Visit QB365-SLMS - https://www.qb365.in/slms/ Complete Questions and Answers

Answers Key:

Long Answer Type Questions

1. n = int(input("How many terms?"))
   # first two terms
   n1, n2 = 0, 1
   count = 0
   # check if the number of terms is valid
   if n < = 0:
   print ("please enter a positive integer")
   elif n ==1:
   print ("Fibonacci sequence upto :", n, ":")
   print (n1)
   else:
   print ("Fibonacci sequence:")
   while count < n:
   print (n1)
   nth= n1 + n2
   # update values
   n1 = n2
   n2 = nth
   count + = 1
2. num = int(input("enter a number: *))
   flag - False
   if num > 1:
   for i in range(2, num):
   if (num % i) == 0:
   flag = True
   break
   if flag:
   print(num, "is not a prime number")
   else :
   print(num, "is a prime number")
3. incr1
   val - 65
   for i in range(0,5):
   for j in range(0, incr):
   ch = chr(val)
   print(ch, end = "")
   val = val + 1
   incr = incr + 2
   print ( )
4. print("The Armstrong numbers in between 1 and 200 are")
   for i in range(1, 201, 1):
   k = 0
   j = i
   while (j > 0) :
   I =j%10
   k = k + I *I*I
   j =j//10
   if (k = - i) :
   print(i)
5. n = 5
   k=n-1
   for i in range(0, n):
   for j in range(0, k):
   print("", end="")
   k-k-1
   for j in range(0, i+1):
   print("*", end-"")
   print ( )
6. num = int(input("Enter a number: "))

```
temp = num
reverse = 0
while temp > 0:
remainder = temp % 10
reverse = (reverse 10) + remainder
temp = temp // 10
if num == reverse:
print ('Palindrome')
else:
print ("Not Palindrome")
```

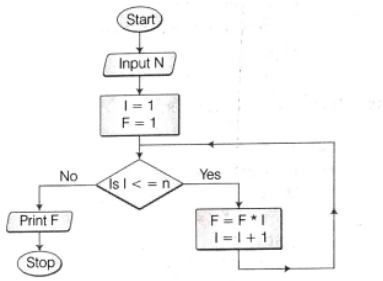
Short Answer Type Questions

7. Differences between identifier and keyword are

Identifier	Keyword
Identifier consists any combination of letters, numbers and underscores.	Keyword must consist only letters.
It allows both uppercase and lowercase.	It allows only lowercase except True, False and None.
e.g. x, sum_5,_mul etc.	e.g. finally, continue, yield etc

8. (i) Num = 2
   (ii) Num = 2
   (iii) Num = 10
9. The operator will be performed before //. Though both have same priority but in this expression * is prior to //.
   A=12//5+5// 7 + 8 - 2+4//2
   =2+5// 7 + 8 - 2+4//2
   operation: //
   = 2 + 0 + 8 - 2 + 4//2
   operation: //
   = 2 + 0 + 8 - 2 + 2
   operation: //
   = 2 + 0 + 6 + 2
   operation : -
   =10
   operation : +
10. i : 10
    j : 15
    k : 13.0
    l :18.0
    m : 2
    n : 49.0
11. a =15
    b=15
    c = False
12. num=7.5
13. **Output**
    x : 15
    y : 3
    z : 12
14. Output is 32
15. **Output**
    a: 175
    b: 15
    x: 50
    y: 25
    z: 100
16. (i) (weight >=115 and weight < 125)
    (ii) ((Donation>=4000 and Donation<= 5000) or Guest = =1)
17. The algorithm is
    Step 1 Input "Enter a Number" N
    Step 2 Let R = N% * 3
    Step 3 If R = 0 then "N is divisible by 3"
    Step 4 If R! = 0 then "N is not divisible by 3"
    Step 5 Stop

QB 365 TM Study Material
QUESTION BANK 365

18. 

19. **Output**
m = 4
k = 2
j = 1

20. for i in range (3, 5):
if (i = = 4) :
print ("Welcome")
else :
print ("No entry")
print ("value of i", i)

21. We can use a for loop with a break statement to stop the loop when the number 7 is encountered.
Program to print numbers from 1 to 10.
stopping at 7
for i in range(1, 11):
if i == 7:
break # Stop the loop when 7 is encountered
print(i)
Output
1
2
3
4
5
6

QB 365 Study Material
QUESTION BANK 365

22. **Output**
23

23. • **range(1, 8)** generates numbers from 1 to 7.
• **if i % 2 == 0:** continue skips even numbers, so only odd numbers are considered for printing.
• **if i == 7:** break stops the loop when i reaches 7.
• **print(i, end="")** prints the numbers on the same line.
Output
1 3 5

