

Exercise Set 1

Hina Arora

- **list comprehensions** are used to create a new list by applying simple operations to an existing list
- **lambda functions** are used to define and use small anonymous functions inline
- **map functions** are used to apply a function to each item in an iterable
- **filter functions** are used to apply a function to each item in an iterable, and return only those elements that meet some criteria

Question

Use list comprehension to convert a list of numbers to a list of absolute values of those numbers.

```
nums = [100, -200, -300, 400, 500]
```

```
Ans: [100, 200, 300, 400, 500]
```

Answer

```
nums = [100, -200, -300, 400, 500]  
numsabs = [abs(n)  
            for n in nums]  
numsabs
```

Question

Use list comprehension to get a list of first characters from a list of words.

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
```

```
Ans: ['a', 'b', 'g', 'o', 'p']
```

Answer

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
```

```
word0 = [w[0]
```

```
        for w in words]
```

```
word0
```

Question

Use list comprehension to sum corresponding elements of two lists.

```
lst1 = [1, 2, 3, 4, 5]
```

```
lst2 = [10, 20, 30, 40, 50]
```

```
Ans: [11, 22, 33, 44, 55]
```

Answer

```
lst1 = [1, 2, 3, 4, 5]
```

```
lst2 = [10, 20, 30, 40, 50]
```

```
lst = [l1+l2
```

```
      for l1,l2 in zip(lst1, lst2)]
```

```
lst
```


Question

Use list comprehension to get a list of words ≤ 5 characters long from a list of words.

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
```

```
Ans: ['apple', 'grape']
```

Answer

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
wordl5 = [w
          for w in words
          if len(w)<=5]

wordl5
```

Question

Create a lambda function to return True if a number is even, else return False. Then use it with the number 5.

Ans: False

Answer

```
(lambda x: x%2==0)(5)
```

Question

Use map and lambda to return a new list with 0 or 1 depending on whether the elements in the existing list are ≥ 100 or < 100 .

lst = [100, 50, 75, 200, 25]

Ans: [0, 1, 1, 0, 1]

Answer

```
lst = [100, 50, 75, 200, 25]
newlst = list(
    map(
        (lambda x: 1 if x<100 else 0),
        lst
    )
)
newlst
```

Question

Use map to convert a list of numbers to a list of absolute values of those numbers.

```
nums = [100, -200, -300, 400, 500]
```

```
Ans: [100, 200, 300, 400, 500]
```

Answer

```
nums = [100, -200, -300, 400, 500]  
numsabs = list(map(abs, nums))  
numsabs
```


Question

Use map to lower case items in a list.

```
words = ["APPLE", "Banana", "Grape", "ORANGE", "pineapple"]
```

```
Ans: ['apple', 'banana', 'grape', 'orange', 'pineapple']
```

Answer

```
words = ["APPLE", "Banana", "Grape", "ORANGE", "pineapple"]  
wordslow = list(map(str.lower, words))  
wordslow
```

Question

Use map and lambda to sum corresponding elements of two lists.

```
lst1 = [1, 2, 3, 4, 5]
```

```
lst2 = [10, 20, 30, 40, 50]
```

```
Ans: [11, 22, 33, 44, 55]
```

Answer

```
lst1 = [1, 2, 3, 4, 5]
```

```
lst2 = [10, 20, 30, 40, 50]
```

```
lst = list(  
    map(  
        (lambda l1,l2: l1+l2),  
        lst1, lst2  
    )  
)
```

```
lst
```

Question

What will this return?

```
lst = [10, 20, 30, 40, 50]
```

```
newlst = list(map((lambda x: x>25), lst))
```

```
newlst
```

Answer

[False, False, True, True, True]

Question

What will this return?

```
lst = [10, 20, 30, 40, 50]
```

```
newlst = list(filter((lambda x: x>25), lst))
```

```
newlst
```

Answer

[30, 40, 50]

Question

Use filter and lambda functions to get a list of words ≤ 5 characters long from a list of words.

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
```

```
Ans: ['apple', 'grape']
```

Answer

```
words = ["apple", "banana", "grape", "orange", "pineapple"]
wordl5 = list(
    filter(
        (lambda x: len(x)<=5),
        words
    )
)
wordl5
```

Question

Use filter and lambda functions to get even elements from a list of integers.

```
nums = [10, 13, 14, 15, 17, 20]
```

```
Ans: [10, 14, 20]
```

Answer

```
nums = [10, 13, 14, 15, 17, 20]
numseven = list(
    filter(
        (lambda x: x%2==0),
        nums
    )
)
numseven
```