In [ ]:

```
#functions have a name
#functions are passed values using parameter
#functions can return a result
```

In [ ]:

```
def my_average(a,b):
    c=(a+b)/2
    return (c)


nFirst=int(input('First number :'))
nSecond=int(input('Second number :'))
nTotal=my_average(nFirst, nSecond)
print(nTotal)
```

In [ ]:

```
def subtract(a,b):
    c=a-b
    return (c)

nFirst=int(input('First number :'))
nSecond=int(input('Second number :'))

nTotal=subtract(nFirst, nSecond)
print(nTotal)
```

In [ ]:

```
def myAdd(a,b,c,d,e):
    return (a+b+c+d+e)

nTotal=myAdd(1,2,3,4,5)
print(nTotal)
```

In [ ]:

```
#comment ....
def my_average(a,b,c,d,e)
    return (a+b+c+d+e/5.0)

#input

#processing
nAve=my_average(1,2,3,4,5)
#output
print(nAve)
```

In [ ]:

```
#comment ....
def my_number(nNum):
    if nNum==1:
        return('one')
    elif nNum==2:
        return('two')
    else:
        return('sorry, not defined')

#input
nNum=int(input('Please enter a number : '))
#processing

#output
print(my_number(nNum))
```

In [ ]:

```
#define the function using default value parameter
def the_sum(a,b=20):
    return (a+b)

#input
myNum=int(input("please enter a number : "))

nTotal=the_sum(myNum)
print(nTotal)
nTotal=the_sum(myNum, myNum)
print(nTotal)
```

In [ ]:

```
#global and local variables
def my_add():
    global x
    print('The value of x in function is : ', x)
    x=x+5
    print('The value of x in function is : ', x)
    return
x=0
my_add()
print('The value of x out of function is : ', x)
```

In [ ]:

```
#global and local variables
def my_add(x):
    #global x
    print('The value of x in function is : ', x)
    x=x+5
    print('The value of x in function is : ', x)
    return
x=0
my_add(x)
print('The value of x out of function is : ', x)
```

In [ ]:

```
#lamda functions
def f(y):
    return (y+2)*3

x=int(input('input a value : '))
print(f(x))
```

In [ ]:

```
#using functions to format input
def my_format(s1, s2):
    fullName = s1 + ' ' + s2
    #fullName = s1.title() + ' ' + s2.title()
    return fullName
#input
firstname=input('input your first name : ')
surname = input('input your surname : ')
#processing
myName=my_format(firstname, surname)
#output
print(myName)
```

In [ ]:

```
def evenval(x):
    return (x%2 ==0)

#input
#nStart=int(input('Enter a start number : '))
#nEnd=int(input('Enter an end number: '))
myEvens=filter(evenval, range(1,10))
print(list(myEvens))
```

In [ ]:

```
def square(nNum):
    return (nNum*nNum)
#input
#nStart=int(input('Enter a start number : '))
#nEnd=int(input('Enter an end number: '))
#process
mySquare=map(square, range(1,11))
#output
print(list(mySquare))
```

In [ ]:

```
#note that functools is a module that is imported
import functools
def myAdd(x, y):
    return (x+y)
mySum=functools.reduce(myAdd, range(1,5))
print(mySum)
```

In [ ]:

```
#define the function attributes
#change the value of the function attributes ....
```

In [ ]:

```
#you can implement recursion using a function
#Write a program using recursion that adds 10 numbers
```

In [ ]:

```
#Write a program to demonstrate the use of an iterator and a generator
```

In [ ]:

```
#there are a number of ways of importing modules .... NB
from calendar import *
year=int(input('The year is : '))
prcal(year)
```

In [ ]:

```
#import math
```

In [ ]:

```
def my_pet(petType,petName):
    print('I have a %s named %s.' %(petType.lower(), petName.title()))

petType = str(input("What type of pet do you have? "))
petName = str(input("What is the name of your pet? "))
my_pet(petType, petName)
```

Function

Defines a function called myDate. The program must prompt the user for the input to the function, namely is three integers (nDay, nMonth, nYear). The output from the function must be a string message describing your birthday. The function call theMessage = myDate (nDay, nMonth, nYear), must be following by a print(theMessage).

In [ ]:

```
def my_date(nDay, nMonth, nYear):
    sMonth = 'XXX'
    months = ['January','February','March','April','May','June','July','August','September','October','November','December']
    if 1 <= nMonth <= 12:
        print ("The month is", months[nMonth-1])
        sMonth = months[nMonth-1]
    else:
        print ("Value is out of the range")
    return ('You were born on %d %s %d' %(nDay, sMonth, nYear))

nDay   = int(input("On which day were you born? "))
nMonth = int(input("In which month were you born? "))
nYear  = int(input("In which year were you born? "))

theMessage = my_date(nDay, nMonth, nYear)
print(theMessage)
```

In [ ]:

In [ ]: