```
'''
Basic string operations
'''
s1 = "ABCDE12345"
print(len(s1))
print(s1[0], s1[-1])  #indexing
print(s1[0:5])
#*and +
s2 = s1 + str(10)
print(s2)
s3 = s1 * 2
print(s3)
#Some useful functions to be used with string
s4 = s1.lower()
print(s4)
s5 = s1 + "    " + "XYZ" + "   "
print(s5)
print(len(s5))
s6 = s5.rstrip()
print(s6)
print(len(s6))
s7 = "     ABCDE"
s8 = s7.lstrip()
print(s8)

# using string to format output
mName = "ICT133"
mCre = 5
mHour = 18
#Creating a string and use it to format the output
mesg = "Module is {} Credit is {} Hours = {}".format(mName, mCre, mHour)
print(mesg)

mesg = "Module is {:20} Credit is {} Hours = {}".format(mName, mCre, mHour)
print(mesg)



10
A 5
ABCDE
ABCDE1234510
ABCDE12345ABCDE12345
abcde12345
ABCDE12345    XYZ
20
ABCDE12345    XYZ
17
ABCDE
Module is ICT133 Credit is 5 Hours = 18
Module is ICT133               Credit is 5 Hours = 18


'''
Ask the user to enter a string in the form of
L.d.dd C.3.04 A.5.20
The program will print:
Block C level 3 Room 4
'''
```

```
loc = input("Enter location")
print(loc)
block = loc [0]
level = loc [2]
room = loc[4:]

#Alternatively, use the split function
block, level, room = loc.split(".")

print("Block is", block , "Level is", level, "Room is", room)


Enter location A.2.12
 A.2.12
Block is  A Level is 2 Room is 12



'''
Ask the user to enter a string which has the following:
dddd   dddd.dd d.dd
The first is the quantity
Second is the unit price
The last is the discount
Compute the total, discount and the final total
'''
inputValue = input("Enter a string")
qty, unitPrice, discount = inputValue.split()
print(qty * 10)
print(qty, unitPrice, discount)

#Convert values into numbers
qtyValue = int(qty)
priceValue = float(unitPrice)
discountValue = float(discount)

total = qtyValue * priceValue
dis = total * discountValue
grand = total - dis

#Print the results
print("Total is", total)
print("Discount is", dis)
print("Grand total is", grand)



Enter a string 10 12.5 0.01
10101010101010101010
10 12.5 0.01
Total is 125.0
Discount is 1.25
Grand total is 123.75

Boolean operators

num1 = 10
num2 = 20
num3 = 30
print(num1 == num2)
print(num1 <= num2)
```

```
print(num1 != num2)

print(num1 < num2 and num2 < num3)
print(num1 < num2 and num2 > num3)
print(num1 < num2 or num2 > num3)
```

```
False
True
True
True
False
True
```

```
'''
Ask the user to enter 2 numbers
If the 2 numbers are the same, print the message "Same numbers"
Print the smaller number
'''
num1 = int(input("Enter the first number"))
num2 = int(input("Enter the second number"))
if num1 < num2 :
    print("Smaller number is", num1)
else :
    print("Smaller number is", num2)

#to handle the third case: both numbers are the same, we use
#if...elif...

if num1 < num2 :
    print(num1)
elif num2 < num1 :
    print (num2)
else :
    print("They are the same")
```

```
Enter the first number 10
Enter the second number   10
Smaller number is 10
They are the same
```

```
'''
Ask the user to enter 3 different numbers
Print the number whose value is the middle value
E.g.
100  500  300 - output is 300
'''
num1 = int(input("Enter the first number"))
num2 = int(input("Enter the second number"))
num3 = int(input("Enter the third number"))

if num2 < num1 and num1 < num3 or num3 < num1 and num1 < num2:
    print(num1)
elif num1 < num2 and num2 < num3 or num3 < num2 and num2 < num1:
```

```
    print(num2)
else:
    print(num3)
```

```
Enter the first number 10
Enter the second number100
Enter the third number1
10
```

ForLoop

```
for idx in [1,2,3,4,5,6,7,8,9,10]:
    print(idx,"x 5", "=", idx*5)
```

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
```

```
for idx in range(1,11):
    print(idx,"x 5", "=", idx*5)

