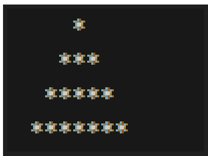
Python Practical Que

Pattern using Loop

```
# Q.1 draw the pattern

#1111

#2222

#3333

#4444


for i in range(1,5):

   for x in range(1,5):

      print(i,end="")

   print()


#Q.2 program to print pattern of numbers.

#1

#12

#123

#1234

#12345


for i in range(1,6):

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

      print(x,end="")
```

```
    print()


#Q.3 program to print pattern of numbers.

#1

#22

#333

#4444

#55555


for i in range(1,6):

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

        print(i,end="")

    print()


#Q.4 program to print the reverse pattern of numbers.

#55555

#4444

#333

#22

#1


for i in range(5,0,-1):

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

        print(i,end="")

    print()

#Q.5  program to print Triangle.
```



```
   *
  ***
 *****
*******
```

```python
a = int(input("Enter Your Value:- "))
for i in range(1,a):
    space=" "*(a-i)
    Print="*"*(2*i-1)
    print(space+Print)
```

Que:- Program to check divisibility of a number

```python
# Check if a number is divisible by another number
num = int(input("Enter a number: "))
divisor = int(input("Enter divisor: "))

if num % divisor == 0:
    print(f"{num} is divisible by {divisor}")
else:
    print(f"{num} is not divisible by {divisor}")
```

Que:- Program to check tax calculation

```python
# Calculate tax based on income
income = float(input("Enter your income: "))
```

```
if income <= 250000:
    tax = 0
elif income <= 500000:
    tax = 0.05 * income
elif income <= 1000000:
    tax = 0.2 * income
else:
    tax = 0.3 * income

print(f"Your tax amount is: {tax}")
```

Que:- Program to find the largest number among three numbers

```
# Find the largest among three numbers
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
c = int(input("Enter third number: "))

if a > b and a > c:
    print(f"The largest number is {a}")
elif b > a and b > c:
    print(f"The largest number is {b}")
else:
    print(f"The largest number is {c}")
```

Que:- Program to check whether a number is positive, negative, or zero

```
# Check if a number is positive, negative, or zero
num = int(input("Enter a number: "))

if num > 0:
    print("Positive Number")
elif num == 0:
    print("Zero")
else:
    print("Negative Number")
```

Que:- Program to check if a number is even or odd

```
# Check if a number is even or odd
num = int(input("Enter a number: "))

if num % 2 == 0:
    print("Even Number")
else:
    print("Odd Number")
```

Que:- Program to calculate the area of a square or rectangle

```python
# Calculate the area of a square or rectangle
print("To calculate area of Square, type: A ")
print("To calculate area of Rectangle, type: R ")
user = input("Enter your choice: ")

if user == "A":
  s = int(input("Enter the side length: "))
  print("The area of the square is:", s * s)
elif user == "R":
  l = int(input("Enter the length: "))
  w = int(input("Enter the width: "))
  print("The area of the rectangle is:", l * w)
else:
  print("Invalid input! Please enter 'A' or 'R'.")
```

Que:- Program to calculate sum or product of two numbers

```python
# Calculate sum or product of two numbers
print("To calculate Sum, type: sum ")
print("To calculate Product, type: pro ")
user = input("Enter your choice: ")
```

```
if user == "sum":
    a = int(input("Enter first number: "))
    b = int(input("Enter second number: "))
    print("The sum is:", a + b)
elif user == "pro":
    x = int(input("Enter first number: "))
    y = int(input("Enter second number: "))
    print("The product is:", x * y)
else:
    print("Invalid input! Please enter 'sum' or 'pro'.")
```

Que:- Program to find factorial of a number

```
# Find factorial of a number
num = int(input("Enter a number: "))
factorial = 1

for i in range(1, num + 1):
    factorial *= i

print(f"Factorial of {num} is {factorial}")
```

Que:- Program to swap two values

```
# Swap two values without using a third variable
a = input("Enter first number: ")
b = input("Enter second number: ")


c = a
a = b
b = c


print("After swapping:")
print("First number:", a)
print("Second number:", b)
```

