NAME= RASHI MEENA

ASSIGNMENT 1

QUES 1) Explain the key features of Python that make it a popular choice for programming?

ANS-

Python is a programming language which is set of rules and symbols that are used to interact with computer and it helps in analysing the data and also make us capable for decision making.

Features:

1) Python language is easy to learn and understand for beginner. It is not complex as other languages like C++, Java and many more .

2)Python has wide range of job opportunities in market which attract the ones to pursue the career in this domain.

3) Python has more libraries than other languages. (Libraries means codes that can be re-use by others)

Currently python has around 1,37,000 libraries which is one of the best feature.

4) Python is compatible with other languages as well .

5) Pythons has capability to understand the data type on its own which is comes under Implicit concept. This feature helps the learner to learn easily without any complexity.

6) Python can be used for Machine learning, Game development, data analysis, Graphics representation etc.

7)During coding, Python verify the codes and indicate the error if it is any and one can easily understand and rectify it.

8) Python codes can be read very easily as it based on basic terminologies which we have learned back in school. For example : a+b , a/b

Any one can understand this syntax '+' ,'-' as it is basic.

9) Python language has many platforms for coding like - Jupyter, anaconda , Google collab, Vscode , GDB compliar, programiz etc.

So, these features makes Python popular among people for programming.

QUES 2) Describe the role of predefined keywords in Python and provide examples of how they are used in a program?

ANS 2-

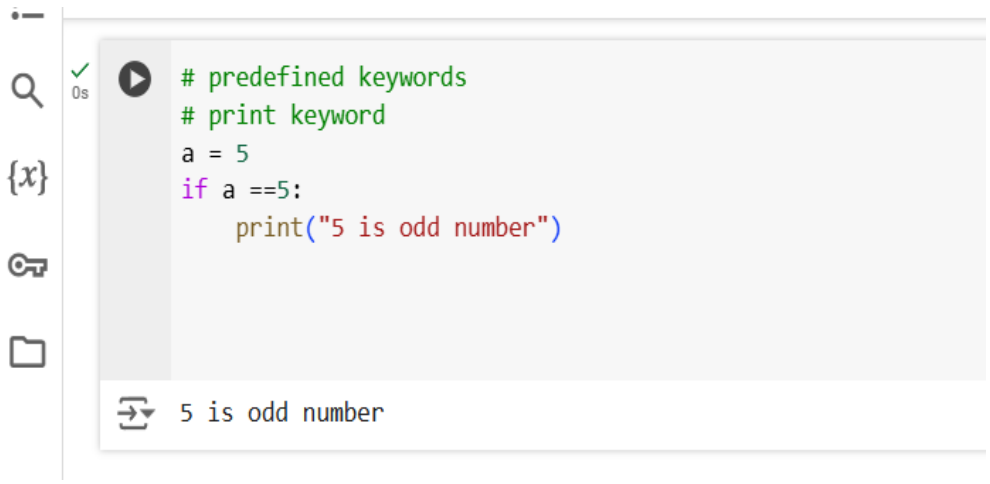
Predefined keywords, as the name state itself those words that are reserve in container and have specific meaning and purpose. These words can be used only if one has known about its role and nature in python.

Example of predefined keywords: - print, input, while, in, etc.

We can neither change the way of writing these keywords nor use it as identifier.

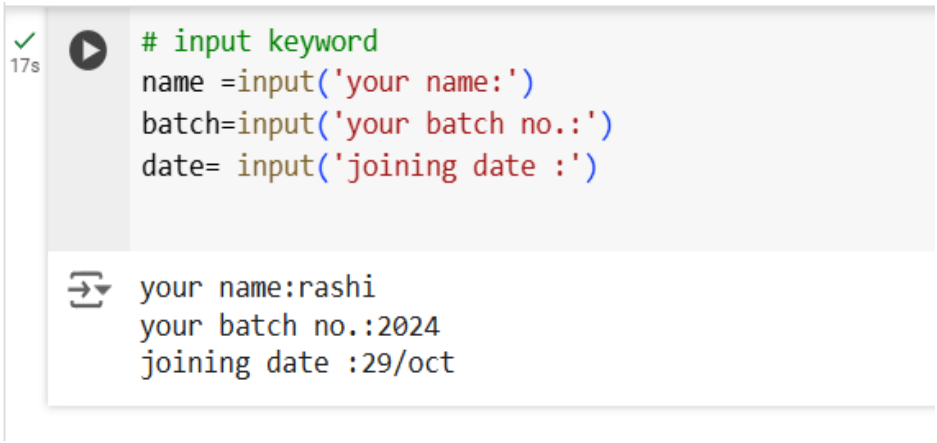
Example: abc= 'Mango' (☑)