24. Value = 30
for val in range (0, Value) : # Error 1
if val % 4 == 0: # Error 2
print (val*4)
elif val%5 == 0: # Error 3
print (val + 3)
else : # Error 4
print (val + 10)

25. **(i) Comments** A hash sign (#) that is not inside a string literal begins a single line comment. We can use triple quoted string for giving multi-line comments.
(ii) Variables A variable in Python is defined through assignment. There is no concept of declaring a variable outside of that assignment.
(iii) Data type It is a set of values and the allowable operations on those values. Python has a great set of useful datatypes

26. (i) p / q + a - c valid
(ii) q (a + b + (- z / 4) invalid because operator missing between q and a
(iii) p / q + a - c valid
(iv) x / y + p * a/b valid
(v) a * +b invalid because use of two operators together
(vi) a +b*c* 2 valid

27. (i) A=10 *4//2+4-2// 2 + 4 - 3 +5//8
=40//2+4-2// 2 + 4 - 3 +5//8 operation: *
=20+4-2// 2 + 4 - 3 +5//8 operation: //
= 20 + 4 - 1 + 4 - 3 +5//8 operation: //
= 20 + 4 - 1 + 4 - 3 + 0 operation: //
= 24 - 1 + 4 - 3 + 0 operation : +

=23+4-3 operation : -
=27-3-24 operation : +
(ii) A=5+6//3+10* 3// 4 + 2 +3//4
= 5 + 2 +10* 3//4+2+3//4
= 5 + 2 +30//4+2+3//4 operation : "
= 5 + 2 + 7 + 2 +3//4 operation: //
= 5 + 2 + 7 + 2 + 0 operation : +
= 7 + 7 + 2 = 16
(iii) A=5* 5+4// 5 + 15 -9//2+10
= 25 + 4//5 +15-9//2 + 10 operation : *
= 25 + 0 + 15 -9//12+10 operation: //
= 25 + 0 + 15 - 4 + 10 operation : +
=40-4+10 operation: -
=36+10=46

28. (i) 21
(ii) -1
(iii) 6
(iv) False
(v) False
(vi) 68
(vii) 86
(viii) True

29. A language is dynamically typed if the type is associated with run-time values and not named variables/fields etc.
This means that you as a programmer can write a little quicker because you do not have to specify types every time.
Python is a dynamically type language. It does not know about the type of the variable until the code is run, so declaration is no use. It stores the value at some memory location and then binds that variable name to that memory container.
We have assigned three values to three variables:
>>> a - 10
>>> name = "Neha"
>>> x = 34.6
But what types are these variables? Let's use the type ( ) module to find out:
Syntax type (variable_name)

QB 365 Study Material
QUESTION BANK 365

30. num = int(input ("Enter a number: "))
if num %2==0:
print("The half number is: ", num/2)
else :
print ("The next number is: ", num+1)
print ("Thank You!")

31. x = int(input("Enter a number:"))
if (x>0) :
print ("Positive")
elif (x<0):
print ("Negative")
else :
print ("Zero")

32. break statement is used to terminate the execution of the loop.
For example,
for i in range(6):
if 1-3:
break
print (1)
The output of the above code will be:
0
1
2
The loop terminates when i becomes 3 due to break statement.
Whereas, continue statement is used to force the next iteration while skipping the statements in the present iteration.
For example,
for i in range(6):
if i == 3:
continue
print(1)
The output of the above code will be
0
1
2

4
5

Case Study Questions

33. (i) Two values - True and False.
(ii) It is an ordered collection of elements (like list, tuple, or string) that allows storing multiple items in a specific order.
(iii) Integer (int), Floating-point (float), and Complex (complex).
(iv) None represents the absence of a value or a null value in Python. It is often used to initialize variables or indicate "no result."
(v) <class 'str'>

34. (i) (colon)
(ii) n==1:
(iii) n1
(iv) n1 + n2
(v) n3

35. (i) (B) When user enters input as location = Chennai and pay =50000, then location == "Chennai" condition will be true. After that, pay < 100000: will be check it also true, so print ("No way") statement 2 will be execute.
(ii) (D) When location = "Surat, pay =50000 entered by user then statement 6 will be execute.
(iii) (D) When location = "Any Other City", pay =1 entered by user then statement 6 will be execute because it does not satisfied any given condition.
(iv) (C) When location = "Delhi", pay =50000 entered by user then statement 4 will be execute.
(v) (D) When location = "Lucknow", pay =65000 entered by user then statement 5 will be execute.

QB365™ Study Material
QUESTION BANK 365