

today

August 5, 2024

0.1 loops in python

0.1.1 for loop

```
[ ]: # printing numbers
for i in range(0,7):
    print(i)
# iterating through sequences [string,list,tuple,set]
myList = ['khushi','rhydham','hello','world']
for item in myList:
    print(item)
for i in range(0,len(myList)):
    print(myList[i])
# iterating through strings
myString = 'my name is rhydham'
for item in myString:
    print(item)
for i in range(0,len(myString)):
    print(myString[i],end='')
for i in range(0,len(myString)):
    print(f'{myString[i]}\n')
```

```
0
1
2
3
4
5
6
khushi
rhydham
hello
world
khushi
rhydham
hello
world
m
y
```

n
a
m
e

i
s

r
h
y
d
h
a
m

my name is rhydhamm

y

n

a

m

e

i

s

r

h

y

d

h

a

m

0.1.2 for loop with else

```
[ ]: # for-else
for i in range(0,5):
    print(i)
else:#it is executed when for loop ends
    print("for ends here")
```

0
1
2
3
4
for ends here

0.1.3 while loop

```
[ ]: # finite loop
a = 1
while (a<10):
    print(a)
    if a==5:#it will break at 5 and 5 will be printed on screen
        break
    a+=1 #a=a+1 or in C(a++)
```

1
2
3
4
5

```
[ ]: #infinite loop
while True:
    #function da kaam
    print("hello world")

#it runs only when condition is true and here a is not changing therefore it
↳ will be a infinite loop
while (a<10):
    print(a)
```

0.1.4 break statement

```
[ ]: # break statement is used to exit loops at specific condition
a = 0
while True:
    print("hello")
    a+=1
    if a==10:
        break
```

hello
hello
hello
hello
hello
hello
hello
hello
hello
hello
hello

0.1.5 continue function

```
[ ]: c =1
while True:
    if c<4:
        print(c)#it will print 1,2,3
    else:
        c=1
        continue #it will return the program to if condition
    c+=1
# if

# elif
# elif-continue
# elif
# else
# loop if else
```

1
2
3

```
[ ]: c = 48
if c%2==0:
    print("it is divisible by 2")
    if c%6==0:
        print("it is divisible by 6")
```

```
elif c%7==0:
    print("it is divisible by 7")
```

it is divisible by 2

it is divisible by 6

0.1.6 pass statement

```
[ ]: #it is used when you have to define a dummy function
#eat 5 star do nothing
def get_my_name(name):
    pass
# calculator
# sum,sub,pro,div
def sum(a,b):
    pass
```

0.1.7 some practice problems (for loop and while loop)

print table of 5 using for loop

```
[ ]: for i in range(0,51):
    if i%5==0:
        print(i)
for i in range(1,11):
    print(5*i)
```

0
5
10
15
20
25
30
35
40
45
50
5
10
15
20
25
30
35
40
45

use for loop to greet all the users given in the list ['khushi','rhydham','anything','not anything']

[]:

using for loop print basic star pattern

0.2 functions

0.2.1 basic functions

0.2.2 recursions

0.2.3 functions with default parameter value

0.2.4 practice problems regarding function and recursion

0.3 random module revision

0.3.1 random.randint function

0.3.2 random.choice function

0.4 stone paper scissor project

0.5 file I/O

0.5.1 text file

reading the file

writing the file

- 0.5.2 other
- 0.6 classes (OOPs)
 - 0.6.1 objects and all
 - 0.6.2 self parameter
 - 0.6.3 static method
 - 0.6.4 constructor
 - 0.6.5 inheritance
- 0.7 Datetime module
- 0.8 guessing the number game
- 0.9 advanced python
 - 0.9.1 walrus operator
 - 0.9.2 type declaration
 - 0.9.3 match case (switch statemnt in cpp/c)
 - 0.9.4 try except
 - 0.9.5 main function in python