Que:- Program to check if a number is an Armstrong number

```
n=int(input("enter the number:"))
m=0
e=n
while (e>0):
   f=e%10
   m+=f**3
   e//=10
if (n==m):
   print("armstrong number")
```

```
else :
   print("not a armstrong number")
```

Que:- Program to print Fibonacci series up to n

```
n=int(input("enter a number:"))
a=0
b=1
c=0
while (c<=n):
   print(c)
   a=b
   b=c
   c=a+b
```

Que:- Convert decimal to binary manually

```
# Convert decimal to binary
n = int(input("Enter a decimal number: "))
binary = ""

while n > 0:
   binary = str(n % 2) + binary
   n //= 2
```

```
print("Binary:", binary)
```

Que:- Check if a number is prime

```
# Check if a number is prime
num = int(input("Enter a number: "))

if num < 2:
  print("Not a Prime Number")
else:
  for i in range(2, num):
    if num % i == 0:
      print("Not a Prime Number")
      break
  else:
    print("Prime Number")
```

Que:- Find prime numbers in a range

```
# Find prime numbers in a range
for num in range(2, 100):
  for i in range(2, num):
    if num % i == 0:
      break
    else:
        print(num, end=" ")
```

Que:- Find GCD of two numbers

```
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))

while b != 0:
    temp = b
    b = a % b
    a = temp

print(f"The GCD of the given numbers is {a}")
```

Que:- Check if a number is a palindrome

```
# Check if a number is a palindrome
num = input("Enter a number: ")

if num == num[::-1]:
    print("Palindrome!")
else:
    print("Not a palindrome.")
```

Que:- Generate multiplication table of a number

```
# Generate multiplication table
num = int(input("Enter a number: "))


for i in range(1, 11):
    print(f"{num} x {i} = {num * i}")
```

Que:- Convert Binary to decimal

```
n = input("Enter a binary number: ")  # Take binary input as a string
decimal = 0


# Convert binary to decimal
for digit in n:
    decimal = decimal * 2 + int(digit)


print("Decimal equivalent:", decimal)
```

Que:- Program to display first ten Mersenne number.

```
for i in range(1,11):
    x=(2**i)-1
print(x)
```

Que:- Sum of odd number div by 5

```
total = 0
for i in range(1,100):
   if(i%2!=0):
      if(i%5==0):
         total+=i
print(total)
```

Que:- WAP to sum of first n even number

```
n = int(input("Enter any Number"))
Sum = 0
i = 2
count = 0

while count<n:
   Sum+=i
   i+=2
   count+=1
print(Sum)
```

Que:- Find character is Capital/Small Letter ?

```
User = input("Enter a character: ")  # Take user input

if "A" <= User <= "Z":
   print("Character is Uppercase Letter")
elif "a" <= User <= "z":
   print("Character is Lowercase Letter")
```

```
else:
    print("Cannot find character")
```

Que:- Program to check if a number is a palindrome

```
# Taking input from the user
num = int(input("Enter any number: "))

# Store the original number
original_num = num
rev = 0

# Reverse the number
while num > 0:
    digit = num % 10  # Extract last digit
    rev = (rev * 10) + digit  # Append the digit to rev
    num = num // 10  # Remove the last digit

# Check if the original number is equal to its reverse
if rev == original_num:
    print("Palindrome Number")
else:
    print(original_num, "is not a Palindrome Number")
```

Que:- WAP to print reversed and sum of given digit form user.

```
# Take input from the user
num = int(input("Enter any number: "))


# Initialize variables
digit_sum = 0  # To store the sum of digits
rev = 0  # To store the reversed number


# Reverse the number and calculate digit sum
while num > 0:
    digit = num % 10  # Extract the last digit
    rev = (rev * 10) + digit  # Build the reversed number
    digit_sum += digit  # Add the digit to sum
    num = num // 10  # Remove the last digit


# Print the reversed number and the sum of digits
print("Reversed Number:", rev)
print("Sum of Digits:", digit_sum)
```

