

Creating list with power of all the numbers from 1 to 10

Without comprehension

```
list1=[]  
for n in range(1,11):  
    list1.append(n**2)
```

```
print(list1)
```

with list comprehension

```
list2=[n**2 for n in range(1,11)]  
print(list2)
```

Example:

```
uppercase_list=[chr(n) for n in range(65,91)]  
print(uppercase_list)  
lowercase_list=[chr(n) for n in range(97,123)]  
print(lowercase_list)  
matrix=[[int(input()) for j in range(2)] for i in range(2)]  
print(matrix)
```

Adding two matrices

```
print("Input elements of matrix1")  
matrix1=[[int(input()) for j in range(2)] for i in range(2)]  
print("Input elements of matrix2")  
matrix2=[[int(input()) for j in range(2)] for i in range(2)]  
matrix3=[[matrix1[i][j]+matrix2[i][j] for j in range(2)] for i in range(2)]
```

```
print(matrix1)  
print(matrix2)  
print(matrix3)
```

Example

```
grade_list=[['naresh','A'],  
             ['suresh','B'],['kishore','A'],['ramesh','B'],  
             ['rajesh','C'],['kiran','A']]
```

```
print(grade_list)
grade_listA=[stud for stud in grade_list if stud[-1]=='A']
grade_listB=[stud for stud in grade_list if stud[-1]=='B']
grade_listC=[stud for stud in grade_list if stud[-1]=='C']

print(grade_listA,grade_listB,grade_listC,sep="\n")
```

Output

```
[['naresh', 'A'], ['suresh', 'B'], ['kishore', 'A'], ['ramesh', 'B'], ['rajesh', 'C'],
['kiran', 'A']]
[['naresh', 'A'], ['kishore', 'A'], ['kiran', 'A']]
[['suresh', 'B'], ['ramesh', 'B']]
[['rajesh', 'C']]
```

Example

```
names_list=['naresh','ramesh','kishore','rajesh','kiran','raman']
print(names_list)
names_list1=[name for name in names_list if name[-1]=='h']
names_list2=[name for name in names_list if name[0]=='k']
names_list3=[name for name in names_list if name[0] in "nk" and
name[-1] in "hn"]
print(names_list1,names_list2,names_list3,sep="\n")
```

Output

```
['naresh', 'ramesh', 'kishore', 'rajesh', 'kiran', 'raman']
['naresh', 'ramesh', 'rajesh']
['kishore', 'kiran']
['naresh', 'kiran']
```

Example:

```
list1=[1,2,3,4,5,6,7,8,9,10,12,45,87,21,34,65,78,34,98,56,45,32,67,23]
print(list1)
even_list=[num for num in list1 if num%2==0]
print(even_list)
odd_list=[num for num in list1 if num%2!=0]
```

```
print(odd_list)
```

Output

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 45, 87, 21, 34, 65, 78, 34, 98, 56, 45, 32, 67, 23]
[2, 4, 6, 8, 10, 12, 34, 78, 34, 98, 56, 32]
[1, 3, 5, 7, 9, 45, 87, 21, 65, 45, 67, 23]
```

Tuple

Tuple is an immutable sequence data type.

After creating tuple changes cannot be done. Tuple does not support mutable operations like append, insert, remove, del,

Where tuples are used?

1. To represent immutable list
2. In application development tuple is used to represent data for other collections like sets, dictionaries

```
>>> list1=[10,20,30,40]
```

```
>>> t1=(10,20,30,40)
```

```
>>> print(list1)
```

```
[10, 20, 30, 40]
```

```
>>> print(t1)
```

```
(10, 20, 30, 40)
```

```
>>> list1[0]=99
```

```
>>> print(list1)
```

```
[99, 20, 30, 40]
```

```
>>> t1[0]=99
```

Traceback (most recent call last):

File "<pyshell#17>", line 1, in <module>

t1[0]=99

TypeError: 'tuple' object does not support item assignment

```
del list1[0]
```

```
print(list1)
```

```
[20, 30, 40]
```

```
>>> del t1[0]
```

```
Traceback (most recent call last):
```

```
File "<pyshell#20>", line 1, in <module>
```

```
del t1[0]
```

```
TypeError: 'tuple' object doesn't support item deletion
```

```
>>> list1.append(10)
```

```
>>> t1.append(10)
```

```
Traceback (most recent call last):
```

```
File "<pyshell#22>", line 1, in <module>
```

```
t1.append(10)
```

```
AttributeError: 'tuple' object has no attribute 'append'
```