Activity 1:

Write a function which prints out even numbers up to the input number. The result is printed down the page.

def func(num):
 for x in range(...):
 if .... == .... :
 print (....)
func(6) # This makes
2
4
6

func(10) #This makes
2
4
6
8
10

Activity 2:

Write a function which prints out odd numbers up to the input number. The result is printed down the page.

def func(num):
 for x in range(...):
 if .... == .... :
 print (....)
func(6) # This makes
1
3
5

func(10) #This makes
1
3
5
7
9

Activity 3:

Write a function which prints out multiples of five up to the input number. The result is printed down the page.

def func(num):
 for x in range(...):
 if .... == .... :
 print (....)

func(13) # This makes
5
10
func(26) #This makes 5
10
15
20
25

Activity 4:

Write a function which prints out multiples of 10 up to the input number. The result is printed down the page.

def func(num):
 for x in range(...):
 if .... == .... :
 print (....)

func(25) # This makes
10
20
func(55) #This makes
10
20
30
40
50

Activity 5:

Write a function which outputs even numbers up to the input number. The str() function turns an integer into a String. The result is printed on one line separated by commas.

def func(num):

 value = ‘’

 for x in range(...):

 if .... == .... :

 value += str(x) + ‘, ‘

 return value[…] # You need to remove the last comma and the space

print(func(6) ) # This makes 2, 4, 6 (This will print them across the page)

print(func(10)) #This makes 2, 4, 6, 8, 10 (This will print them across the page)

Activity 6:

Write a function which outputs odd numbers up to the input number. The result is printed on one line separated by commas.

def func(num):

 value = ‘’

 for x in range(...):

 if .... == .... :

 value += str(x) + ‘, ‘

 return value[…]# You need to remove the last comma and the space

print(func(6)) # This makes 1, 3, 5 (This will print them across the page)

print(func(10)) #This makes 1, 3, 5, 7, 9 (This will print them across the page)

Activity 7:

Write a function which outputs multiples of five up to the input number. The result is printed on one line separated by commas.

def func(num):

 value = ‘’

 for x in range(...):

 if .... == .... :

 value += str(x) + ‘, ‘

 return value[…]# You need to remove the last comma and the space

print(func(13)) # This makes 5, 10 (This will print them across the page)

print(func(26)) #This makes 5, 10, 15, 20, 25 (This will print them across the page)

Activity 8:

Write a function which outputs multiples of 10 up to the input number. The result is printed on one line separated by commas.

def func(num):

 value = ‘’

 for x in range(...):

 if .... == .... :

 value += str(x) + ‘, ‘

 return value[…]# You need to remove the last comma and the space

print(func(25)) # This makes 10, 20 (This will print them across the page)

print(func(55)) #This makes 10, 20, 30, 40, 50 (This will print them across the page)

Activity 9:

Write a function which outputs a word the number of times that is required, as set by the second parameter.

def func(text, num):

 value = ‘’

 for x in range(...):

 value += ...

 return value

print(func(‘Hi’, 3)) # This makes Hi Hi Hi

print(func(‘John’,5)) #This makes John John John John John

Activity 10:

Write a function which outputs only the starting part of the word (defined by the first number) the correct number of times (defined by the second number).

def func(text, num1, num2):

 value = ‘’

 for x in range(...):

 value += ....

 return value

print(func(“chocolate”, 2, 3)) # This makes ch ch ch

print(func(“chocolate”, 3, 2)) #This makes cho cho

Activity 11:

Write a function which outputs the first letter of the word, the first two letters and so on for as many times as the number requires. *Hint you can use variables in slicing eg text[x:]*

def func(text, num):

 value = ‘’

 for x in range(...):

 value += ...

 return value

print(func(‘chocolate’, 3)) # This makes c ch cho

print(func(‘chocolate’, 6)) #This makes c ch cho choc choco chocol

Activity 12:

Write a function which counts the number of times that a particular number is found in a list of numbers.

def func(list1, num):

 count= 0

 for x in ... :

 if ... == ... :

 count+=1

 return count

print(func([1,2,3,4,5,4,3], 3)) # This makes 2

print(func([1,3,5,7,9],2)) #This makes 0

Activity 13:

Write a function which checks if a number is in a list of numbers. It will return True or False.

def func(list1, num):

 value = False

 for x in ... :

 if ... == ... :

 value =True

 return value

print(func([1,3,5,7,9], 3)) # This makes True

print(func([1,3,5,7,9], 6)) #This makes False

Activity 14:

Write a function which checks if a number is in a list of numbers Version 2. It will return True or False. This is a better way to construct the function.

def func(list1, num):

 for x in ... :#You could also do this without this line

 if ... == ... :

 return True

 return False

print(func([1,3,5,7,9], 3)) # This makes True

print(func([1,3,5,7,9], 6)) #This makes False

*This works because as soon as the function hits the return it exits the function.*

**NB there is also way to do this without using a loop, look back in the notes of this lesson for a clue**.Activity 15:

Write a function called bigger which takes two numbers and returns the larger of the two numbers

def bigger(num1, num2):

 ...

print(bigger(3,5])) # This makes 3

print(bigger(5,3)) #This makes 3

Activity 16:

Write a function called biggest which returns the largest number of three numbers

def biggest(num1, num2, num3):

 ...

print(biggest(3,4,5)) # This makes 5

print(biggest(5,3,4)) # This makes 5

print(biggest(4,3,5)) # This makes 5

Activity 17:

Write a function called middle which returns the number in the middle if the numbers are ranked in order. If there are two numbers the same it must return the smaller ranked number, never the largest number.

def middle(num1, num2, num3):

 ...

print(middle(1,4,5)) # This makes 4

print(middle(7,7,9)) # This makes 7

print(middle(4,3,5)) # This makes 4