is charging else 0:1
Charging.
```

5) program on Driving license check.

```
age=int(input("Enter your Age:"))
testpassed = int(input("Enter 1 if you passed the test
                        or else 0:"))
if (age>=18 and testpassed==1) or age>=60:
   print("Eligible")
else:
   print("Not Eligible").
```

output:-

Enter your Age : 60

Enter 1 if you passed the test or else 0 : 1

Eligible.

6) program on Conline food delivery.

```
amount int (input ("Enter order amount:"))
is Gold = int (input ("Enter 1 if you are a Gold member
                       else 0 :"))
dis = int (input ("Enter distance in km:"))
if (amount >= 500 and dis <10) or (is Gold == 1 and
                                   dis<10):
    print ("free delivery")
els:
    print ("delivery is never free").
```

output:-

Enter order amount : 700

Enter 1 if you want are a Gold member or else
a : 1

Enter distance in <m : 11

delivery is never free.

7) program on Bank loan Approval.

```
Salary = int (input (" Enter the salary :")
score = int (input ("Enter the credit score :"))
if salary >= 30000 and score >= 70 or salary =
                                    50000:
    print ("Loan approved")
else :
    print ("Loan rejected").
```

Output:-

Enter Salary : 300
Enter the credit score : 800
Loan reseeted.

8) program on electricity bill.

```
Bill1 = int (input ("Enter first units :"))
bill2 = int (input ("Enter next units :"))
  b = int()
if bill = = 100:
    b = 2
      if bill2 == 100:
        b = 3
  bill = bill + bill2
   if bill > 200:
       b = 5
  print ("final bill amount :", b).
```

output:-

Enter first units : 100
Enter next units : 100
final bill amount : 3.

9) program on student scholarship.

```
Marks = int (input ("Enter your marks :"))
famin c = int (input ("Enter family income: "))
if sinpar = int (input ("Enter , if you have single
                    parent else 0 :"))
```

```
if ssenpar==1 and marks>=85:
print("you are eligible for a scholarship").
elif marks>=85 and famic<500000:
    print("you are eligible for a scholarship").
else:
    print("you are not eligible for a scholarship")
```

output:-

```
Enter your marks: 85
Enter family income: 600000
Enter 1 if you have single parent else 0:1
you are eligible for a schloarship.
```

10) program on online Exam Result.

```
theory=int(input("Enter your theory marks:"))
practical=int(input("Enter your practical marks:"))
total = theory + practical
if total>100 or (theory)=40
  print("pass")
else:
    print("fail").
```

output:-

```
Enter your theory marks:11
Enter your practical marks:91
pass.
```

11) program on Hotel Room pricing.

```
Weekend=int(input("Enter if it is weekend or not
```

```
daysstay = int (input ("Number of days stayed : "))
    b = int ()
    if weekend == 1 :
        b = 4000
        print ("final amount :", b)
    elif weekend == 0 :
        b = 3000
        print ("final bill :", b)
    if days stay >= 3 :
        d = daysstay * b
        discount = d * 0.05
        final_amount = d . discount
        print ("final amount :", final_amount)
```

output :-

```
Enter if it is weekend or not : 0
Number of days stayed : 4
final amount : 11400.
```

12 program on gaming level unlock.

```
score = int (input ("Enter score :"))
premium = int (input ("Enter premium pass :"))
used cheat = int (input ("used cheating :"))
    if used cheat == 1 :
        print ("Access denied")
        exit()
if score >= 100 or premium == 1 :
```

print("Game next level unlocked").

Output:-

Enter score: 333
Enter premium pass: 1
used cheating: 1
process denied.

13 program on office Entry system.

```
id = int(input("Enter if id is valid or not: "))
facescan = int(input("Enter if face scan is matched
                    or not: "))
fingerprint = int(input("Enter if finger print is
                    Matched or not: "))
is holiday = int(input("Enter if it is holiday or
                    not: "))
if it is holiday == 1:
    print("Entry is denied")
    exit().
if id == 1 and fauscan == 1 or finger print == 1:
    print("entry is allowed")
else:
    print("Entry is denied).
```

Output:-

Enter if id is valid or not: 1
Enter if face scan is matched or not: 0
Enter if it is holiday or not: 0
Entry is allowed.

14 program on Movie rating display.

```
averagerating= float (input ("Enter the average rating:"))
editor choice = int (input ("Enter if editor choice or
                                    not:"))
if editorchoice==1:
    print ("Recommended)
    exit()
if averagerating>=8.5:
    print ("Excellent")
elif 6.0<averagrating >8.4:
    print ("Good")
elif averagrating <=6.4:
    print ("Average").
```

output:-

Enter the average rating: 8.5
enter if editor choice or not: 0
Excellent.

15 program on mobile Data usage:

```
dataused =int (input ("Enter data used in GB"))
unlimpla = int (input (" Enter 1 if you have unlimited
                                    plan else 0"))
if Roaming = int (input ("Enter 1 if roamingis an
                                    else 0"))
if data used <=2 or unlimpla ==1) and if Roaming ==0:
    print ("A network gives unlimited data")
```

```
elif if Roaming==1:
    print("unlimited plan does not work")
else:
    print("limited data applies")
```

Output :-

```
Enter data used in GB:5
Enter 1 if you have unlimited plan else 0:0
Enter 1 if roaming is on else 0:0
limited data applies.
```