🌟 Flow Control in Python

This notebook covers examples of flow control structures in Python, such as conditionals and loops.

`Title: Mastering Control Structures, Loops, and Operators in Python`

Task 1: Write a Python program to find the largest of three numbers.

In [ ]:
```python
# Taking input from users.
a = int(input("Enter the first number: "))
b = int(input("Enter the second number: "))
c = int(input("Enter the third number: "))

# using if-elif-else to find the largest number.
if a >= b and a >=c:
  print("The largest number is:", a)
elif b >= a and b >= c:
  print("The largest number is:", b)
else:
  print("The largest number is:", c)
```

```
Enter the first number: 10
Enter the second number: 60
Enter the third number: 40
The largest number is: 60
```

Task 2: Create a program that checks if a number is even or odd.

In [ ]:
```python
# Taking input from users.

num = int(input("Enter a number:"))

# Using if-else to check even or odd.

if num % 2 == 0:
  print(num,"is an even number")
else:
  print(num,"is an odd number")
```

```
Enter a number:10
10 is an even number
```

Task 3: Write a program that prints numbers from 1 to 100 but skips numbers divisible by 7.

In [ ]:
```python
for num in range(1,101):
  if num % 7 == 0:  #  checks if the number is divisible by 7
    continue  # skip this number
  print(num,end=',')
```

```
1,2,3,4,5,6,8,9,10,11,12,13,15,16,17,18,19,20,22,23,24,25,26,27,29,30,31,32,33,3
4,36,37,38,39,40,41,43,44,45,46,47,48,50,51,52,53,54,55,57,58,59,60,61,62,64,65,6
6,67,68,69,71,72,73,74,75,76,78,79,80,81,82,83,85,86,87,88,89,90,92,93,94,95,96,9
7,99,100,
```

Task 4: Write a program to print the multiplication table of a given number.

In [ ]:
```
num = int(input("enter a number:"))

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

```
enter a number:6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

Task 5: Write a program that uses a nested loop to print a right-angled triangle pattern of stars based on user input.

In [ ]:
```
rows = int(input("Enter the number of rows:"))

for i in range(1,rows+1): # outer loop rows
  for j in range(i):      #inner loop for *
    print("*", end="")
  print()
```

```
Enter the number of rows:5
*
**
***
****
*****
```

Task 6: Create a Python program that prints the following pattern using nested loops:

```
1
12
123
1234
```

In [ ]:
```
for i in range(1,5):
  for j in range(1, i+1):
    print(j,end='')

  print()
```

```
1
12
123
1234
```

Task 7: Create a Python program to check if a character entered by the user is a vowel or consonant.

In [ ]:
```python
# Input from users
char = input("Enter a single charcter: ").lower() # lower handle both uppercase

# check if the input is a single character
if len(char) == 1 and char.isalpha():
  if char in 'aeiou':
    print(f"{char} is a vowel.")
  else:
    print(f"{char} is a consonent.")
else:
  print("Invalid input. Please enter a single alphabet character.")
```

```
Enter a single charcter: r
r is a consonent.
```

🔷 Task 8: Diamond Pattern Generator

Write a Python program that generates a diamond-shaped star pattern.

Example Output (n = 5):

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

In [2]:
```python
n = 5

#Top half of diamond
for i in range(1, n +1):
  spaces = n - i        # spaces before stars
  stars = 2 * i -1      # odd number of stars (1,3,5,..)
  print(" " * spaces + "*" * stars)    #print the rows

# Bottom half of the diamond
for i in range(n -1, 0, -1):
  spaces = n - i
  stars = 2 * i -1
  print(" " * spaces + "*" * stars)
```

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

Task 9: Write a Python program to reverse the digits of a given number.

In [3]:
```
# Input from users
num = int(input("Enter the number:"))

# Initialize reversed number to 0
reversed_num = 0

# using a loop to reverse the digits
while num != 0:
    digit = num % 10                          # Gets the last digit of num (e.g.,
    reversed_num = reversed_num * 10 + digit # Shift reversed_num left and add a
    num = num // 10                           #Remove the last digit from num (e.

print("Reversed Number:",reversed_num)
```

```
Enter the number:98765
Reversed Number: 56789
```

Task 10. Create a Python program that generates the following pyramid pattern:

```
1
121
12321
1234321
123454321
```

In [ ]:
```
n = 5  # Number of rows

for i in range(1, n + 1):
    # First half: increasing numbers from 1 to i
    for j in range(1, i + 1):
        print(j, end='')

    # Second half: decreasing numbers from i-1 to 1
    for j in range(i - 1, 0, -1):
        print(j, end='')

    print()  # Move to the next line
```

```
1
121
12321
1234321
123454321
```

In [5]:
```
print("The End".center(100, "*"))
```

```
**********************************************The End**************************
******************
```