21or53n4j

January 23, 2025

[ ]:

```
#loop
for i in range (0,10): #for loop
  print("Hello W0rld")
```

```
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
Hello W0rld
```

[ ]:

```
word="python"
for i in word:
  print(i)
```

```
p
y
t
h
o
n
```

[ ]:

```
#multiplication table
def mult(x):
  for i in range(1,11):
    print(f"{x}*{i}={x*i}")

a=int(input("Enter the number"))
mult(a)
```

```
Enter the number2
2*1=2
2*2=4
2*3=6
```

```
2*4=8
2*5=10
2*6=12
2*7=14
2*8=16
2*9=18
2*10=20
```

[ ]:
```python
i=0
while i<6: #while loop
  print(i+1)
  i+=1
```

```
1
2
3
4
5
6
```

[ ]:
```python
names=["raman",'santosh','hari','gopal']
for name in names:
  print(name)
  for letter in name:
    print(letter)
```

```
raman
r
a
m
a
n
hari
h
a
r
i
gopal
g
o
p
a
l
```

[ ]:
```python
def sum():
  x=0
  y=0
  for i in range(0,101):
    if i%2==0:
      x=x+i
    else:
      y=y+i
  return (x,y)
sum_even,sum_odd=sum()
print("Sum of even till 100",sum_even)
print("Sum of odd till 100",sum_odd)
```

```
Sum of even till 100 2550
Sum of odd till 100 2500
```

[ ]:

```
def primes():
  lst=[]
  for i in range(1,101):
      for j in range(2,int(i/2)+1):
        if i%j==0:
          break
      else:
        lst.append(i)
  return lst
def print_primes():
  for prime in primes():
    print(prime)
print_primes()
```

```
1
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
```

```
79
83
89
97
```

[ ]:

```
lst=['banana', 'orange', 'mango', 'lemon'] #reversing a list using positive indexing
for i in range(len(lst)-1,-1,-1):
  print(lst[i])

lst=['banana', 'orange' , 'mango', 'lemon'] #reversing a list using negative indexing
revlst=[]
for i in range(-1,-len(lst)-1,-1):
  revlst.append(lst[i])
print(revlst)
```

```
['lemon', 'mango', 'orange', 'banana']
```

[ ]:

```
lst
lst.reverse()
lst
```

[ ]:

```
['banana', 'orange', 'mango', 'lemon']
```

[ ]:

```
#args=list/tuples, kwargs=dictionary
def func(*args): #using args *
  sum=0
  for x in args:
    sum=sum+x #sum+=x
  return sum

print("4 values sum",func(1,2,3,4))
print("5 values sum",func(1,2,3,4,5))
print("7 values sum",func(1,2,3,4,5,6,7))
```

```
4 values sum 10
5 values sum 15
7 values sum 28
```

[ ]:

```
#kwargs --for dictionary
def func1(**kwargs): #using kwargs **
  for key,value in kwargs.items():
    print(f"{key} is {value}")

func1(name='raman',no=1)
func1(name='santosh',no=2)
func1(name='hari',no=3)
```

```
name is raman
no is 1
name is santosh
no is 2
name is hari
no is 3
```

[ ]:

```python
#fibonacci using recursion
def recursion(n):
  if n==0:
    return 0
  elif n==1:
    return 1
  else:
    return recursion(n-1)+recursion(n-2)

def fiboser(x):
  lst=[]
  for i in range(0,x+1):
    lst.append(recursion(i))
  return lst

print(fiboser(10))
```

```
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
```