

# Conditionals

Check if the x is perfect square. If it is, print "x is a perfect square". If not print "x is not a perfect square".

In [1]:

```
#Write your code here
x = 30

x_sqrt= x**0.5

if x_sqrt==int(x_sqrt):
    print(str(x)+" is a perfect square")
else:
    print(str(x)+" is not a perfect square")
```

30 is not a perfect square

In [2]:

```
var={3,4,4,4,4,4,3,3,3,'tub','tub','hechnology'}
var
```

Out[2]:

{3, 4, 'hechnology', 'tub'}

In [3]:

```
x={'a','b','c'}
y={'d','e','a'}

x.intersection_update(y)
print(x)
```

{'a'}

In [4]:

```
x={'a','b','c'}
y={'d','e','a'}

x.intersection(y)
print(x)
```

{'c', 'b', 'a'}

Create a list L of arbitrary length. Get the median.

In [5]:

```
#Write your code here
L = [213, 123, 21, 2, 12, 3]

#Median:
#1) Sort in ascending order [2,3,12,21,123,213]
#2) Check if the number of elements are odd or even
#3) if it is odd: index= INT(n/2)
#    if it is even: n/2 and n/2+1 = (12 + 21) /2 =33/2 = 16.5

sorted_list=sorted(L)

if len(L)%2==1:
    median_index=int(len(L)/2)
    median = sorted_list[median_index]

else:
    index_1=int(len(L)/2)
    index_2=int(index_1 - 1)
    median = (sorted_list[index_1] + sorted_list[index_2])/2

print(median)
```

16.5

If there are fewer than 10 students in the class, then the top three students get an A and the rest get a B. Otherwise, the grade breakdown is as follows:

- A: 90-100
- B: 80-89
- C: 70 - 79
- D: 65 - 69
- Failing: Below 65

Given the list of grades in grades\_list, print the grade received by the last student in this list. So if we have

```
grade_list = [50, 90, 91, 88, 75]
```

we want to print the grade received by the student who received a 75. In this case it would be a B.

In [6]:

```
#Write your code here
grade_list = [50, 90, 91, 88, 75]
a1 = sorted(grade_list,reverse = True)
marks = 75
grade0 = 'A'
grade1 = 'B'
if len(a1)<10:
    if marks in a1[0:3]:
        print("The grade for {} marks is : ".format(marks),grade0)
    elif marks in a1[3:len(a1)]:
        print("The grade for {} marks is : ".format(marks),grade1)
    else:
        print("The marks are not there in list")
```

The grade for 75 marks is : B

## For Loops

Count the number of n's in your name (lower and uppercase count)

In [7]:

```
#Write your code here
name = "Jake Feldman"
a = name.upper()
for i in a:
    if i.upper() == 'n'.upper():
        c = a.count('n'.upper())
print(c)
```

1

Get all of the factors of a number and store them in a list

In [8]:

```
#Write your code here
n = 6 # [1, 2, 3, 6]
l = []
for i in range(1,n+1):
    if n%i == 0:
        l += [i]
print(l)
```

[1, 2, 3, 6]

Generate the first N numbers of the Fibonacci sequence where

$$a_0 = 1$$

$$a_1 = 1$$

$$a_n = a_{n-1} + a_{n-2}$$

In [9]:

```
#Write your code here
n = int(input("Enter a number to check fibonnaci series : "))
a = 0
b = 1
if n == 0:
    print(a)
elif n == 1:
    print(str(a) + ' ' + str(b))
else:
    print(str(a)+' '+str(b),end = ' ')
    for i in range(2,n+1):
        c = a + b
        a = b
        b = c
        print(c,end=' ')
```

Enter a number to check fibonnaci series : 7  
0 1 1 2 3 5 8 13

Compute the dot product of lists l1 and l2. **This is an example where iterating over the index is better than over the elements themselves**

In [10]:

```
#Write your code here
l1 = [2, 3, 4, 5, 6, 1, 90]
l2 = [3, 1, 5, 1, 2, 1, 10]
s = 0
for a,b in zip(l1,l2):
    s += a*b
print(s)
```

947

Given two strings, test how many times the first string contains the second. For example if str1 = "coding is cool" and str2 = "co" then output should be 2.

In [11]:

```
#Write you code here
str1 = "coding is cool"
str2 = "co"
str3 = ""
for i in str2:
    str3 += i
    count = str1.count(str3)
print(count)
```

2