

UNIVERSITY *of* WISCONSIN
LA CROSSE
COMPUTER SCIENCE

CS 224 Introduction to Python

List Comprehension Exercises

1

Exercise 1:

Use a list comprehension to create a list of Fahrenheit temperatures from a list of Centigrade temperatures.

```
dF = [1.8 * c + 32 for c in dC]
```

What does this comprehension do:

```
dF = [1.8 * c + 32 for c in [randint(0, 100)  
                             for i in range(10)]]
```

2

Exercise 2:

Use a list comprehension to create a list of the odd values in a list of data.

```
odds = [i for i in data if i % 2 == 1]
```

3

Using a function

Let coords be a list containing 2-element lists of GPS coordinates:

```
[[[lat1, lon1], [lat2, lon2]], [[lat3, lon3], [lat4, lon4]] ...]
```

Create a list of distances between the pair of cities in each sublist.

```
def distance(city1, city2): ← Function definition  
    # compute and return distance
```

```
dists = [distance(x, y) for x, y in coords]
```

4

Exercise 3:

Staying with the setup from the previous slide, create a list of the pairs of cities that are closer than threshold d :

```
close = [[x, y] for x, y in coords if distance(x, y) < d]
```

function in the filter



5

Exercise 4:

Let `data` be a list containing lists of instrument readings:

```
[[d0_0, d0_1, ... d0_n], [d1_0, d1_1, ... d1_m] ...]
```

Write a list comprehension that creates a list of the mean values of each of the sublists in `data`:

```
means = [sum(L)/len(L) for L in data if len(L) > 0]
```

Why is the filter necessary?

6

Exercise 5:

Let `chars` be a list of unique characters. Write a list comprehension that uses `chars` to create a list of 3-letter “words” such that no two letters in a word are the same.

Without the restriction of no repeated characters:

```
words = [x+y+z for x in chars for y in chars
         for z in chars]
```

This version eliminates repeated characters:

```
words = [x+y+z for x in chars for y in chars
         for z in chars if x!=y and y!=z
         and x!=z]
```

7

Exercise 6:

Write a list comprehension that uses a list of random integers in the range 1..100 to create a list of those integers in the list that contain a 7. You are not given the list of random integers – you must create it in the comprehension.

```
sevens = [x for x in [randint(1,100) for _ in
                    range(20)] if '7' in str(x)]
```

8

Exercise 7:

Let `L1`, `L2`, and `L3` be lists of `n` integers. Write a list comprehension that creates list of integers `L4` such that `L4[i]` is: `L1[i]` if `i % 3` is 0, `L2[i]` if `i % 3` is 1, or `L3[i]` if `i % 3` is 2.

```
L4 = [L1[i] if i%3==0 else L2[i] if i%3==1 else
      L3[i] for i in range(len(L1))]
```

9

Exercise 8:

Let `L1` be a list of random integers in the range 100 to 10000. Write a list comprehension to create a list of integers `L2` from `L1` that do not contain a '4' in the second position. For example, 234 would be in `L2` while 345 would not.

```
L2 = [x for x in [y for y in L1 if list(str(y))[1] != '4']]
```

10