Introduction to Computation for the Humanities and Social Sciences

CS 3
Chris Tanner
Lecture 12

Ready, Set, Go!
Lecture 12

- My Project Demo
- New Data Structure: Set()
- Scope
Sets

New Data Structure!

• very similar to Lists

• A Set is an unordered collection of items

• Collections which don’t preserve the order in which elements are added are called unordered

• Doesn’t support indexing

```python
a = {1,2,3}
a.add(4)
if 2 in a:
    print(a)
```
Sets

New Data Structure!

- very similar to **Lists**, but
- is incapable of storing duplicate items (maintains just 1 copy)

```python
students = set()
students.add("jackie")
students.add("emily")
students.add("hank")
students.add("emily")
print(students)

>>> students = {'emily', 'jackie', 'hank'}
```
New Data Structure!

- checking if an item is contained in the Set happens instantly (doesn’t need to check each item one by one)

```python
students = set()
students.add("jackie")
students.add("emily")
students.add("hank")
students.add("emily")

if cur_student in students:
    print(student + " is present!")
else:
    print(student + " is NOT present!")
```
Sets

New Data Structure!

• The downside: Sets have no indices and no order!

• So, you can’t access specific items. If you need to maintain an order for your items, use a List.

```python
students = set()
students.add("jackie")
students.add("emily")
students.add("hank")
students.add("emily")
students[0]  # CRASHES
```
When to use a `Set()`?

- Use a `Set()` when you only care about storing a collection of unique values (i.e., don’t care about duplicates) and the order doesn’t matter to you.

- e.g., store license plate #s or list of senator names.
What is Scope?

• Each variable has its own scope.

• Scope refers to the places in your code which has access to the variable.

• You’ve encountered scope before. Where?
What is Scope?

- A variable’s scope starts when it’s first declared/initialized.

- A variable’s scope terminates once your code has less indentation than that of when it was declared.
What is Scope?

• The concept we saw with the scope within functions continues on a smaller scale, too: if we define a variable within a for-loop or if-statement, it’s like its own private room, and things outside the room shouldn’t try to access these variables.

• Variables inside the room (i.e., inside the for-loop or if-statement) can access variables that were declared outside of it — as long as they were executed earlier.
In Summary

- It’s okay to access/use/update a variable if it’s:
  - your code already executed the line that declares/initializes it
  - the variable was declared with indentation that’s either (1) equal or (2) less than the line in question.

Otherwise, you can’t rely on the variable’s value, and often Python won’t even let you run such code.
names = ["malik", "stephanie", "ellie", "rico"]

for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)
```python
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list's length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
```
Scope

```python
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
```
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list's length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ['malik', 'stephanie', 'ellie', 'rico']
for name in names:
    print("list's length:" + str(len(names)))
    if name[0] == 'e':
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]

for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]

for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]

for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print(“list’s length:” + str(len(names)))
    if name[0] == “e”:
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False
print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]

for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)
names = ["malik", "stephanie", "ellie", "rico"]
for name in names:
    print("list’s length:" + str(len(names)))
    if name[0] == "e":
        starts_with_e = True
    else:
        starts_with_e = False

print(starts_with_e)
print(names)