Nested conditional operators
Using more than one conditional operator
Syntax:
opr1 if opr2 else opr3 if opr4 else opr5 if opr6 else opr7

Example:
```
# write a program to find max of three numbers
n1=int(input("Enter n1 value"))
n2=int(input("Enter n2 value"))
n3=int(input("Enter n3 value"))
print(n1,"is max") if n1>n2 and n1>n3 else print(n2,"is max") if n2>n3 else
print(n3,"is max")
```
Output:
```
Enter n1 value40
Enter n2 value20
Enter n3 value10
40 is max
>>>
=== RESTART: C:/Users/user/Desktop/python6pm/py14.py ==
Enter n1 value30
Enter n2 value60
Enter n3 value10
60 is max
>>>
=== RESTART: C:/Users/user/Desktop/python6pm/py14.py ==
Enter n1 value10
Enter n2 value20
Enter n3 value30
30 is max
>>>
```

Example:
```
# write a program or script to find input character is alphabet,digit or special
character

ch=input("Enter any character")
print("alphabet") if ch>='a' and ch<='z' or ch>='A' and ch<='Z' else
print("digit") if ch>='0' and ch<='9' else print("sepcial character")
```

Output:

```
Enter any charactera
alphabet
>>>
=== RESTART: C:/Users/user/Desktop/python6pm/py15.py ==
Enter any character7
digit
>>>
=== RESTART: C:/Users/user/Desktop/python6pm/py15.py ==
Enter any character$
sepcial character
>>>
```

ord()

it is a predefined function in python, which return ascii value or input character

ord('a') ➔ 97
ord('A') ➔ 65
ord('b') ➔ 98
ord('B') ➔ 66

chr()

it is a predefined function in python, which return character value of input ascii value

chr(65) ➔ A
chr(66) ➔ B
chr(97) ➔ a
chr(98) ➔ b

Example:

```
# write a program to convert input character into uppercase or lowercase
ch=input("enter any character")
res=chr(ord(ch)-32) if ch>='a' and ch<='z' else chr(ord(ch)+32) if ch>='A' and ch<='Z'  else "invalid input"
print(ch,res)
```

Output:

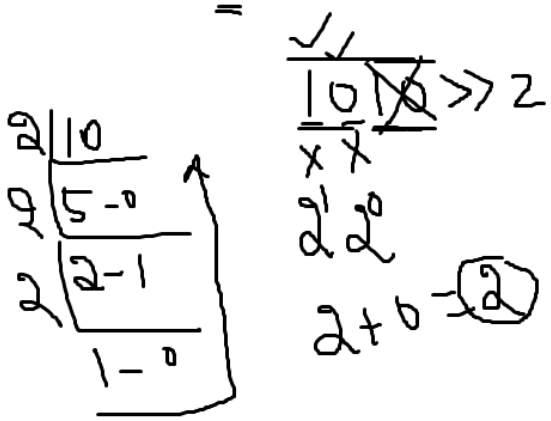
```
enter any charactera
a A
>>>
```

```
=== RESTART: C:/Users/user/Desktop/python6pm/py16.py ==
enter any characterB
B b
>>>
=== RESTART: C:/Users/user/Desktop/python6pm/py16.py ==
enter any character*
* invalid input
>>>
```

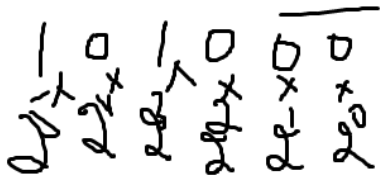
In python relational operators are used for comparing strings and numeric values.

https://csiplearninghub.com/python-fundamentals-practice-questions/

Bitwise operators
Bitwise operators are binary operators
Bitwise operators perform operations by converting data into 0's and 1's

Operator	Description
>>	Right shift operator This operator is used for shifting number of bits towards right side. This operator is used for memory management By shifting number of bits towards right side the value get decremented. By removing bits a=10 a=a-2 8 a=10 a=a>>2 

	>>> a=10
	>>> b=a>>2
	>>> print(a,b)
	10 2
	>>> print(bin(a),bin(b))
	0b1010 0b10
	>>>
	Formula : num//2 pow n
	>>> a=56
	>>> b=a>>4
	>>> print(a,b)
	56 3
	>>>

<<	Left shift operator
	This operator is used to shift number of bits towards left side, by shifting number of bits towards left side the value get incremented, it will those number of bits at right side.
	
	a=10
	
	b=a<<2
	
	 
	
	 
	
	>>> a=10
	>>> b=a<<2
	>>> print(a,b)
	10 40
	>>>
	
	Formula : num * 2 pow n

&	Bitwise and operator

|	Bitwise or operator
~	Bitwise not operator
^	Bitwise XOR operator

A Logic gate is a kind of the basic building block of a digital circuit having two inputs and one output. ... Logic gates are **used to carry out the logical operations on single or multiple binary inputs and result in one binary output**. In simple words, logic gates are the electronic circuits in a digital system.