1) What is Python, and why is it popular ?

Ans: Python is a object-oriented ,interpreted , high-level

programming language .

It is popular :

- Simple and easy to learn
- It is free and open-source
- Lot of libraries(1,37,000 nearly)

2) What is an interpreter in Python ?

Ans: Interpreter is a translator , which translates high level programming language to machine language and it translate code 'line by line ''.

3) What are pre-defined keywords in Python?

Ans: Pre-defined keywords are the reserved words in the python that have their fixed meaning , we cannot use them as variable name.

4) Can keywords be used as variable names

Ans: No , keywords are not used as variable names , because they have their fixed meaning.

5) What is mutability in Python?

Ans: Mutability in python means objects can be modified or changes possible. We can change elements.

6) Why are lists mutable, but tuples are immutable?

Ans: List :

- Dynamic Memory Allocation
- Insertion and deletion
- Reference semantics

Tuples:

- Statics Memory Allocation
- No insertion or deletion
- Value Semantics

7) What is the difference between "==" and "is" operators in Python?

Ans: '==' compares the values of two objects, such as numbers, strings ,lists, etc. , whereas 'is' compares the identities of two objects , i.e. , whether they are in same object in memory.

8) What are logical operators in Python ?

Ans: Logical operators in python are used to combine and make decisions based on mutltiple conditions. There are three logical operators. 'AND', ' OR' , 'NOT' .

9) What is type casting in Python ?

Ans: Typecasting is the process of conversion of one data-type to another data-type , It can be categorised in two categories. Explicit and Implicit.

10) **What is the difference between implicit and explicit type casting?**

Ans: Implicit : This is the type-conversion in which the interpreter automatically converts data-type, without loosing any data.It automatically selects size of higher size data-type.

Explicit : This is the type-conversion which is done by user manually , user can change to any data-type according to his/her need .Data may or may not losts.

11)**What is the purpose of conditional statements in Python ?**

Ans: Conditional statements are those statements which alter the normal execution of program and tests the condition to be carried out.

12)**How does the elif statement work**

Ans: The elif statement also known as ‘else if’. It allows you to check another condition if the initial if condition is False.

13)**What is the difference between for and while loops**

Ans: For loop : Initialisation is done in for loop. In for loop , it is necessary in advance to know how many times to execute the loop. Exp -> List, Tuple, etc

While loop: Initialisation is done before while loop , it is not necessary to know in advance ,how many times to execute the loop.

14)**Describe a scenario where a while loop is more suitable than a for loop.**

Ans:Let’s suppose banking system, the ATM has a limited amount of cash available and you want to ensure that the user doesn’t withdraw more cash than is available.

- While loop allows you to continue iterating until a certain condition is met (in this case, the user’s account balance reaches zero or the ATM runs out of cash.
- The user’s withdrawal amount is dynamic and unpredictable . A while loop allows you to handle this dynamic input and respond accordingly.

November 28, 2024

1 Assignment Practical Questions

1.0.1 1) Write a python program to print "Hello world"

[70]:
```
# Program to print Hello world
print('Hello world')
```

```
Hello world
```

1.0.2 2) Write a python program that displays your name and age

[78]:
```
#  Program to print name and age
name = input('Enter name : ')
age=int(input('Enter age : '))
print("Your name is  " , name," and age is :" , age)
```

```
Enter name :  Ravi kumar yadav
Enter age :  18
```

```
Your name is   Ravi kumar yadav  and age is : 18
```

1.0.3 3) Write code to print all the pre- defined keywords in python using the keyword library

[121]:
```
## Two ways to print python keywords
import keyword
print(keyword.kwlist)

print('\n')

keywords = help('keywords')
keywords
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break',
'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for',
'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or',
'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

```
Here is a list of the Python keywords.  Enter any keyword to get more help.

False               class               from                or
None                continue            global              pass
True                def                 if                  raise
and                 del                 import              return
as                  elif                in                  try
assert              else                is                  while
async               except              lambda              with
await               finally             nonlocal            yield
break               for                 not
```

1.0.4 4) Write a python program that checks if a given word is python keyword or not

[145]:
```
# Checking given word is keyword or not
import keyword
key = keyword.kwlist
keyword = input('Enter a word : ')
if keyword in key :
    print('\nIt is Python keyword ' )
else :
    print('\nNot a Python keyword ')
```

```
Enter a word :  for

It is Python keyword
```

1.0.5 5) Create a list and tuple in python and demonstrate how attempting to change an element works differently for each

[65]:
```
#  List
lis_t = [1,34.00,6,79.56,64,5,3578,99,46]
print('List : ',lis_t)
lis_t[4]='Ravi kumar yadav'
print('List after changes : ',lis_t)