Que:- Program to calculate daily wages based on working hours

```
# Program to calculate daily wages based on working hours


# Take input for working hours
h = int(input("Enter your working hours: "))
wage = 0  # Initialize wage
```

```
# Calculate daily wages based on working hours
if h <= 8:
    wage = h * 100  # Base rate: ₹100 per hour
elif h <= 12:
    wage = 800 + (h - 8) * 30  # First 8 hours = ₹800, Extra hours (up to 4) = ₹30 per hour
elif h <= 16:
    wage = 800 + (4 * 30) + (h - 12) * 40  # Next 4 hours = ₹40 per hour
else:
    wage = 800 + (4 * 30) + (4 * 40) + (h - 16) * 60  # 8h = ₹800, 4 extra hours = ₹30/hr, next 4 hours = ₹40, beyond 16 hours = ₹60 per hour


# Print the calculated daily wage
print("Daily Wages: ₹", wage)
```

Que:- Find Days, Month, and Seconds

```
# Take input for the number of days
days = int(input("Enter any number of days: "))


# Calculate months and remaining days
month = days // 30  # 1 month = 30 days
remaining_days = days % 30


# Define hours, minutes, and seconds
```

```
hours = days * 24  # 1 day = 24 hours
minutes = hours * 60  # 1 hour = 60 minutes
seconds = minutes * 60  # 1 minute = 60 seconds

# Print results
print(f"Months: {days // 30}, Days: {days}")
print(f"Total hours: {hours}, Total minutes: {minutes}, Total seconds: {minutes * 60}")
```

Que:- Find Hours from Days

```
# Take input for the number of days
days = int(input("Enter number of days: "))

# Convert days to hours
hours = days * 24  # 1 day = 24 hours

# Print the result
print(f"{days} days is equal to {hours} hours.")
```

Que:- program to check whether the word is palindrome or not

```
a = input("Enter any word: ")
x = a[::-1]  # Reverse the string

if a == x:
    print("Palindrome")
else:
    print("Not a Palindrome")
```

Que:- Armstrong Number in range

```
for i in range(100, 800):  # Checking numbers from 100 to 799
    x = i  # Store original number
    y = 0  # Initialize sum variable
    while i > 0:
        r = i % 10  # Get last digit
        i = i // 10  # Remove last digit
        y = y + r ** 3  # Cube and sum the digits

    if x == y:  # Check if it's an Armstrong number
        print(y, "Armstrong Number")
```

Que:- WAP to count alphabets, digits characters in a string.

```
# Input from user
string = input("Enter a string: ")

# Initialize counters
alphabets = 0
digits = 0
special_chars = 0

# Iterate through each character in the string
for char in string:
    if char.isalpha():  # Check if it's an alphabet
        alphabets += 1
    elif char.isdigit():  # Check if it's a digit
        digits += 1
    else:  # Otherwise, it's a special character
        special_chars += 1

# Print the counts
print("Alphabets:", alphabets)
print("Digits:", digits)
print("Special Characters:", special_chars)
```

Que:- WAP to find the length of the string without using len() Function.

```
# Input from user
string = input("Enter a string: ")

# Initialize counter
length = 0

# Iterate through each character in the string and count
for char in string:
    length += 1

# Print the length of the string
print("Length of the string:", length)
```

Que:- WAP to reverse a string.

```
# Input from user

string = input("Enter a string: ")



# Reverse the string using slicing

reversed_string = string[::-1]



# Print the reversed string

print("Reversed string:", reversed_string)
```

Que:- Remove consonants from a string and count the vowels:

```
# Input from user
input_string = input("Enter a string: ")

# Define vowels
vowels = "AEIOUaeiou"

# Create a new string with only vowels and count them
new_str = ""
vowel_count = 0

for char in input_string:
    if char in vowels:
        new_str += char  # Append only vowels to new_str
        vowel_count += 1  # Count vowels

# Print the result
print("String after removing consonants:", new_str)
print("Number of vowels:", vowel_count)
```

Que:- Program to take full name as input and display he abbreviations of the first and middle name.

```
a = input("Enter Your Name: ")  # Take full name as input
b = a.split()  # Split the name into parts

# Check if the input has at least two parts (first name and last name)
if len(b) >= 3:
    print(b[0][0] + ".", b[1][0] + ".", b[2])  # Abbreviate first and middle name, keep last name
elif len(b) == 2:
    print(b[0][0] + ".", b[1])  # If only first and last name are provided
else:
    print("Please enter at least a first and last name.")
```

Python List Programs

1. Traverse a List and Print All Odd Numbers

```
a = [1, 3, 56, 7, 9, 3]

for i in a:
    if i % 2 != 0:
        print(i)
```

Explanation: Loops through the list and prints only odd numbers.