while = 'mango'(x)



```
# predefined keywords
# print keyword
a = 5
if a ==5:
    print("5 is odd number")
```

```
5 is odd number
```



```
# input keyword
name =input('your name:')
batch=input('your batch no.:')
date= input('joining date :')
```

```
your name:rashi
your batch no.:2024
joining date :29/oct
```

QUES 3) Compare and contrast mutable and immutable objects in Python with examples.

Object are classified into 2 types

a) Mutable object= In simple term, those object that can be change after creation of it, are consider as Mutable object. If we are capable to change the Integer

into float, Float into string and vice-versa then it will know as Mutable object. Here, list container is an example of this

```
#type casting
#mutable object example
list_cont= [28, 4.89, 65, 'Mango', 'Pineapple'] #i want to change mango with cherry
list_cont[3]= 'cherry'
list_cont
```

```
[28, 4.89, 65, 'cherry', 'Pineapple']
```

```
# Another example
Name= [2+8j, 32, 'Water']
Name[1]= 23
Name
```

```
[(2+8j), 23, 'Water']
```

b) Immutable object= Those objects that cannot be change after the creation of it, are consider as immutable object. Here, string is an example of it.

```
#Immutable object
A = "Deta_analyitics" # i want to change e into a
A[1]= 'a' # it will throw an error as it can't be change because it is immutable object
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-15-7c6f6faa3a04> in <cell line: 3>()
      1 #Immutable object
      2 A = "Deta_analyitics" # i want to change e into a
----> 3 A[1]= 'a' # it will throw an error as it can't be change because it is immutable object

TypeError: 'str' object does not support item assignment
```

Next steps: Explain error

Ques 4) Discuss the different types of operators in Python and provide examples of how they are used.

a) Arithmetic operator: These are basic mathematical operation LIKE addition, subtraction etc.

Addition

```
#Airthmatic operator
a= 5
b= 10
a+b
```

```
15
```

Minus

```
a= 2
b= 5
b-a
```

```
3
```

Multiplication

```
a= 9
b= 2
a*b
```

```
18
```

division

```
# divide
x= 10
y= 5
x/y
```

```
2.0
```

b) Modulous operator: this operator provide remainder after dividing two variables.

```
# modulous operator
a= 31
b= 2
a%b
```

```
1
```

c) Floor operator: this operator provides neared value after division.

```
# floor operator
a= 27
b= 12
a//b
```

```
2
```

d) Comparison operator: This operator is useful to compare two values.
Example :- ==, >=,<=,!=

```
#comparison operator
2==2
```

```
True
```

```
10>12
```

```
False
```

e) Logical operator: This operator used to connect two variable and produce Boolean value as a result. This is sub-divide into AND and OR operator.

AND

```
#logical operator
#and operator
1 and 1
```

```
1
```

```
1 and 0
```

```
0
```

OR

```
# or operator
1 or 1
```

```
1
```

```
1 or 0
```

```
1
```

f) Assignment operator= Assign the value to operator is known as assignment operator

```
#assignment operator
a= 10
a+=5
a
```

```
15
```

```
b= 5
b-=1
b
```

```
4
```

g) Membership operator: this operator helps to check whether the object belongs to that particular block or not.

```
#membership operator
abc= "Mango"
'g' in abc
```

```
True
```

```
xy= "Rainbow"
'r' not in xy
```

```
True
```

h) Identity operator: It helps to compare the location of two object and always gives result in Boolean value.

```
#identify operator
x=9
y=7
x is y
```

```
False
```

i)Binary operator: Operation at bit level i.e., representing the Integer in binary form..

```
[52] #binary operator
     3 & 6
```

```
2
```

```
[53] bin(3)
```

```
'0b11'
```

```
bin(6)
```

```
'0b110'
```

```
[60] # or operator
     3 | 6
```

```
7
```

