

List Comprehension



Session Objectives



- Concept
- Simple List Comprehension
- Conditional using Tuple
- Conditional using Dictionary
- Summary

NubeEra



CORPORATE



RECORDED



INTERACTIVE



• Concept

- List comprehension offers a shorter syntax when you want to create a new list based on the values of an existing list.
- With list comprehension you can do all that with only one line of code
- **Syntax:**
- *newList = [expression(element) **for** element **in** oldList **if** condition]*
- Example
 - cities = ["mumbai", "pune", "jaipur", "nagpur", "panji"]
 - newList = [x **for** x **in** cities]
 - print(newlist)





● Example(Without list comprehension)

- Based on a list of cites, you want a new list, containing only the cites with the letter "n" in the name.
- Without list comprehension you will have to write a **for** statement with a conditional test inside:

```
cities = ["mumbai", "pune", "jaipur", "nagpur", "panji"]
newlist = []
for x in cities:
    if "n" in x:
        newlist.append(x)
print(newlist)                                # [ 'pune' , 'nagpur' , 'panji' ]
```

NubeEra



CORPORATE



RECORDED



INTERACTIVE

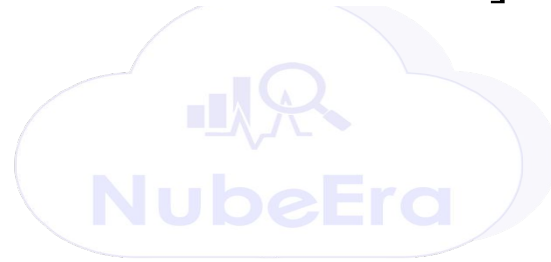


- Example(**With Conditional list comprehension**)

- cities = ["mumbai", "pune", "jaipur", "nagpur", "panji"]

```
newlist = [x for x in cities if "n" in x]
```

```
print(newlist)
```



Output:

```
[ 'pune' , 'nagpur' , 'panji' ]
```



CORPORATE



RECORDED



INTERACTIVE



• Conditional using Tuple

- `cities = ("mumbai", "pune", "jaipur", "nagpur", "panji")`

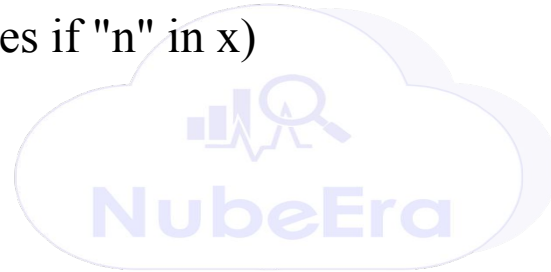
```
print(cities)
```

```
newtuple = (x for x in cities if "n" in x)
```

```
print(newtuple)
```

Output:

`('pune' , 'nagpur' , 'panji')`



CORPORATE



RECORDED



INTERACTIVE



● Conditional comprehension using Dictionary

- Dictionary comprehension is a method for transforming one dictionary into another dictionary. During this transformation.
- Items within the original dictionary can be conditionally included in the new dictionary and each item can be transformed as needed.

■ `dict1 = {'a': 1, 'b': 2, 'c': 3, 'd': 4, 'e': 5}`

```
double_dict1 = {k:v*2 for (k,v) in dict1.items()}    #double each value  
print(double_dict1)
```

Output: {'e': 10, 'a': 2, 'c': 6, 'b': 4, 'd': 8}





- Concept
- Simple List Comprehension
- Conditional using Tuple
- Conditional using Dictionary
- [Demo](#)

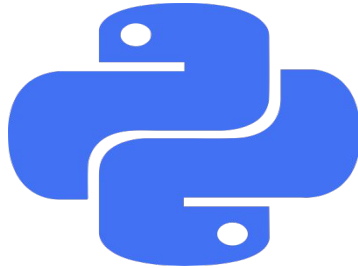
NubeEra



???

The Important thing is not to
stop

Questioning



RECORDED



CORPORATE



INTERACTIVE



CORPORATE



RECORDED



INTERACTIVE