---

2. Find the Sum of All Items in a List

```
a = [1, 3, 56, 7, 9, 3]
print("This is the sum of your list:", sum(a))
```

Explanation: Uses sum() function to calculate the total sum of the list.

---

3. Find the Product of All Items in a List

```
a = [1, 3, 56, 7, 9, 3]
pro = 1

for num in a:
    pro *= num

print("Product of all items in the list:", pro)
```

Explanation: Multiplies all elements in the list using a loop.

---

4. Find the Sum of Odd and Even Numbers in a List

```
a = [1, 3, 56, 7, 9, 3]
even = 0
odd = 0

for i in a:
    if i % 2 == 0:
        even += i
    else:
        odd += i

print("Sum of all Even Numbers:", even)
print("Sum of all Odd Numbers:", odd)
print("Total Sum of Even and Odd Numbers:", even + odd)
```

Explanation: Separates even and odd numbers and calculates their sums.

---

5. Find the Largest and Smallest Number in a List

```
numbers = [12, 45, 78, 23, 89, 5, 34]

largest = max(numbers)
```

```
smallest = min(numbers)

print("Largest number:", largest)
print("Smallest number:", smallest)
```

Explanation: Uses max() and min() functions to find the largest and smallest numbers.

---

6. Traverse a List

```
a = [1, 3, 56, 7, 9, 3]

for i in a:
   print(i)
```

Explanation: Iterates through the list and prints each element.

---

7. Insert an Element at the 2nd Position in a List

```
a = [1, 3, 56, 7, 9, 3]

a.insert(2, "Vikas")
print(a)
```

Explanation: Uses insert(index, value) to insert an element at a specific position.

---

8. Count the Occurrence of a Number in a List

```
numbers = [1, 2, 3, 4, 2, 5, 2, 6, 7, 2]

target = 2

count = numbers.count(target)
```

```
print("The number", target, "appears", count, "times in the list.")
```

Explanation: Uses count(value) to find the number of times a value appears in the list.

---

9. Calculate the Mean of Elements in a List

```
numbers = [10, 20, 30, 40, 50]
sum_of_numbers = sum(numbers)
count_of_numbers = len(numbers)
mean = sum_of_numbers / count_of_numbers

print(f"The mean of the list is: {mean}")
```

Explanation: Calculates the mean by dividing the sum of elements by the number of elements.

Python Dictionary Programs - Notes

1. Reverse a Dictionary and Display Elements on Different Lines

- Concept: Convert dictionary to a list of items, reverse it, and create a new dictionary.

```
a = {"Name": "Vikas", "Course": "O Level", "Age": 18}
b = a.items()
c = list(b)
c.reverse()

x = dict(c)

for i in x:
   print(i, ":", x[i])
```

Output Example:

```
Age : 18
Course : O Level
Name : Vikas
```

---

2. Traverse a Dictionary and Display Elements on the Same Line

- **Concept: Use a loop to print dictionary elements in a single line.**

```python
a = {"Name": "Vikas", "Course": "O Level", "Age": 18}

for i in a:
    print(i, ":", a[i], end=" ")
```

Output Example:

```
Name : Vikas Course : O Level Age : 18
```

---

3. Input Student Details in a Dictionary and Delete a Particular Roll No Using Name

- **Concept: Store student details with roll numbers as keys and delete a specific roll number.**

```python

b = int(input("Enter How Many Students: "))
a = {}

for i in range(0, b):
    x = int(input("Enter Your Roll No: "))
    y = input("Enter Your Name: ")
    a[x] = y

print(a)

c = int(input("Enter Which Roll You Want to Delete: "))

if c in a:
    del a[c]
    print("Student", c, "has been Deleted")
else:
    print("Roll", c, "Not Found")
print(a)
```