      #  Tuple
tup = (1,35,656,87,4.66,23)
print('\nTuple : ',tupl_e)
tup = tup[:2]+('Ravi kumar yadav',)+tup[3:]      # Breaking the tuple and␣
 ↪adding with new element
print("Tuple after changes : ",tup)
```

```
List :  [1, 34.0, 6, 79.56, 64, 5, 3578, 99, 46]
```

```
List after changes :  [1, 34.0, 6, 79.56, 'Ravi kumar yadav', 5, 3578, 99, 46]

Tuple :  (1, 35, 656, 87, 4.66, 23)
Tuple after changes :  (1, 35, 'Ravi kumar yadav', 87, 4.66, 23)
```

1.0.6 6 and 7) Write a function to demonstrate the behaviour of mutable and immutable arguments.

[85]:
```
def mutable_immutable(mutable_arg,immutable_arg):
    print('Before changes')
    print('Mutable argument : ',mutable_arg)
    print('Immutable argument : ',immutable_arg)

    mutable_arg.append(78)
    immutable_arg +=22

    print('\nAfter changes')
    print('Mutable argument : ',mutable_arg)
    print('Immutable argument : ',immutable_arg)
listt = [23,5,46,76,89]
integer = 78
mutable_immutable(listt,integer)
```

```
Before changes
Mutable argument :  [23, 5, 46, 76, 89]
Immutable argument :  78

After changes
Mutable argument :  [23, 5, 46, 76, 89, 78]
Immutable argument :  100
```

1.0.7 8) Write a program to demonstrate the use of logical operators.

[179]:
```
#  Program to turn off alaram using if-else statement.
day = input('Enter today\'s day : ')
time = float(input('Enter current time : '))
if time == 6.00 and (day == 'monday' or 'tuesday' or 'wednesday' or 'thrusday'␣
 ↪or 'friday' or  'saturday'):
    print('\nRing the alaram.')
elif time== 6.00 and day == 'sunday':
    print("\nNot Ring the alarm ")
else :
    print("\nEnter valid day or time ")
```

```
Enter today's day :  friday
Enter current time :  6.00
```

```
Ring the alaram.
```

1.0.8 9) Write a python program to convert user input from string to integer , float and boolean types.

[186]:
```
# Typecasting
string = input('Enter a string : ')
print('Conversion string to int and float is not possible ') ,#␣
 ↪int(string),float(string)
print('Conversion string to boolean : ',bool(string))
```

```
Enter a string :  Ravi

Conversion string to int and float is not possible
Conversion string to boolean :  True
```

1.0.9 10) Write code to demonstrate type casting with list elements.

[14]:
```
# Typecasting
lis = ['1','23','34','56','5666',76,4.56,654,56.576]
integer = [int(i) for i in lis]
print(integer)
```

```
[1, 23, 34, 56, 5666, 76, 4, 654, 56]
```

1.0.10 11) Write a program that checks if a number is positive , negative or zero.

[18]:
```
#program to check number is +ve , -ve or zero
num = int(input('Enter a number : '))
if num>0:
    print('Positive number')
elif num<0:
    print('Negative number')
else :
    print('Zero')
```

```
Enter a number :  -54

Negative number
```

1.0.11 12) Write a loop to print numbers from 1 to 10.

[28]:
```
# Program to print 1-10 no's.
n=1
while (n<=10):
    print(n)
    n=n+1
```

1
2
3
4
5
6
7
8
9
10

1.0.12 13) Write a python program to find the sum of all even numbers between 1 and 50.

[11]:
```
# Program to sum of even no.'s b/w 1-50:
a=0
for i in range(51):
    if i%2==0:
        a=a+i
print('Sum of all even numbers between 1 to 50 is : ',a)
```

```
Sum of all even numbers between 1 to 50 is : 650
```

1.0.13 14) Write a program to reverse a string using a while loop.

[35]:
```
# Program for string reversal
string = 'vaday ramuk ivaR'
reverse = string[::-1]
print("string after reversal is : ",reverse)
```

```
string after reversal is :  Ravi kumar yadav
```

1.0.14 15) Write a python program to calculate the factorial of a number provided by the user using a while loop.

[17]:
```
# Program to find factorial
num = int(input('Enter a number to find factorial : '))
Fact =1
while (num>0):
    Fact=Fact*num
    num=num-1
print('Factorial is : ',Fact)
```

```
Enter a number to find factorial :  7
```

```
Factorial is :  5040
```

[ ]: