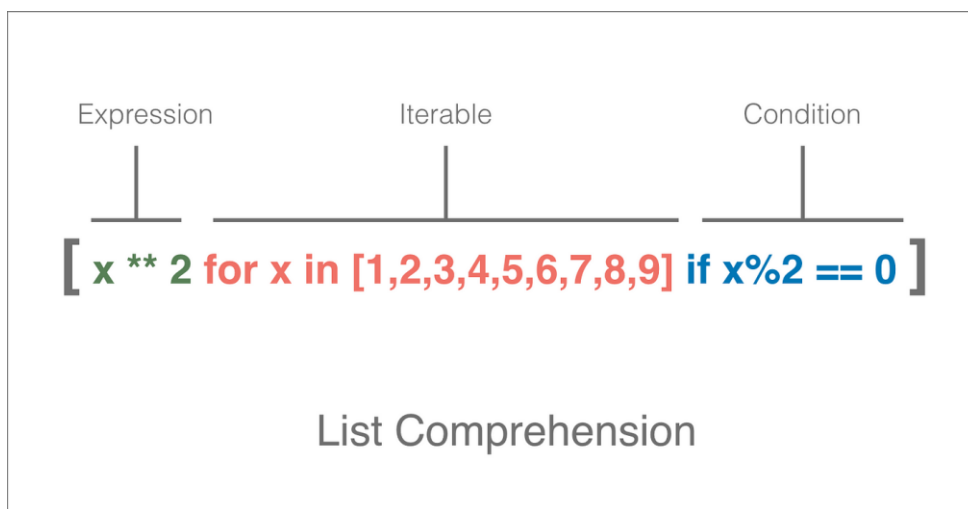


Training Report Day-12

19 June 2024

LIST COMPREHENSION

- List Comprehensions provide an elegant way to create new lists.
- It consists of brackets containing an expression followed by a for clause, then zero or more for or if clauses.



Example:-

```
mystring = "WELCOME"
mylist = [ i for i in mystring ] # Iterating through a string Using List Comprehension
mylist
```

```
l1 = []
for i in mystring:
    l1.append(i)
print(l1)
```

```
l2 = [num**2 for num in range(10) if num%2==0]
print(l2)
```

```
mylist1 = [ i for i in range(40) if i % 2 == 0] # Display all even numbers between 0 and 40
mylist1
```

```
# Multiple whole list by 10
```

```
list1 = [2,3,4,5,6,7,8]
```

```
list1 = [i*10 for i in list1]
```

```
list1
```

```
# Extract letters from a string
```

```
mystr = "One 1 two 2 three 3 four 4 five 5 six 6789"
```

```
numbers = [i for i in mystr if i.isalpha()]
```

```
numbers
```

DICTIONARY COMPREHENSION

```
{ key: value for vars in iterable }
```

```
{ num: num*num for num in range(1, 11) }
```

Example:-

```
double = {i:i*2 for i in range(10)} #double each value using dict comprehension
```

```
double
```

```
square = {i:i**2 for i in range(10)}
```

```
square
```

```
key = ['one', 'two', 'three', 'four', 'five']
```

```
value = [1,2,3,4,5]
mydict = {k:v for (k,v) in zip(key,value)} # using dict comprehension
mydict
```

```
str1 = "Natural Language Processing"
mydict2 = {k:v for (k,v) in enumerate(str1)} # Store enumerated values in a dictionary
mydict2
```

```
str1 = "abcdefghijklmnopqrstuvwxyz"
mydict3 = {i:i.upper() for i in str1} # Lower to Upper Case
mydict3
```