Output Example:

```
Enter How Many Students: 2
Enter Your Roll No: 101
Enter Your Name: Rahul
Enter Your Roll No: 102
Enter Your Name: Raj
{101: 'Rahul', 102: 'Raj'}
Enter Which Roll You Want to Delete: 101
Student 101 has been Deleted
{102: 'Raj'}
```

---

4. Display Information in Ascending Order of Names

- Concept: Store names with age and display them in sorted order.

```
b = int(input("Enter How Many Students: "))
a = {}

for i in range(0, b):
  x = input("Enter Your Name: ")
  y = int(input("Enter Your Age: "))
  a[x] = y

print(sorted(a))
```

Output Example:

```
Enter How Many Students: 3
Enter Your Name: Rahul
Enter Your Age: 20
Enter Your Name: Amit
Enter Your Age: 22
Enter Your Name: Raj
Enter Your Age: 19
['Amit', 'Rahul', 'Raj']
```

---

5. Student Mark List

- **Concept: Store student names with marks and display them.**

```
b = int(input("Enter How Many Students: "))
a = {}

for i in range(0, b):
    x = input("Enter Your Name: ")
    y = int(input("Enter Your Grade: "))
    a[x] = y

for c in a:
    print(c, "\t :", a[c])
```

Output Example:

```
Enter How Many Students: 2
Enter Your Name: Rahul
Enter Your Grade: 85
Enter Your Name: Raj
Enter Your Grade: 90
Rahul    : 85
Raj      : 90
```

Test Que

Write a Python program that takes input for the prices of 5 items, stores them in a list, prints the list, calculates and prints the total sum, product, and average of the prices.

```
x = []
y = 1

for i in range(5):
    a = int(input("Enter Your Items Price: "))
    x.append(a)

print("Your Items Price:-", x)
print("Total:-", sum(x))

for b in x:
    y *= b

print("Product:-", y)
print("Average", sum(x) / 5)
```

Find Second Largest Number

```
x = []


for i in range(5):

    a = int(input("Enter Your Value: "))

    x.append(a)


print(x)

x.sort()

x.reverse()

print(x[1])
```

Write a Python program that prints a pyramid pattern

```
a = int(input("Enter Your Value:- "))
for i in range(1,a):
    space=" "*(a-i)
    Print="*"*(2*i-1)
    print(space+Print)
```

Write a Python program that creates a list of numbers from 1 to 10 and then removes all the odd numbers from the list. Finally, print the updated list containing only even numbers.

```
x=[]
```

```
for i in range(11):
   if(i%2==0):
      x.append(i)

print(x)
```

Practice Que of User Defined Function

Que:- Factorial Number

```
def fact(a):
  if a < 0:
    return "Factorial is not defined for negative numbers!"

  f = 1
  x = 1
  while x <= a:
    f *= x  # Multiplying numbers to get factorial
    x += 1

  return f  # Returning the factorial value

# Taking user input
z = int(input("Enter Your Value: "))
result = fact(z)

# Printing the result
print("Factorial:", result)
```

Que:- Fibonacci Series

```
def fib(x):
  if x < 0:
    print("Fibonacci series is not defined for negative numbers!")
    return

  a=0
  b=1
  while a <= x:
    print(a, end=" ")  # Print sequence in one line
    a, b = b, a + b  # Update values

# Taking user input
z = int(input("Enter Your Value: "))
fib(z)
```

Que:-Armstrong Number

```
def arm(x):
  z = x
  num_digits = len(str(x))  # Count the number of digits
  f = 0

  while x > 0:
    r = x % 10
    x //= 10
    f += r ** num_digits  # Raise to the power of the number of digits

  return f == z  # Return True if Armstrong, otherwise False

# Taking user input
a = int(input("Enter Your Value: "))

# Printing the result
if arm(a):
  print(f"{a} is an Armstrong Number")
else:
  print(f"{a} is Not an Armstrong Number")
```

