**QB Python Loops**

1. What is a loop? What are the looping constructs available in Python?
2. What is the main difference between a for loop and a while loop?
3. When does the else clause of a loop get executed? Is else clause mandatory in a loop?
4. What is the use of break and continue statements in a loop? Give a suitable example to explain.
5. What is an infinite loop? Give an example of an infinite loop.
6. What is meant by nesting of loops? Give a suitable example to explain.
7. Each of the following code segments is written to perform the job specified in the respective comment. Check each of these code segments and make the necessary changes to correct the code:
   1. #To display "Hello" 10 times for i in range(1, 10):

print("Hello")

* 1. #To print first 10 natural numbers i=1

while (i==10): print("i") i+=1

* 1. #To display first 10 even natural numbers num=1

while (num<=10): print(num)

num+=1

* 1. #To display first 10 odd natural numbers for i in range (10):

print(i)

* 1. #To calculate DA as 10% of basic salary for 10 employees basic=input("Enter basic salary")

da=10\*basic print("DA =",da)

1. Find syntax error(s), if any, in each of the following code segments:
   1. a=int(input("Enter a number: ")) c==1

for b in range (a,a\*10.0,a) c=\*b

print(a,b,c)

* 1. a=int(input("Enter a number: ")) b=int(input("Enter another number: ")) start,end=a,b

if a>b:

start,end=b,a i=start

while i=<end

if (i%start=0): print(i,end='--',start)

i+=1

print(i)

* 1. a=int(input("Enter a number: ") b=int(input("Enter another number: ") if a>b: step=-1

else: step=1 while a!=b:

if (a+b%2>0):

print(a+b,sep="--') a=+step

* 1. a=int(input("Enter a number: ") while a%5!=0

if (a%2>0):

print(a,sep="--)

a+=1

else: print(a)

(v) x=123045

while x%10:

x=//10 print(x,sep="--") else print(a)

(vi) x=123045 while x%10:

d=x%10

for i in range(d): print(i)

else: print (d) print(x,sep='--') x/=10

else print(a)

1. Find output generated by each of the following scripts:
   1. a=4 c=0

for b in range (a,a\*a,a): c+=b

print(a,b,c)

* 1. #Find output for (i)a,b=4,10 (ii) a,b=2,-4 a=int(input("Enter a number: ")) b=int(input("Enter another number: ")) start,end=a,b

if a>b:

start,end=b,a i=start

while i<=end:

if (i%start==0): print(i,end='--')

i+=1

print(i)

* 1. a=10 b=a-6 if a>b:

step=-1 else: step=1 while a!=b:

if (a+b%2>0):

print(a+b,sep="--") a+=step

* 1. #Find output for (i)a=10 (ii) a=-6 a=int(input("Enter a number: ")) while a%5!=0:

if (a%2>0):

print(a,sep="--")

a+=1

else: print(a)

(v) x=123045

while x%10:

x//=10 print(x,sep="--")

else: print(x)

(vi) x=123045 while x%10:

d=x%10

for i in range(d): print(i)

else: print (d) print(x,sep='--') x//=10

1. Write a program to display first n natural numbers, where n is to be input from the user.
2. display even natural numbers from 2 to n, where n is to be input from the user. For example, if the value of n is 13, the output should be 2 4 6 8 10 12.
3. Write a program to find the sum of first n natural numbers, where n is to be input from the user.
4. To find the sum of first n even natural numbers, where n is to be input from the user.
5. To input an integer and find its factorial. Factorial of an integer n (n>=1), is defined as the product of all natural numbers from 1 to n, and is represented by n!. For example, factorial of 5 is represented by 5! and is equal to 120 (=1x2x3x4x5).

Factorial of 0 is defined as 1 and factorial of negative integers is not defined.

1. script to input 10 numbers and then display the largest of the numbers entered.
2. Write a program to input 10 numbers and then display their sum and average. Also display the largest and the smallest of the numbers entered.
3. input a number and display its first 10 multiples.
4. Write a program to input two numbers ***m*** and ***n***. Then display first ***m*** multiples of ***n***.
5. Write a program to input a number. If the number is negative, then input the number again. Keep on doing so until the user enters a positive number or zero.
6. script to input 10 numbers and then display the smallest number entered. If the smallest number is an integer, then find the sum of its digits, otherwise display the message “smallest number is not an integer”.
7. Write a program to input two numbers and find their LCM and HCF.
8. Write a program to input a number and check whether it is a prime or not.
9. Write a program to display all the prime numbers between ***m*** and ***n***, where ***m*** and ***n*** have to be input from the user.
10. Write a program to input a list of n number and count how many of the entered numbers were prime.
11. input an integer and find the sum of its digits.
12. input an integer and find the product of its odd digits. If the number does not contain any odd digit, then the program should display an appropriate message instead of the product.
13. input 10 integers and find the sum of the digits of each integer.
14. Armstrong number is an integer which is equal to the sum of the cubes of its digits. For example 153 is an Armstrong number because 153 = 13+53+33.

