Problem Solving in Python

12 June 2019

Day Objectives

- String Slicing
- Functions
- Functions in Python
- Basic problems related to conditional statement using functions
- Iteration in Python

String Slicing

In [ ]:
```
s1="PythonProgram"
s1[0] ##returns the first character
s1[-1] # returns the last character
s1[0:len(s1)] # retunrs whole string
s1[0:] # retunrs whole string
s1[:2] # returns first two character
s1[:-2] # leaves last two characters
s1[-2:] # returns last two characters
s1[-1::-1]  ##reverse string
s1[-1:-3:-1] ## reverse last two characters
s1[len(s1)//2:len(s1)//2-2:-1] ## reverse the middle characters
s1[::2] # accessing alternate characters
s1[::-2] # accessing alternate characters in reverse
```

Functions

In [ ]:
```
# function to reverse a string
def reverseString(s):
    return s[-1::-1] # [::-1] also gives the reverse string
s=input("Enter a String")
reverseString(s)
```

In [ ]:
```
# function to check if a string is a palindrome

def palindrome(s):
    if s==s[::-1]:
        return True
    else:
        return False
s=input("Enter a String")
palindrome(s)  # "racecar" is a palindrome
```

In [ ]:
```
# function to check if a given year is a leap year
def isLeapyear(year):
    if year%400==0 or (year%100!=0 and year%4==0):
        return True
    return False
year=int(input("Enter a year"))
isLeapyear(year)
```

In [ ]:
```
# function to count the number of  digits in a given number
def digitcount(num):
    return len(num)
num=input("Enter a number")
digitcount(num)
```

In [ ]:
```
# function to identify the greatest of 4 numbers
def greatest(n1,n2,n3,n4):
    if n1>n2 and n1>n3 and n1>n4:
        return n1
    elif n2>n3 and n2>n4:
        return n2
    elif n3>n4:
        return n3
    else:
        return n4
n1=int(input("enter first number"))
n2=int(input("enter second number"))
n3=int(input("enter third number"))
n4=int(input("enter last number"))
greatest(n1,n2,n3,n4)
```

Iteration

- for
- while

In [ ]:
```
# function to print n natural numbers
def n_numbers(n):
    for counter in range(1,n+1):
        print(counter,end=" ")
    print()
    return
n=int(input("enter a range"))
n_numbers(n) # in python we have to print the function--print(n_numbers(n))
n_numbers(n+10)
```

In [ ]:
```
# function to print n natural numbers using while loop
def n_numbers(n):
    i=1
    while i<=n:
        print(i,end=" ")
        i=i+1
    return
n=int(input("enter a range"))
n_numbers(n)
```

In [ ]:
```
# function to print all numbers divisible by 6 and not a factor of  100 in a give
def div6_fact100(lb,ub):
    for i in range(lb,ub+1):
            if i%6==0 and 100%i!=0:
                print(i,end=" ")
lb=int(input("enter lower bound"))
ub=int(input("enter upper bound"))
div6_fact100(lb,ub)
```

In [ ]:
```
# function to find average of cubes of all even numbers in a given range(lb,ub) 
def cubesAvg(lb,ub):
    tot=0
    count=0
    for i in range(lb,ub+1):
        if i%2==0:
            tot=tot+(i**3);
            count=count+1;
    print(tot/count)
lb=int(input("enter lower bound"))
ub=int(input("enter upper bound"))
cubesAvg(lb,ub)
```

In [ ]:
```
# function to generate the list of factors for a given number
def factorOf_N(num):
    for i in range(1,num+1):
        if num%i==0:
            print(i,end=" ")
num=int(input("enter a number"))
factorOf_N(num)
```

In [ ]:
```
# function to print factorial of a given number
def factorial(n):
    sum=1
    for i in range(1,n+1):
        sum=sum*i
    print(sum)
n=int(input("enter a number"))
factorial(n)
```

In [ ]:
```
# function to check if a given number is prime or not
def prime(n):
    sum=0
    for i in range(2,n):
        if n%i==0:
            sum=sum+1;
    if sum==0:
        return True
    else:
        return False
n=int(input("enter a number"))
prime(n)
```

In [5]:
```
# function to calcute the average of prime numbers in a given range
def avgPrime(ran):
    tot=0
    count=0
    for i in range(2,ran):
        sum=0
        for j in range(2,i):
            if i%j==0:
                sum=sum+1
        if sum==0:
            tot=tot+i
            print(i)
            count=count+1
    return tot/count
ran=int(input("enter a range"))
avgPrime(ran)
```

```
enter a range10
2
3
5
7
```

Out[5]: 4.25

In [6]:
```
# function to print perfect numbers in a given range
def perfectNumber(lb,ub):
    for i in range(lb,ub):
        sum=0
        for j in range(1,i):
            if i%j==0:
                sum=sum+j
        if sum==i:
            print(i,end=" ")
lb=int(input("enter lower bound"))
ub=int(input("enter upper bound"))
perfectNumber(lb,ub)
```

```
enter lower bound1
enter upper bound1000
6 28 496
```