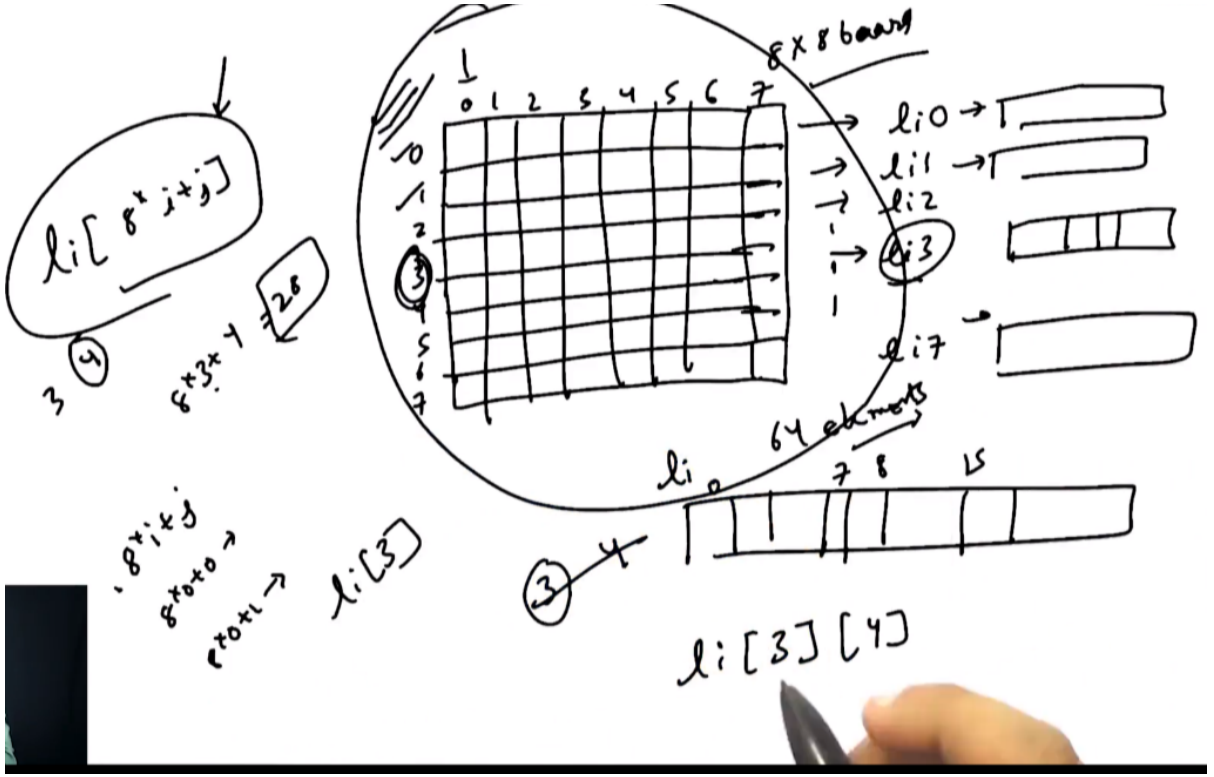
What is the need of 2-D array(list)?