Write script to input an integer and check whether it is an Armstrong number or not.

1. Write a program to input a number and check whether it is palindrome or not.
2. Write a program to input a number and check whether it is a perfect number or not. (A number is a perfect number if it is equal to the sum of its own factors, except itself. For example, 6 is a perfect number as 6=1+2+3, and 8 is not a perfect number as 8 != 1+2+4
3. Write script to display first n terms of Fibonacci series. The Fibonacci series is: 0 1 1 2 3 5 8 13 . . .

First and Second terms of the Fibonacci series are 0 and 1 resepectively, and after that any term is the sum of its previous two terms.

1. Write a program to input a number and check whether it is a Fibonacci number or not. A number is a Fibonacci number if it is a term in the Fibonacci series.
2. Write a menu driven program to calculate the total surface area and volume of a cube, cuboid, or sphere depending upon user’s choice. The program should continue until the user selects the option to exit the program.
3. Write a program to find the sum of first ***n*** terms of the following series without using inbuilt function pow() and math.factorial():

(i) x + x2 + x3 + . . .

(ii) x - x2 + x3 - . . .

*x*2

1. 1+ x +

# 2

*x*2

1. 1+ x +

# 2!

*x*2

*x*3

+ + . . .

# 3

*x*3

+ + . . .

# 3!

*x*3

1. 1- x +

-

# 2! 3!

+ . . .

*x*3

1. 1+ +

# 3!

*x*5 *x*7

+

# 5! 7!

. . .

*x*3

1. 1- +

# 3!

*x*5 *x*7

-

# 5! 7!

. . .

1. Write a program to generate n lines of the following pattern on the computer screen:
   1. 1 12 123

.

.

* 1. 1

12

123

.

.

* 1. 1

121

12321

1234321

.

.

* 1. \*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

### .

**.**

* 1. \*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

.

.

1. Write a program to generate 2n+1 lines of the following pattern on the computer screen:
   1. \*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

* 1. \* @@@

\*\*\*\*\*

@@@@@@@

\*\*\*\*\* @@@

\*

* 1. \*

\* \*

\* \*

\* \*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

**ANSWERS**

* 1. \*

\* \*

\* \*

\* \*

\* \*

\* \*

\*

* + 1. Loop is repetition of a set of statements.

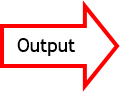
There are 2 looping constructs available in Python. These are:

while and for

* + 1. for loop iterates over the elements of an iterable whereas a while loop iterates while a specified condition is true.
    2. else clause of a loop gets executed after the normal completion of the loop. else clause is not mandatory in a loop.
    3. The *break* statement prematurely ends execution of the current *while or for* loop. It brings the control to the statement immediately following the current control structure, skipping the optional "else" clause if the loop has one.

The *continue* statement is used to skip the execution of the current iteration of a loop, and continue with the next. *continue* does not terminate the loop, but continues with the next iteration of the loop. **Example:**

#break and continue for i in range(1,11):

if (i%3 == 0):

continue; if (i\*3 > 25):



break; print(i,end=' ')

else: print("Else part of loop") print()

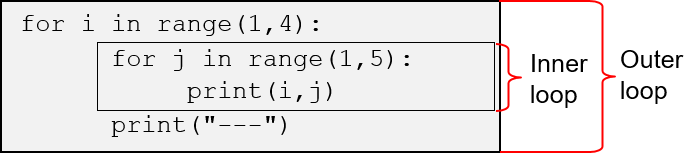
print(i);

* + 1. A loop which never terminates is called an infinite loop. Example:

i=1

while i<5:

print(i)

* + 1. Nesting of loops means creating a loop within the body within the body of another loop. The contained loop is called the *nested* loop or *inner* loop and the container loop is called the *outer loop*. Example::

7.

