Exercise 6-2

Determine the output displayed.

|  |  |
| --- | --- |
| 1 | for z in range(3, 5, 1):  print( z \* z) |
| 2 | count = 0  for n in numbers:      if n>30:          count+=1 |
| 3 | counter = 0  while counter < 10:  print (format(counter, "<10d"), \  format(counter\*2, "<10d"))  counter += 1 |
| 4 | for i in range(0, 10):  if i < 4:  continue  elif i == 7:  break  else:  print(i)  print(i) |
| 5 | n = 10  # initialize sum and counter  sum = 0  i = 1  while i <= n:  sum = sum + i  i = i+1 # update counter  # print the sum  print("The sum is", sum) |
| 6 | counter = 0  while counter < 3:  print("Inside loop")  counter = counter + 1  else:  print("Inside else") |
| 7 | for letter in "word":  print(letter, end="") |

|  |  |
| --- | --- |
| 8 | # List of numbers  numbers = [6, 5, 3, 8, 4, 2, 5, 4, 11]  # variable to store the sum  sum = 0  # iterate over the list  for val in numbers:  sum = sum+val  # Output: The sum is 48  print("The sum is", sum) |
| 9 | # Program to iterate through a list using indexing  genre = ['pop', 'rock', 'jazz']  # iterate over the list using index  for i in range(len(genre)):  print("I like", genre[i]) |
| 10 | digits = [0, 1, 5]  for i in digits:  print(i)  else:  print("No items left.") |
| 11 | Single-celled amoeba divides itself into two cells every 3 hours. What is the correct piece of code to create a loop to calculate the number of amoebas in 24 hours? Python Test For Loop Question 6  amoebas=1amoebas \*=2print(amoebas)   a) for hour in range (1, 25, 3):  b) for hour in range (3, 25, 3):  c) for hour in range (3, 24, 3):  d) for hour in range (1, 8, 3): |
| 12 | There are steps of bricks with the following structure. Python Test For Loop Question 7 What is the correct piece of code to print out the pattern of the steps? ## #### ###### ########   a) for row in range (5):                print ("##"\*row)  b) for row in range (5):                     print ("#"\*row)  c) for bricks in range (2,8):                 print ("#"\*bricks)  d) for row in range (4):                 print ("##"\*row) |