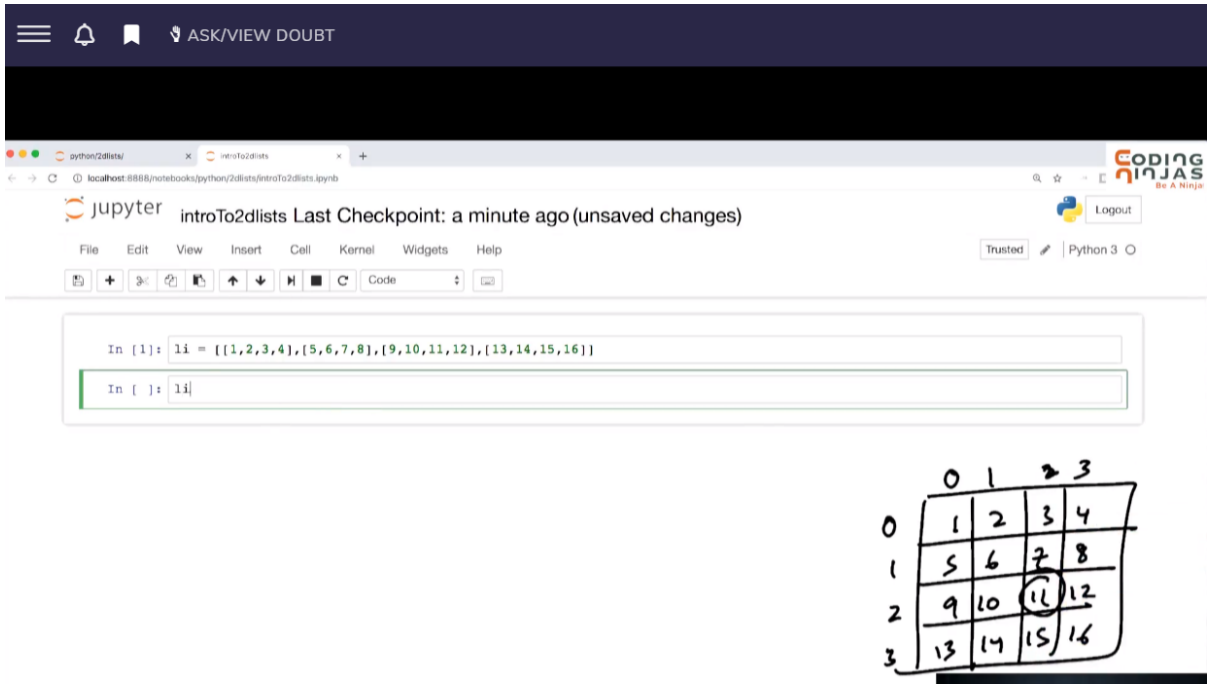
2-D List is a list of lists.

Declaration of 2-d list

In [38]:
```
li = [[1,2,3,4], [5,6,7,8], [9,10,11,12],[13,14,15,16]]
```

In [39]:
```
li
```

Out[39]:
```
[[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]]
```

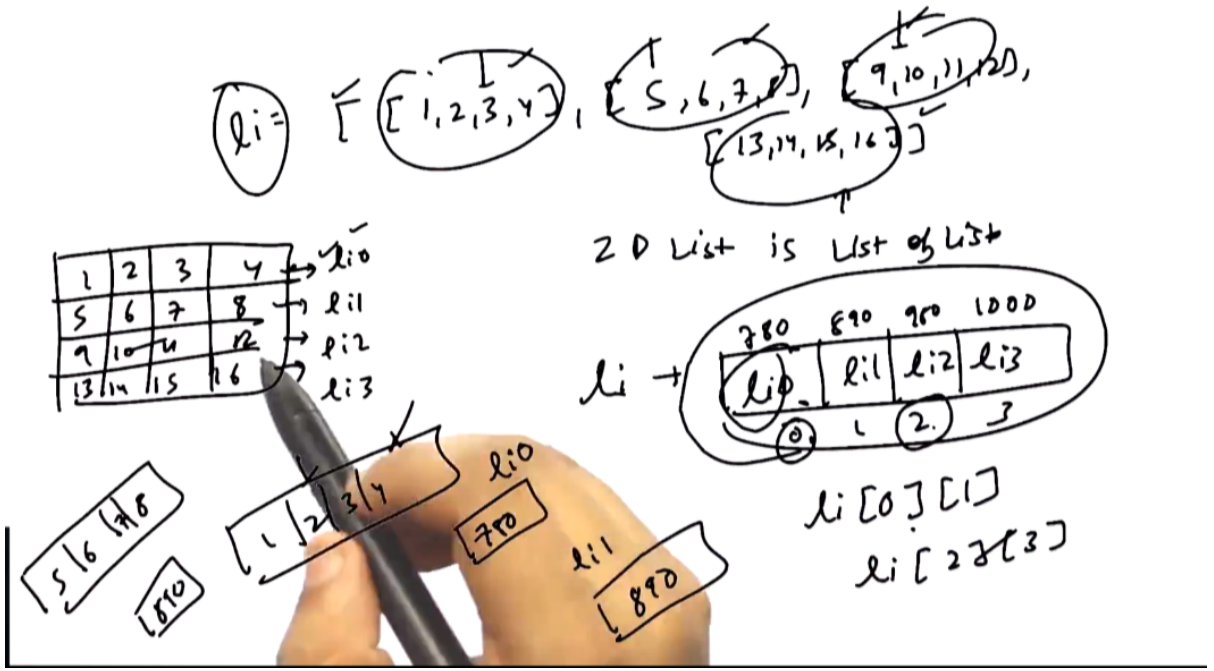


In [40]:
```
li[2][5]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-40-8a1343be0e21> in <module>
----> 1 li[2][5]

IndexError: list index out of range
```

```
In [41]: li[2][2]

Out[41]: 11


In [42]: li[1][0] = 0


In [43]: li

Out[43]: [[1, 2, 3, 4], [0, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]]
```

How 2-d list are stored?