* + - 1. #display "Hello" 10 times

for i in range(1, ~~10~~**11**): OR range(~~1,~~ **10**): print("Hello")

* + - 1. #To print first 10 natural numbers i=1

while (i**<=**10):

print(**i**) i+=1

* + - 1. #To display first 10 even natural numbers num=**2**

while (num<=**20**):

print(num)

**num+=2 #Indentation**

* + - 1. #To display first 10 odd natural numbers for i in range (10):

print(**2\*i+1**)

* + - 1. #To calculate DA as 10% of basic salary for 10 employees

## for i in range(10):

basic=input("Enter basic salary") da=10\*basic

print("DA =",da)

8.

* + - * 1. a=int(input("Enter a number: ")) c==1

for b in range (a,a\*10.0,a)**:** c**\*=**b

print(a,b,c)

* + - * 1. a=int(input("Enter a number: ")) b=int(input("Enter another number: ")) start,end=a,b

if a>b:

start,end=b,a i=start

while i**<=**end**:**

if (i%start**==**0): print(i,start**,end='--'**)

i+=1

print(i)

* + - * 1. a=int(input("Enter a number: ")**)** b=int(input("Enter another number: ")**)** if a>b: step=-1

else: step=1 while a!=b:

if (a+b%2>0):

print(a+b,sep="--**"**) a=+step

* + - * 1. a=int(input("Enter a number: ")**)** while a%5!=0**:**

if (a%2>0):

print(a,sep="--**"**)

a+=1

else: print(a)

(v) x=123045

while x%10:

x**//=**10 print(x,sep="--")

else**:** print(a) #Indentation

(vi) x=123045 while x%10:

d=x%10

for i in range(d): print(i)

else: print (d) #Indentation print(x,sep='--')

x/=10 else**:** print(a)

9.

(i) 4 4 4 4 8 12 4 12 24

(ii) 4--8--11 # for a=4, b=10

-4--0--3 # for a=2, b=-4

1. 14

13

12

11

10

9

1. 10 # for a=10

-5 # for a=-6

(v) 12304

1230

1230

(vi) 0 1 2 3 4 5

123045

0

1

2

3

4

12304

1. # To display first n natural numbers

n = int(input("Enter the value of n: ")) for i in range(1,n+1):

print(i, end=' ')

1. # To display even natural numbers from 2 to n n = int(input("Enter the value of n: "))

for num in range(2, n+1, 2): print(num)

1. # To find the sum of first n natural numbers n = int(input("Enter the value of n: ")) sum=0

for i in range(1,n+1): sum += i

print("sum of first",n,"natural numbers =",sum)

1. # To find the sum of first n even natural numbers n = int(input("Enter the value of n: "))

sum=0

for num in range(2, 2\*n+1, 2): sum += num

print("sum of first",n,"even natural numbers =",sum)

### OR

# To find the sum of first n natural numbers n = int(input("Enter the value of n: ")) sum=0

for num in range(1, n+1): sum += 2\*num

print("sum of first",n,"even natural numbers =",sum)

1. # To find the factorial of an integer

n = eval(input("Enter the value of n: "))

n=abs(int(n)) #because factorial of negative numbers is not defined factorial=1

for num in range(2, n+1): #multiplication with 1 is redundant factorial \*= num

print(factorial)

1. # To find the largest of 10 numbers entered n = eval(input("Enter a number: "))

max=n

for i in range(1,10): n=eval(input("Enter a number: ")) if n>max:

max=n

print(max,"is the largest number entred")

1. sum=0

n=Max=Min=eval(input("Enter first number: ")) sum+=n

for i in range(9):

n=eval(input("Enter next number: ")) sum+=n

if n>Max:

Max=n elif n<Min:

Min=n avg=sum/10 print("Sum =",sum)

print("Average =",avg)

print("Largest number entered =",Max) print("Smallest number entered =",Min)

1. # To Input a number and display its first 10 multiples n = eval(input("Enter a number: "))

for i in range(1,11): print(n,"x",i,"=",n\*i)

1. # To display first m multiples of n m=int(input("Enter the value of m: ")) n=eval(input("Enter the value of n: ")) print("First",m,"multiples of",n,"are: ") for i in range (1,m+1):

print(n,"x",i,"=",n\*i)

1. while(True):

n=float(input("Enter a positive number: ")) if n>=0:

break

1. #To find:

# (i) the smallest of 10 numbers entered

# (ii) Sum of the digits of the smallest number, if it is an integer

n = eval(input("Enter a number: ")) min=n

for i in range(1,10): n=eval(input("Enter a number: ")) if n<min:

min=n

print(min,"is the smallest number entred") if (type(min)==int):

min=abs(min) # to make the number positive, if it is negative sum=0

while min != 0: digit = min%10 sum += digit min//=10

print ("Sum of the digits of smallest number",sum) else: print("Smallest number is not an integer")

