CSIT110 / CSIT810

Python

Lecture 9

Dr. Joseph Tonien

School of Computing and Information Technology
University of Wollongong

Objectives

Understanding of:

- List
- List manipulation

List

- used to hold a list of items

```
list1 = [] # this is an empty list

list2 = [1, 4, 4, 10, -1] # this list has 5 items
```

- items in a list can have different data types

```
list3 = [1, "hi there", True, 10, "john"]
```

- list items can be added, removed and modified

```
list1.append("hi")      # now list1 = ["hi"]
list1.append(100)       # now list1 = ["hi", 100]

list2[4] = "cat"        # now list2 = [1, 4, 4, 10, "cat"]
del list3[1]            # now list3 = [1, True, 10, "john"]
```

List

```
list1 = []

list2 = [1, 4, 4, 10, -1]

list3 = [1, "hi there", True, 10, "john"]
```

using function `print` to print out the whole list

```
print(list1)

print(list2)

print(list3)
```

List

```
list1 = []

list2 = [1, 4, 4, 10, -1]

list3 = [1, "hi there", True, 10, "john"]
```

using `len` to find out how many items in the list:

```
list1_length = len(list1)  # ⟶ 0

list2_length = len(list2)  # ⟶ 5

list3_length = len(list3)  # ⟶ 5

print("list3 has {0} items".format(list3_length))
print("list3 has {0} items".format(len(list3)))
```

List

```
list3 = [1, "hi there", True, 10, "john"]
```

List items can be accessed using index, starting from index 0

```
print(list3[0])  # → 1

print(list3[1])  # → "hi there"

print(list3[2])  # → True

print(list3[3])  # → 10

print(list3[4])  # → "john"
```

Note that `len(list3)` is 5, i.e. the list has 5 items, but the **last index** is 4.

List

```
list3 = [1, "hi there", True, 10, "john"]
```

Note that `len(list3)` is 5, i.e. the list has 5 items, but the **last index** is 4.

Therefore, we can use **for-loop** to print out all items:

```
for i in range(0, len(list3)):
  print(list3[i])
```

List

```
list3 = [1, "hi there", True, 10, "john"]
```

Using index, we can also change the items:

```
list3[0] = 3  # now list3 = [3, "hi there", True, 10, "john"]

list3[1] = 4  # now list3 = [3, 4, True, 10, "john"]

list3[2] = 5  # now list3 = [3, 4, 5, 10, "john"]

list3[3] = 6  # now list3 = [3, 4, 5, 6, "john"]

list3[4] = 7  # now list3 = [3, 4, 5, 6, 7]
```

List

```
list2 = [1, 4, 4, 10, -1]
```

Using for-loop, increase each item by 10:

```
for i in range(0, len(list2)):
  list2[i] = list2[i] + 10

print(list2)    # → [11, 14, 14, 20, 9]
```

List

```
list1 = []
list2 = [1, 4, 4, 10, -1]
```

items can be appended to the end of the list:

```
list1.append("hi") # now list1 = ["hi"]
list1.append(100)  # now list1 = ["hi", 100]

list2.append("hi") # now list2 = [1, 4, 4, 10, -1, "hi"]
list2.append(100)  # now list2 = [1, 4, 4, 10, -1, "hi", 100]
```

List

```
list4 = [10, 7, 5]
```

items can be inserted into the list:

```
list4.insert(1, "a") # now list4 = [10, "a", 7, 5]
Insert at index 1: we can check now that list4[1] = "a"


list4.insert(3, "b") # now list4 = [10, "a", 7, "b", 5]
Insert at index 3: we can check now that list4[3] = "b"


list4.insert(5, "c") # now list4 = [10, "a", 7, "b", 5, "c"]
Insert at index 5: we can check now that list4[5] = "c"
```

List

```
list3 = [1, "hi there", True, 10, "john"]
```

items can be deleted from the list:

```
# Deleting the item at index 1
del list3[1]   # now list3 = [1, True, 10, "john"]


# Deleting the item at index 2
del list3[2]   # now list3 = [1, True, "john"]


# Deleting the item at index 0
del list3[0]   # now list3 = [True, "john"]


# Deleting the item at index 1
del list3[1]   # now list3 = [True]

# Deleting the item at index 0
del list3[0]   # now list3 = []
```

List

```
list2 = [1, 4, 4, 10, -1]
```

remove an item in the list
(only the first appearance)

```
list2.remove(4)   # now list2 = [1, 4, 10, -1]
# this is the same as: del list2[1]

list2.remove(10)  # now list2 = [1, 4, -1]
# this is the same as: del list2[2]

list2.remove(5)   # ValueError: 5 is not in the list
```

List

```
list2 = [1, 4, 4, 10, -1]
```

count how many an item appears in the list

```
four_count = list2.count(4)   # → 2

ten_count = list2.count(10)   # → 1

five_count = list2.count(5)   # → 0
```

find the lowest index of an item in the list

```
four_index = list2.index(4)   # → 1

ten_index = list2.index(10)   # → 3

five_index = list2.index(5)   # ValueError: 5 is not in the list
```

List

```
list2 = [1, 4, 4, 10, -1]
```

finding min item

```
list2_min = min(list2)  # ⟶ -1
```

finding max item

```
list2_max = max(list2)  # ⟶ 10
```

List

```
list2 = [1, 4, 4, 10, -1]
```

Sorting a list and return a new list, original list is unchanged

```
list2b = sorted(list2)
```

now list2b = [-1, 1, 4, 4, 10]
but list2 is **unchanged:** list2 = [1, 4, 4, 10, -1]

Sorting a list and modify the original list

```
list2.short()
```

now list2 is **changed**, list2 = [-1, 1, 4, 4, 10]

List

```
list2 = [1, 4, 4, 10, -1]
```

items can be reversed

```
list2.reverse()   # now list2 = [-1, 10, 4, 4, 1]
```

remove all items

```
list2.clear()     # now list2 = []
```

List

```
list2 = [1, 4, 4, 10, -1]
list4 = [10, 7, 5]
```

adding two lists

```
list5 = list2 + list4
# now list5 = [1, 4, 4, 10, -1, 10, 7, 5]

list6 = list4 + list2
# now list6 = [10, 7, 5, 1, 4, 4, 10, -1]
```

multiply a list

```
list7 = list4 * 3
# now list7 = [10, 7, 5, 10, 7, 5, 10, 7, 5]
```

List

```
list2 = [1, 4, 4, 10, -1]
```

Slicing a list

```
list5 = list2[1:4]
# now list5 = [4, 4, 10]
```

list[i:j] gives items from index i up to index (j-1), so altogether, there are (j-i) items

```
list5 = list2[1:]
# now list5 = [4, 4, 10, -1]
```

list[i:] gives items from index i up to the end

```
list5 = list2[:4]
# now list5 = [1, 4, 4, 10]
```

list[:j] gives items from index 0 up to index (j-1), so altogether, there are j items

Square sequence
0, 1, 4, 9, 16, 25, ...

Create a list and put the first 10 squares into the list

```
square = []

for i in range(0, 10):
  square.append(i*i)

print(square)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

Fibonacci sequence

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13, ...

Create a list and put the first 5 fibonacci numbers into the list

```
fibo = []
print(fibo)

fibo.append(0)
print(fibo)

fibo.append(1)
print(fibo)

for i in range(2, 5):
  f = fibo[i-1] + fibo[i-2]
  fibo.append(f)
  print(fibo)
```