In [44]:
```
li
```

Out[44]:
```
[[1, 2, 3, 4], [0, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]]
```

In [45]:
```
print(li[0])
print(type(li))
print(id )
```

```
[1, 2, 3, 4]
<class 'list'>
<built-in function id>
```

In [46]:
```
print(id(li))

print(id(li[1]))
```

```
2053933985160
2053934220872
```

In [47]:
```
li[1] = "mnr"
```

In [48]:
```
li
```

Out[48]:
```
[[1, 2, 3, 4], 'mnr', [9, 10, 11, 12], [13, 14, 15, 16]]
```

In [49]:
```
print(id(li))

print(id(li[1]))
```

```
2053933985160
2053935042096
```

In [50]:
```
li[1] = [6,7,8,9]
li
```

Out[50]:
```
[[1, 2, 3, 4], [6, 7, 8, 9], [9, 10, 11, 12], [13, 14, 15, 16]]
```

In [51]:
```
print(id(li))

print(id(li[1]))
```

```
2053933985160
2053933877128
```

In [52]:
```
li[0][5]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-52-17af20611d20> in <module>
----> 1 li[0][5]

IndexError: list index out of range
```

Jagged Lists

Jagged- List are those whose column are not same.

In [53]:
```
li = [[1,2,3,4], [2,4],[23,412,412,23,2]]
```

In [54]:
```
for i in li:
    print(i)
```

```
[1, 2, 3, 4]
[2, 4]
[23, 412, 412, 23, 2]
```

In [55]:
```
li[0][3]
```

Out[55]:
```
4
```

In [56]:
```
li[2][5]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-56-8a1343be0e21> in <module>
----> 1 li[2][5]

IndexError: list index out of range
```

List Comprehension

In [57]:
```
l = [1,2,3,4]
k = []
for i in l:
    k.append(i**2)

print(k)
```

```
[1, 4, 9, 16]
```

Using List Comprehension

```
list[output for expression condition]
```

In [58]:
```
l = [1,2,3,4,5]
```

In [59]:
```
# above code same as
h = [x**2 for x in l]
print(h)
```

```
[1, 4, 9, 16, 25]
```

```
new_list = [expression for member in iterable (if conditional)]
```

Square of only even elements

In [60]:
```
l = [1,2,3,4,5]
```

In [61]:
```
l1 = []
for i in l:
    if i%2 == 0:
        l1.append(i**2)

print(l1)
```

```
[4, 16]
```

Same as

In [62]:
```
l = [x**2 for x in l if (x%2==0)]
print(l)
```

```
[4, 16]
```

Multiple of 2 or 3 using list comprehension

In [63]:
```
l = [i for i in range(1,11)]
l
```

Out[63]:
```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

In [64]:
```
l1 = []
```

In [65]:
```
for i in l:
    if (i%2==0) and (i%3==0):
        l1.append(i)

print(l1)
```

```
[6]
```

In [66]:
```
l1 = [k for k in l if (k%2==0) and (k%3==0)]
print(l1)
```

```
[6]
```

Common of two list

In [75]:
```
a = [1,2,3,4,5]
b = [2,4,6,7]
inter = []

for i in a:
    for j in b:
        if i == j:
            inter.append(i)

print(inter)
```

```
[2, 4]
```

Using list comprehension

list[element|| mulitple for loop || condition]

In [68]:
```
a = [1,2,3,4,5]
b = [2,4,6,7]

# multiple for loop
inter = [i for i in a for j in b if i==j]

print(inter)
```

```
[2, 4]
```

If-else condition

In [76]:
```
a = [1,2,3,4,5,6,7]
g = []

for i in a:
    if i%2 == 0:
        g.append(i)

print(g)
```

```
[2, 4, 6]
```

In [77]:
```
# using list comprehesion
g = [i for i in a if i%2== 0]
g
```

Out[77]:
```
[2, 4, 6]
```

> if-else using for loop
>
> list[element if condition else output for loop]

In [78]:
```
g = [ele if ele%2==0 else pass for ele in a]
```