1. n1=int(input("Enter first number: ")) n2=int(input("Enter second number: "))

#LCM is the smallest multiple of n1 which is a multiple of n2 also i=1

while (i\*n1%n2 != 0): i+=1

LCM=i\*n1 HCF=(n1\*n2)//LCM print("LCM =",LCM) print("HCF =",HCF)

1. # To input a number and check whether it is prime or not n = eval(input("Enter a number: "))

n = abs(int(n)) count, factor=0,1 factor=1

while factor<=n:

if (n%factor==0): count+=1

factor+=1 if count==2:

print(n,"is a prime number") else:

print(n,"is not a prime number")

### OR

# To input a number and check whether it is prime or not n = eval(input("Enter a number: "))

n = abs(int(n)) count=0 factor=2

while factor<=n\*\*0.5: if (n%factor==0):

count+=1 break

factor+=1 if (count==0):

print(n,"is a prime number") else:

print(n,"is not a prime number")

1. m=int(input("Enter the value of m: ")) n=int(input("Enter the value of n: ")) if m<1:

m=1 #We don't check any number less than 2 for prime for k in range(m+1,n):

high=int(k\*\*0.5)

for divisor in range(2,high+1): if k%divisor==0:

break

else: print(k,end=" ")

1. # To input a list of n number and

# count how many of the entered numbers were prime n = eval(input("Enter the value of n: "))

n = abs(int(n)) count=0

for i in range(1,n+1):

num = eval(input("Enter a number: ")) num = int(abs(num))

prime=1 factor=2

while factor<num and prime==1: if num%factor==0:

prime=0 else: factor+=1;

if num>1 and prime==1: count+=1

print(count,"number(s) was/were prime")

1. #To find the sum of the digits of an integer n = int(input("Enter an integer: "))

sum=0

while n != 0:

digit = n%10 sum += digit n//=10

print (sum)

1. # To input an integer and find the product of its odd digits. n = int(input("Enter a number: "))

n = abs(int(n)) prod, Found=1, False while n != 0:

digit = n%10

if digit%2 == 1: prod \*= digit Found=True

n //= 10

if Found: # OR Found==True print("Product of odd digits =",prod)

else:

print("No odd digit is present in the number")

1. #To find the sum of the digits of an integer for i in range (10):

n = int(input("Enter an integer: ")) n=abs(n)

sum=0

while n != 0:

digit = n%10 sum += digit n//=10

print (sum)

1. #To input a number and check whether it is an Armstrong number. n=eval(input("Enter an integer: "))

n=abs(int(n)) sum, temp=0, n while n:

d=n%10

sum += d\*\*3 n//=10

if temp==sum:

print("Armstrong number")

else: print("Not an Armstrong number")

1. n=int(input("Enter the number: ")) temp=n

rev=0 while n>0:

d=n%10 rev=rev\*10+d n//=10

if rev==temp:

print(n,"is a Palindromic number")

else: print(n,"is not a Palindromic number")

1. n=int(input("Enter the number: ")) sum\_of\_factors=0

for i in range(1,n//2+1): if n%i==0:

sum\_of\_factors+=i if n==sum\_of\_factors:

print(n,"is a perfect number") else: print(n,"is not a perfect number")

1. #To display first n terms of the Fibonacci series: # 0 1 1 2 3 5 8 13 . . .

n = eval(input("Enter the number of terms wanted: ")) n=abs(int(n))

T1,T2=0,1

if (n==1):

print(T1, end=' ') if (n>=2):

print(T1, T2, end=' ') for i in range(3,n+1):

T3=T1+T2

print(T3, end=' ') T1, T2 = T2, T3

1. n=int(input("Enter the number: ")) if n==0 or n==1:

print(n,"is a Fibonacci number")

else:

t1,t2=0,1

while n>t1+t2: t1,t2=t2,t1+t2

if n==t1+t2:

print(n,"is a Fibonacci number") else: print(n,"is not a Fibonacci number")

1. #To calculate total surface are and volume of a

#cube, cuboid, or sphere depending upon the user's choice. menu='''