J) Negotiation operator

```
[61] # negotiation operator
     ~8
```

```
-9
```

K) Bitwise XOR operator: it is return 1 when one operand is 1 .

```
[63] # bitwise Xor operator
     2^4
```

```
6
```

```
[64] # bitwise Xor operator
     bin(4)
```

```
'0b100'
```

```
bin(2)
```

```
'0b10'
```

L) Shift operator:

<u>Left operator</u>

```
# shift operator >> left operator
4<<2
```

```
16
```

```
[71] bin(4)
```

```
'0b100'
```

Right operator

```
[73] # shift operator>> right operator
     12>>1
```

```
6
```

```
bin(12)
```

```
'0b1100'
```

QUES 5) Explain the concept of type casting in Python with examples.

ANS-

Type casting or conversion means changing the data type of value / object. sometimes we see mismatch between 2 operator and for performing the operation, we required to change the type of data.

```
[75] #type casting
     a= '78'
     type(a)
```

```
str
```

```
[76] #convert into int
     int(a)
```

```
78
```

```
[77] # conventing Int into float
     float(a)
```

```
78.0
```

```
#Converting float into str
str(a)
```

```
'78'
```

QUES 6) How do conditional statements work in Python? Illustrate with examples.

ANS-

1) if statement

```
[79] # condition statement
     #if statement
     score = 185
     if score >=100:
         print ('century')
```

```
century
```

```
# dual condition
score = 100
Name= 'Alex'
if ((score>=100)&(Name=='Alex')):
    print("we won ")
```

```
we won
```

2) if else statement: In this, if previous 1st condition is not true then else is printed

```
[81] # if else statement
     marks=86
     if marks>50:
         print("pass")
     else:
         print("fail")
```

```
pass
```

3) if Elif else: This statement used when we want to add multiple statement and print the statement if previous were not true.

```
# if elif else
marks = 76
if 50>marks>33:
    print("grade C")
elif 75>marks>50:
    print("grade B")
else:
    print ("grade A")
```

```
grade A
```

4) nested if else: In this, one can add multiple if else statement in if.

```
[97] # Nested if else
name= input("your name")
phone_no= input("your phn no")
date_of_joining= input("Date of joining ")
if name=="":
    print("your name")
elif len(phone_no)<10:
    print ('invalid number ')
elif '2024' not in date_of_joining:
    print ("give valid year")
else:
    print("successfully registered")
```

```
your namerashi
your phn no4585498654
Date of joining 24/12/2024
successfully registered
```

```
#nested if else
age=20
status= 'Old'
if age>=18:
    if status=='young':
        print("pay 100")
    else:
        print("pay 50")
else:
    print("not eligible ")
```

```
pay 50
```

QUES 7) Describe the different types of loops in Python and their use cases with examples

ANS-

There are 2 types of loops in python

a) while loop - it is executed the code until the condition is met.

```
[102] # while loop
      a= 10
      b= 5
      while b<a:
          print (b)
          b=b+1
```

```
5
6
7
8
9
```

```
# along with else
a= 6
b= 2
while b<a:
    print (b)
    b=b+1
else:
    print('complete')
```

```
2
3
4
5
complete
```

```
[111] # along with break
      a= 10
      b=5
      while b<a:
          print(b)
          b=b+1
          if b==9:
            break
```

```
5
6
7
8
```

b) For loop - it is executed the elements of code in sequence wise.

```
#for loop
a="Moonlight"
for a in 'moonlight':
    print (a)
```

```
m
o
o
n
l
i
g
h
t
```

```
list_c=['lily',45,829,63,"Varnish"]
for a in list_c:
    print (a)
for b in range(0,5):
    print (b)
```

```
lily
45
829
63
Varnish
0
1
2
3
4
```