for idx in range(1,100,9):
    print(idx,"x 5", "=", idx*5)
```

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50

1 x 5 = 5
10 x 5 = 50
19 x 5 = 95
28 x 5 = 140
37 x 5 = 185
46 x 5 = 230
55 x 5 = 275
64 x 5 = 320
73 x 5 = 365
82 x 5 = 410
```

```
91 x 5 = 455
```

```
for idx in range(100,-1,-5):
    print(idx)
```

```
100
95
90
85
80
75
70
65
60
55
50
45
40
35
30
25
20
15
10
5
0
```

```
s99 = "ABCABC"
s100 = s99.replace("A", "XXX")
print(s100)
```

```
XXXBCXXXBC
```

```
'''
Ask user to enter 3 different integers
Sum and print the numbers following the rules below:
  if all different, add them all
  if any 2 are the same, add the other alone
  if all same, the sum is 0
'''

#Use eval to read 3 numbers in one statement
num1, num2, num3 = eval(input("Enter 3 different numbers"))

#Test before doing more
#Remember to use, when entering numbers
print(num1, num2 , num3)

#Next, use if..elif.. to figure out what to add
#There are many ways to write, this is just one example
if num1 == num2 and num2 == num3:
    print(0)
elif num1 != num2 and num2 != num3 and num3 != num1:
    print(num1 + num2 + num3)
```

```
elif num1 == num2:      #Don't need check num3
    print(num3)
elif num2 == num3:
    print(num1)
else:
    print(num2)
# eval() is able to break the input into 3 numbers, and assign them
# to num1, num2, and num3 at the same time.
# If you use int() , you have to do 3 input statements, and convert
# the input to integer one by one.
# be careful with the syntax: the : at the end and the identation


Enter 3 different numbers 5,5,5
5 5 5
0


'''
Here is the question
Ask user to enter 3 integer scores (0 ~ 100)
The program will print the result based on the following
  all scores >= 50        --> Pass
  1 score (anyone) < 50   --> Retest
  2 or more scores <50    --> Fail

'''
#Use eval or int to get 3 numbers, whichever u like
score1, score2, score3 = eval(input("Enter 3 scores"))

#Use if...
if score1 >= 50 and score2 >= 50 and score3 >= 50:
    print ("Pass")
elif (score1 >= 50 and score2 >= 50) or\
     (score2 >= 50 and score3 >= 50) or\
     (score1 >=50 and score3 >=50):
    print ("Retest")
else:
    print("Fail")

# line17: I need to check 2 scores in pair instead of just 1 score
# e.g if score1 < 50 or score2 < 50 or score3 < 50
# Imagine both score 1 and 2 are < 50, it won't give the right answer
# The key thing to note is line 17, the logic to check only 1 score < 50
# line 17 is very long, you can use \ to break into multiple lines
# to make it easy to read
# This approach is fine if the number of scores are not too many
# If the qn is changed to: 4 score or 5 scores, if..elif.. will become very
lengthly
# Next excercise will use a different approach



Enter 3 scores      10, 56, 70
Retest


'''
```

```
'''
Ask user to enter 4 test scores, and print the result as follows
    all scores >= 50          --> Pass
    1 score (anyone) < 50     --> Retest
    2 or more scores < 50     --> Fail
'''
score1, score2, score3, score4 = eval(input("Enter 4 scores"))
failCount = 0

#We are not going to use the if .... because the combinations to
#check will be too many, and if is easy to make mistake.
# Instead, we check each score individually, and count how many< 50
# Use 4 separate if statements ....
if score1 < 50:
    failCount = failCount + 1 # Update this count for each score < 50
if score2 < 50:
    failCount = failCount + 1
if score3 < 50:
    failCount = failCount + 1
if score4 < 50:
    failCount = failCount + 1

#Determine result based on the checks done above
if failCount == 0:
    print ("Pass")
elif failCount == 1:
    print("Retest")
elif failCount ==2:
    print("Supp paper")
else:
    print("Fail")

# Different ways of solving the problem.
# This is easier and more flexible when the number of stores are too many.
# Imagine the question changed to:
# 1 score < 50 --> retest
# 2 scores < 50 --> supp paper
# 3 or more scores < 50 --> fail
# We don't need to change the if statements, just modify the last if statement.


Enter 4 scores 10,20,40,60
Fail



#We could have use 2 input statements but the question require us to use
# string, split, ....
inputValue = input("Enter plan and age") #This input ha 2 parts:
plan, age = inputValue.split() #Divide the 2 values and save to variables

ageValue = int(age)  #Now it is changed to integer
#Again, you may have done it differently, as long as you get it to work, it's ok

#Determine the rate based on the plan selected
if plan == "C" or plan == "c": #Must handle both upper and lower case
    monthlyFee = 15
else:
    monthlyFee = 25
# Determine discount based on age
```

```
if ageValue <= 16:
    discount = 0.25  #This is 25%
else:
    discount = 0
#Calculate the final feee
finalFee = monthlyFee - (monthlyFee * discount)
print (finalFee)


Enter plan and age C 29
15


'''
Lab2 Question 7
I will modify the question slight: we will not use file, list, etc.
Will do that next week, Right now, focus on if.. else
Later on, will add a loop to that
Think about the part on checking if 3 numbers form a triangle

Ask the user to enter 3 numbers, and check if the 3 numbers
form a triangle
'''
n = int(input("How many times to run?"))
# Set up a loop to run n times
invTri = 0      #Count how many invalid triangles
equTri = 0      #Count equilateral triangle
isoTri = 0      #Count isoceles triangle
scaTri = 0      #Count scalene triangle
for index in range(0, n):
    side1, side2, side3 = eval(input("Enter 3 sides:"))
    #First check: can the 3 sides form a triangle
    if side1 + side2 <= side3 or side2 + side3 <= side1 or side1 + side3 <= side2:
        print("Invalid triangle")
        invTri = invTri + 1
    elif side1 == side2 == side3:  #Alternatively: side1 == side2 == side3
        print("Equilateral")
        equTri = equTri + 1
    elif side1 != side2 and side2 != side3 and side3 != side1: # Yes, must check
side 3 and side 1
        print ("Scalene")
        scaTri = scaTri + 1
    else:
        print("Isoceles")
        isoTri = isoTri + 1

#After the loop, print the counts
print("Invalid triangles:", invTri)
print("Equilateral triangles:", equTri)
print("Scalene triangles:", scaTri)
print("Isoceles triangles:", isoTri)

# Notice i check for different sides on line 15 so that i don't
# need to check ever 2 side after that?
# To summarise the logic in English:4
# 1. If all equal --> equilateral
# 2. else if all different --> Scalene
# 3. else: since they are the same and yet not all different, must
```

```
# be isosceles

# Last part, we add a loop to this.
# We want to be able to enter multiple sets of values.
# We need a for loop

#Introduce the last thing for this session: counting
#I want to count the number of triangles.


How many times to run? 3
Enter 3 sides:      4,4,4
Equilateral
Enter 3 sides: 3,3,3
Equilateral
Enter 3 sides:      5,6,7
Scalene
Invalid triangles: 0
Equilateral triangles: 2
Scalene triangles: 1
Isoceles triangles: 0
```