* 1. Cube
  2. Cuboid
  3. Sphere

0. Exit'''

Pi=3.14

while (True):

print(menu)

choice=eval(input("Enter your choice (1, 2, 3, or 0): ")) if (choice==1):

s=eval(input("Enter side of cube: ")) TSA=6\*s\*s

Vol=s\*s\*s

print("Total Surface Area =",round(TSA,2),"sq. units") print("Volume =",round(Vol,2),"cu. units")

elif (choice==2):

l=eval(input("Enter length of cuboid: "))

b=eval(input("Enter breadth of cuboid: ")) h=eval(input("Enter height of cuboid: ")) TSA=2\*(l\*b+l\*h+b\*h)

Vol=l\*b\*h

print("Total Surface Area =",round(TSA,2),"sq. units") print("Volume =",round(Vol,2),"cu. units")

elif (choice==3):

r=eval(input("Enter radius of sphere: ")) TSA=4\*Pi\*r\*r

Vol=4\*Pi\*r\*r\*r/3

print("Total Surface Area =",round(TSA,2),"sq. units") print("Volume =",round(Vol,2),"cu. units")

elif choice==0: break

34.

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum=0

for i in range(1,n+1): sum+=x\*\*i

print(sum)

### OR

n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,term=0,x

for i in range(1,n+1): sum+=term

term \*= x print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,term=0,x

for i in range(1,n+1): sum+=term

term \*= -x print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,term=1,x

for i in range(1,n): sum+=term/i term \*= x

print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,term=1,x

for i in range(1,n): sum+=term

term \*= x/(i+1) print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,term=1,x

for i in range(1,n): sum+=term

term \*= -x/(i+1) print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,num,den=1,x\*\*3,3\*2\*1

for i in range(1,n): sum+=num/den num\*=(x\*x) den\*=(2\*i+2)\*(2\*i+3)

print(sum)

1. n=int(input("Enter the number of terms: ")) x=float(input("Enter the value of x: ")) sum,num,den=1,-x\*\*3,3\*2\*1

for i in range(1,n): sum+=num/den num\*=-(x\*x)

den\*=(2\*i+2)\*(2\*i+3) print(sum)

35.

1. n=int(input("Enter the number of lines: ")) for i in range(1,n+1):

for j in range(1,i+1): print(j,end='')

print()

1. n=int(input("Enter the number of lines: ")) for i in range(1,n+1):

print(' '\*(n-i),end='') #For leading space in each line for j in range(1,i+1):

print(j,end='') print()

1. n=int(input("Enter the number of lines: ")) for i in range(1,n+1):

print(' '\*(n-i),end='') #For leading space in each line for j in range(1,i+1):

print(j,end='')

for j in range(i-1,0,-1):

print(j,end='') print()

1. n=int(input("Enter an integer: ")) for k in range(1,n+1):

for i in range (1,k+1): print('\*',end='')

print()

1. n=int(input("Enter an integer: ")) for k in range(1,n+1):

print(' '\*(n-k),end='') for i in range (1,k+1):

print('\*',end='')

print()

36.

1. n=int(input("Enter the value of n: ")) for i in range(1,n+2):

print(' '\*(n+1-i),end='') #For leading space in each line print('\*'\*(2\*i-1))

for i in range(n,0,-1):

print(' '\*(n+1-i),end='') #For leading space in each line print('\*'\*(2\*i-1))

1. n=int(input("Enter the value of n: ")) ch='\*'

for i in range(1,n+2):

print(' '\*(n+1-i),end='') #For leading space in each line print(ch\*(2\*i-1))

if ch=='\*':

ch='@' else: ch='\*'

for i in range(n,0,-1):

print(' '\*(n+1-i),end='') #For leading space in each line print(ch\*(2\*i-1))

if ch=='\*':

ch='@' else: ch='\*'

1. n=int(input("Enter the value of n: ")) for i in range(1,n+1):

print(' '\*(n+1-i),end='') #For leading space in each line if i>1:

print('\*',' '\*(2\*i-3),'\*',sep='') else: print('\*')

for i in range(n+1,0,-1):

print(' '\*(n+1-i),end='') #For leading space in each line print('\*'\*(2\*i-1))

1. n=int(input("Enter the value of n: ")) for i in range(1,n+1):

print(' '\*(n+1-i),end='') #For leading space in each line if i==1:

print('\*')

else: print('\*',' '\*(2\*i-3),'\*',sep='') for i in range(n+1,0,-1):

print(' '\*(n+1-i),end='') #For leading space in each line if i==1:

print('\*')

else: print('\*',' '\*(2\*i-3),'\*',sep='')