

## 2nd feb test loops (1)

August 9, 2023

Q1. Explain with an example each when to use a for loop and a while loop.

answer) FOR LOOP:- A for loop is typically used when you know the exact number of iterations you want to perform. It's especially useful when you need to iterate over a sequence (like a list or range) a specific number of times.

```
[7]: L=[1,2,3]

for i in L:
    print(i**2)
    i
```

1  
4  
9

```
[ ]: WHILE LOOP
```

answer ) WHILE LOOP A WHILE LOOP WE ARE USING WHEN WE DONT KNOW EXACT NUMBER OF ITERATION JUT WE ARE GIVING CONDITION AND ITARATING UNTIL THE VALUE IS TRUE

```
[ ]: sum=0
i=1
while i>=100:
    sum +=i
    i+1
print(sum)
```

```
[ ]: total = 0
i = 1

while i <= 100:
    total += i
    i += 1

print(total)
```

[ ]:

```
[2]: # Using for loop
sum_for = 0
product_for = 1

for i in range(1, 11):
    sum_for += i
    product_for *= i

print("Using for loop:")
print("Sum of the first 10 natural numbers:", sum_for)
print("Product of the first 10 natural numbers:", product_for)
```

Using for loop:  
Sum of the first 10 natural numbers: 55  
Product of the first 10 natural numbers: 3628800

```
[4]: ##usin while looq

sum_while = 0
product_while = 1
count = 1

while count <= 10:
    sum_while += count
    product_while *= count
    count += 1

print("\nUsing while loop:")
print("Sum of the first 10 natural numbers:", sum_while)
print("Product of the first 10 natural numbers:", product_while)
```

Using while loop:  
Sum of the first 10 natural numbers: 55  
Product of the first 10 natural numbers: 3628800

[ ]: Q3

Create a python program to compute the electricity bill for a household. The per-unit charges in rupees are as follows: For the first 100 units, the user will be charged Rs. 4.5 per unit, for the next 100 units, the user will be charged Rs. 6 per unit, and for the next 100 units, the user will be charged Rs. 10 per unit, After 300 units and above the user will be charged Rs. 20 per unit. You are required to take the units of electricity consumed in a month from the user as input.

```
[ ]: unit = int(input("Enter the number of units consumed: "))
if units <= 100:
    total_bill = units * 4.5
elif units <= 200:
    total_bill = 100 * 4.5 + (units - 100) * 6
elif units <= 300:
    total_bill = 100 * 4.5 + 100 * 6 + (units - 200) * 10
else:
    total_bill = 100 * 4.5 + 100 * 6 + 100 * 10 + (units - 300) * 20

return total_bill
```

Q4. Create a list of numbers from 1 to 100. Use for loop and while loop to calculate the cube of each number and if the cube of that number is divisible by 4 or 5 then append that number in a list and print that list.

```
[ ]: ## using for loop
numbers = list(range(1, 101)) # Create a list of numbers from 1 to 100

result_numbers = [] # Initialize a list to store the desired numbers

for num in numbers:
    cube = num ** 3
    if cube % 4 == 0 or cube % 5 == 0:
        result_numbers.append(num)

print("Numbers whose cube is divisible by 4 or 5 (using for loop):",
      ↪result_numbers)
```

```
[ ]: #using while loop
numbers = list(range(1, 101)) # Create a list of numbers from 1 to 100

result_numbers = [] # Initialize a list to store the desired numbers

i = 0
while i < len(numbers):
    cube = numbers[i] ** 3
    if cube % 4 == 0 or cube % 5 == 0:
        result_numbers.append(numbers[i])
    i += 1

print("Numbers whose cube is divisible by 4 or 5 (using while loop):",
      ↪result_numbers)
```

```
[ ]:
```

Q5. Write a program to filter count vowels in the below-given string.

```
string = "I want to become a data scientist"
```

```
[ ]: def count_vowels(input_string):  
    vowels = "aeiouAEIOU"  
    vowel_count = 0  
  
    for char in input_string:  
        if char in vowels:  
            vowel_count += 1  
  
    return vowel_count  
  
string = "I want to become a data scientist"  
vowel_count = count_vowels(string)  
  
print("Number of vowels in the string:", vowel_count)  
When you run this program, it will output:
```

c

Copy code

Number of vowels in the string: 11

This program defines a function `count_vowels` that takes an `input` string and  
↳ iterates through each character. If a character `is` a vowel (both uppercase  
↳ and lowercase), it increments the `vowel_count`. Finally, it prints the total  
↳ count of vowels in the given string.

```
[ ]:
```