for loops and List Comprehension

By : Ramkrushna Pradhan

in **keyword**

- General format :

  x in *list*

  returns true or false

  Example :

  ```
  >>> 5 in [6,3,2,6,8,5]
  >>> False
  ```

is **keyword**

- General format :

  x is *y*

  returns true or false

  Example :

  ```
  >>> a = 5
  >>> a is 5
  >>> True
  ```

for **loop**

● General format :

for x in list:

{do this}

{and this}

...

for loop

- Sum of elements in a list

```
a = [1,5,2,3,7]
s = 0
for i in a:
    s += i
print s
```

for loop and Strings

- Print every character in newline

```
s = "Hello"
for i in s:
    print i
```

for loops on 2D lists

- ```
  a = [ [1,2] , [3,6], [6,9] ]
  for i in a:
      print i
  ```
- ```
  a = [ [1,2] , [3,6], [6,9] ]
  for i,j in a:
      print i,j
  ```

some functions

```
>>> print range(10)
>>> [0,1,2,3,4,5,6,7,8,9]


>>> a = ['a','b','c']
>>> for i,j in enumerate(a):
        print i,j
```

list comprehension

- Dynamically create and manipulate lists
- Usage,

  new_list = [expr(*item)* for *item* in list]
- Using with if,

  new_list = [expr(*item)* for *item* in list **if**

  condition(item)]

list comprehension

- nested for loop,

  [expr(i,j) **for** i in A **for** j in B]

- if else,

  [e1(*item)* **if condition(item)** else e2(*item)* for *item* in list]

list comprehension

- Square each element of a list,

  ```
  a = [1,2,3,4,5]

  [i**2 for i in a]
  ```

- Check if elements are even or odd,

  ```
  a = [2,3,6,5,1,4,8]

  [i%2==0 for i in a]
  ```

list comprehension

- Print if an element is even or odd,

```
a = [2,3,6,5,4,1,5,8,9,11]
['even' if i%2==0 else 'odd' for i in a]
```

Assignments - ½ (list comprehension is must)

- Input two lists, five elements each. And print the element wise sum of those lists.
- Input a list of 10 integers, and print the sum of numbers greater than 10.
- Duplicate each character in a string. For example, for input "apple", output should be "aappppllee".

Assignments - 2/2

- Input an integer , and print the factorial of that number.
- Input two strings, and print 'True' if the smaller string appears in the bigger one. Also print the start index and end index i.e. where the samll strings starts and ends in the bigger string.