Que:- Find Palindrome Number

```
def pal(x):
    z=x
    f=0
    while(x>0):
        r=x%10
        x=x//10
        f=(f*10)+r
    if(f==z):
        print("Palindrome Number")
    else:
        print("Not Palindrome Number")

a=int(input("Enter Your Value:-"))
pal(a)
```

Que:- Find Palindrome Word

```
def pal(x):
    a=x[::-1]
    if(a==x):
        print(" Your Word is a Palindrome ")
    else:
        print(" Your Word not  is a Palindrome ")

z=input("Enter Your Value:-")
pal(z)
```

Que:- Find Prime Number

```
def pri(x):
    for i in range(2,x):
        if(x%i==0):
            print("Not Prime")
            break
    else:
        print("Prime Number")
```

```
z=int(input("Enter Your Value:-"))
pri(z)
```

1. Write a program to generate three random integers between 100 and 999 that are divisible by 5.

```
import random

count = 0
while count < 3:
    num = random.randint(100, 999)
    if num % 5 == 0:
        print(num)
        count += 1
```

2. Write a Python program that accepts a sentence from the user and displays the longest word along with its length.

```
sentence = input("Enter a sentence: ")
words = sentence.split()
longest = max(words, key=len)
print("Longest word:", longest)
print("Length:", len(longest))
```

3. Write a function that accepts a number of days and returns the equivalent number of weeks and remaining days.

```
def convert_days(days):
    weeks = days // 7
    remaining_days = days % 7
    return weeks, remaining_days

# Example
d = int(input("Enter number of days: "))
w, r = convert_days(d)
print(f"{d} days = {w} week(s) and {r} day(s)")
```

4. Write a program to display only the last line of a given text file.

```
with open("example.txt", "r") as file:
    lines = file.readlines()
    if lines:
        print("Last line:", lines[-1].strip())
    else:
        print("File is empty.")
```

5. Write a function that accepts a date in "DD-MM-YYYY" format as an argument and returns the day, month, and year separately.

```
def split_date(date_str):
    day, month, year = date_str.split("-")
    return day, month, year

# Example
date_input = input("Enter date in DD-MM-YYYY format: ")
d, m, y = split_date(date_input)
print("Day:", d)
print("Month:", m)
print("Year:", y)
```

6. Write a NumPy program to create an array containing ten 0s, ten 1s, and ten 7s.

```
import numpy as np

array = np.concatenate((np.zeros(10), np.ones(10), np.full(10, 7)))
print(array)
```

7. Write a program to create a one-dimensional NumPy array containing numeric values from 0 to 9, representing holidays.

```
import numpy as np

holiday_array = np.arange(10)
print("Holiday array:", holiday_array)
```

---

8. Write a NumPy program to extract all even numbers from a given array.

```
import numpy as np

arr = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
even_numbers = arr[arr % 2 == 0]
print("Even numbers:", even_numbers)
```

---

9. Write a Python program to create a new string by removing consecutive duplicate letters from a given string.

```
def remove_consecutive_duplicates(s):
    result = s[0] if s else ""
    for char in s[1:]:
        if char != result[-1]:
            result += char
    return result

# Example
string = input("Enter a string: ")
new_string = remove_consecutive_duplicates(string)
print("Processed string:", new_string)
```

---

10. Write a Python program to print all odd numbers from 1 to 100, followed by the number 8, in ascending order.

```
odds = [i for i in range(1, 101) if i % 2 != 0]
odds.append(8)
odds.sort()
print("Sorted numbers:", odds)
```

11. Write a program to print the first ten numbers of the Fibonacci sequence.

```
a, b = 0, 1
count = 0

print("Fibonacci sequence:")
while count < 10:
    print(a, end=" ")
    a, b = b, a + b
    count += 1
```

12. Write a program to calculate the length of all sub-tuples of a tuple and store the lengths in a new tuple named tags.

```
data = (("apple", "banana"), ("cat",), ("dog", "elephant", "fish"))
tags = tuple(len(sub) for sub in data)
print("Lengths of sub-tuples:", tags)
```