```
  File "<ipython-input-78-94cd969d5e7d>", line 1
    g = [ele if ele%2==0 else pass for ele in a]
                              ^
SyntaxError: invalid syntax
```

In [79]:
```
g
```

Out[79]:
```
[2, 4, 6]
```

In [80]:
```
a
```

Out[80]:
```
[1, 2, 3, 4, 5, 6, 7]
```

Square if element is multiple of 2

In [82]:
```
g = [ele**2 if ele%2 == 0 else ele for ele in a]
g
```

Out[82]:
```
[1, 4, 3, 16, 5, 36, 7]
```

In [84]:
```
m = "mohit"
l=[]
```

In [85]:
```
for i in m:
    l.append(i)

print(l)
```

```
['m', 'o', 'h', 'i', 't']
```

In [86]:
```
print(*l)
```

```
m o h i t
```

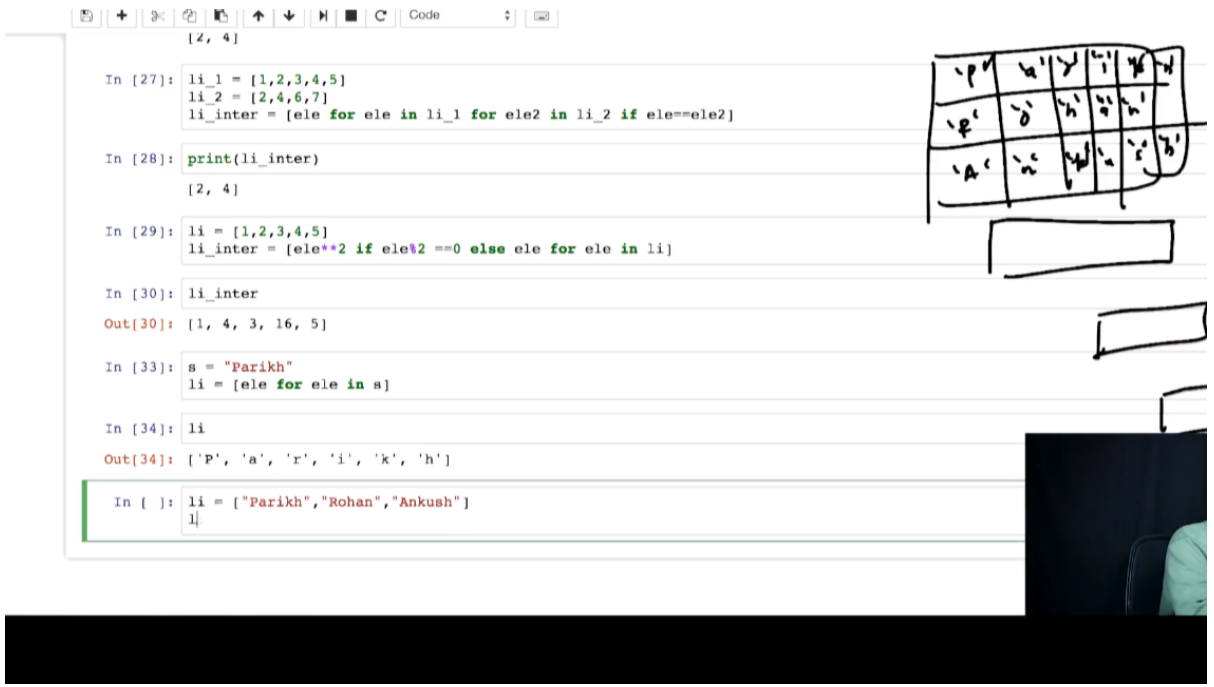
In [87]:
```
l = [m for s in m]
l
```

Out[87]:
```
['mohit', 'mohit', 'mohit', 'mohit', 'mohit']
```

In [89]:
```
# char of strings in one list
l = [i for i in m]
print(l)
```

```
['m', 'o', 'h', 'i', 't']
```

Lists of Lists



In [94]:
```
l = ['mohit','mnr','lalu']
```

In [95]:
```
# 2d list
l1 = [[j for j in i \
        ]for i in l]
l1
```

Out[95]:
```
[['m', 'o', 'h', 'i', 't'], ['m', 'n', 'r'], ['l', 'a', 'l', 'u']]
```

Inner loop output returns a string

In [102]:
```
l1 = [[j for j in i] for i in l]
l1
```

Out[102]:
```
[['m', 'o', 'h', 'i', 't'], ['m', 'n', 'r'], ['l', 'a', 'l', 'u']]
```

In [96]:
```
print(l1)
```

```
[['m', 'o', 'h', 'i', 't'], ['m', 'n', 'r'], ['l', 'a', 'l', 'u']]
```

In [101]:
```
print(*l1)
```

```
['m', 'o', 'h', 'i', 't'] ['m', 'n', 'r'] ['l', 'a', 'l', 'u']
```

In [ ]:

In [ ]:

In [ ]: