Python For Loop Practice Questions with Explanations (No Answers)

1. Print each character of a string.

Explanation: Use a for loop to iterate through each character in a string sequence.

CODE:

```
abc = "abhi"

for i in abc:

    print(i)
```

O/p:

a

b

h

i

2. Print all even numbers from a list.

Explanation: Iterate through a list and use a condition to check for even numbers.

CODE:

```
a=[2,3,24,31,5,60]

for i in a:

    if i%2==0:

        print(i)
```

O/P:

2

24

60

3. Calculate the sum of numbers in a tuple.

Explanation: Use a for loop to iterate through a tuple and keep adding each number to a total variable.

CODE:

```
a = (2,4,12,4,8)

sum=0

for i in a:

    sum+=i

print(sum)
```

O/P:

30

4. Print names from a list of strings.

Explanation: Loop through a list of names and print each one.

CODE:

```
names = ["Abhi","Ram","Nikki","Sujju"]

for i in names:

    print(i)
```

O/P:

Abhi

Ram

Nikki

Sujju

5. Print square of numbers using range.

Explanation: Use range to generate numbers, then square each number inside the loop.

CODE:

```
for i in range(1,11):

    sq=i*i

    print(sq)
```

O/P:

1

4

9

16

25

36

49

64

81

100

6. Count vowels in a string.

Explanation: Loop through each character and check if it is a vowel using a condition.

CODE:

```
txt = "hello how are you"

vowel_count=0

for char in txt:

    if char in 'aeiou':

        vowel_count += 1

print(vowel_count)
```

O/P: 7

7. Reverse a string using for loop.

Explanation: Iterate through the string and build a new reversed string by prepending characters.

CODE:

```
str = "abhignya"

rev_str = ""

for char in str:
```

```
    rev_str = char+rev_str
print(rev_str)
```

O/P:

ayngihba

8. Check if elements in list are positive.

Explanation: Use a loop and condition to check and print whether each element is positive or not.

CODE:

```
numbers = [10, -5, 0, 23, -1, 8]
for num in numbers:
    if num > 0:
        print(f"{num} is positive")
    else:
        print(f"{num} is not positive")
```

O/P:

10 is positive

-5 is not positive

0 is not positive

23 is positive

-1 is not positive

8 is positive

9. Print odd-indexed characters in a string.

Explanation: Use range and indexing to print characters that are at odd-numbered positions.

CODE:

```
str = input("enter string:")
for i in range(1,len(str),2):
    print(str[i])
```

O/P:

enter string:abcdef

b

d

f

10. Print multiples of 3 using range.

Explanation: Use range and an if condition to print numbers divisible by 3.

CODE:

```
for num in range(3, 20, 3):
    print(num)
```

O/P:

3

6

9

12

15

18

11. Find the product of numbers in a list.

Explanation: Iterate through the list and multiply each number to get the final product.

CODE:

```
num = [1,2,4,3,5,6]

prod=1

for n in num:
    prod *= n

print(prod)
```

O/P: 720

12. Count how many times a specific character appears in a string.

Explanation: Loop through the string and count how many times a specific character appears.

CODE:

```
text = "iam just sad"

char_to_count = 'a'

count = 0

for char in text:

    if char == char_to_count:

        count += 1

print(count)
```

O/P: 2

13. Print each element of a tuple with its index.

Explanation: Use range and indexing to print the index and corresponding element in the tuple.

CODE:

```
my_tuple = ('apple', 'banana', 'cherry', 'date')

for i in range(len(my_tuple)):

    print(my_tuple[i])
```

O/P:

apple

banana

cherry

date

14. Print numbers from 10 to 1 using range.

Explanation: Use a reversed range to print numbers in descending order.

CODE:

```
for num in range(10, 0, -1):

    print(num)
```

O/P: 10

9

8

7

6

5

4

3

2

1

15. Convert each string in a list to uppercase.

Explanation: Loop through a list of strings and convert each one to uppercase.

CODE:

```
words = ['apple', 'banana', 'cherry', 'date']

for word in words:
    upper_word = word.upper()
    print(upper_word)
```

O/P:

APPLE

BANANA

CHERRY

DATE