THE UNIVERSITY OF MELBOURNE

COMP10001

Foundations of Computing

Semester 1, 2021

Tutorial 4

Andrew Naughton

andrew.naughton@unimelb.edu.au

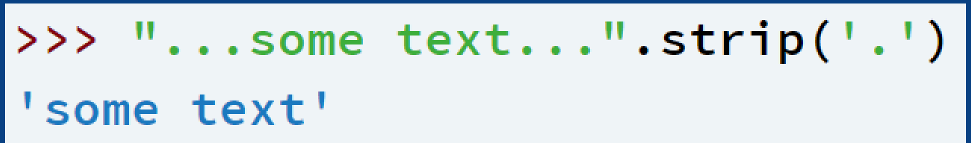
Outline

- Functions & Methods
- Lists & Tuples
- Mutability
- Iteration
- for vs. while
- Exercises

Functions & Methods

- Similarities:
  - Run some pre-defined code to achieve a task
  - Get called with brackets which can contain arguments

- Differences:
  - Methods are “attached” to an object, with a dot after the object name
  - Methods can edit the object they are called on
    - E.g.

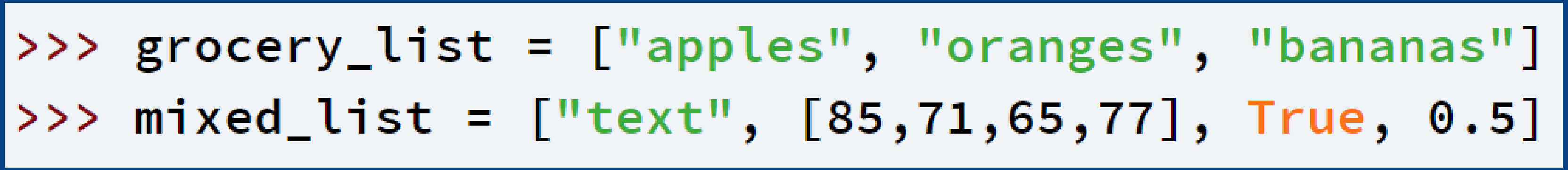


```
>>> "...some text...".strip('.')
'some text'
```

THE UNIVERSITY OF
MELBOURNE

Lists

- ❖ Data type that stores objects in an ordered sequence
- ❖ Defined with square brackets, and commas separating the objects/elements
- ❖ E.g.



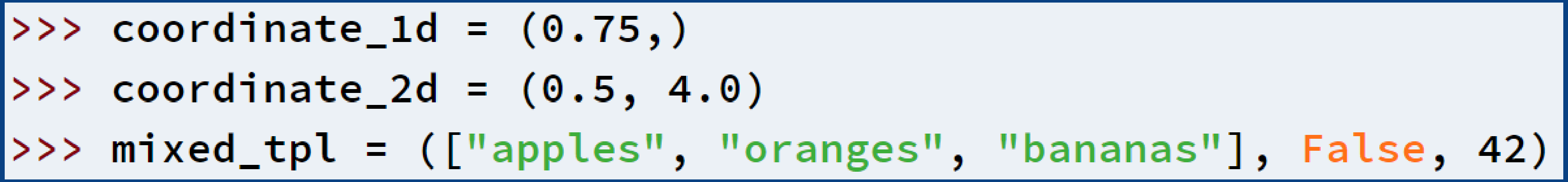
```
>>> grocery_list = ["apples", "oranges", "bananas"]
>>> mixed_list = ["text", [85,71,65,77], True, 0.5]
```

Lists

- Useful methods
  - .append()
    - Adds a single element to the end of the list
  - .pop()
    - Removes the element at a specified index (-1 if unspecified)
    - Returns the removed element
  - .sort()
    - Sorts the list in place
  - .copy()
    - Makes a copy of the list
    - Returns a new list

Tuples

- Data type that stores objects in an ordered sequence
- Defined with brackets, and commas separating the objects/elements
- Commonly used to group related objects together, e.g. a set of coordinates (x, y)
- E.g.

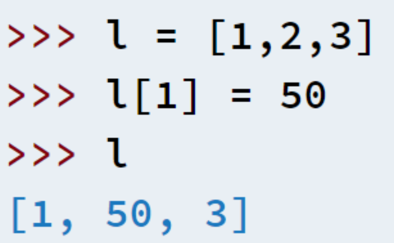


```
>>> coordinate_1d = (0.75,)
>>> coordinate_2d = (0.5, 4.0)
>>> mixed_tpl = (["apples", "oranges", "bananas"], False, 42)
```

THE UNIVERSITY OF
MELBOURNE

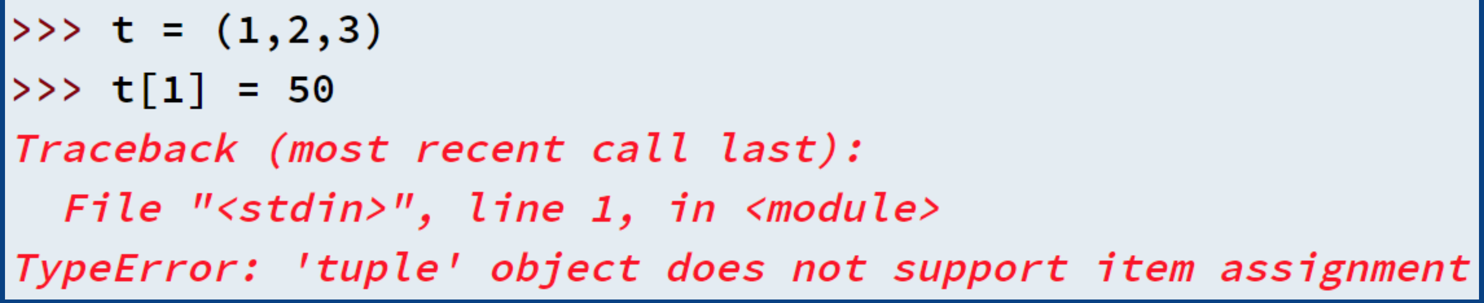
Mutability

- Lists are mutable
  - After we define it, we can change its contents



```
>>> l = [1,2,3]
>>> l[1] = 50
>>> l
[1, 50, 3]
```

- Tuples are immutable
  - After we define it, we cannot change its contents



```
>>> t = (1,2,3)
>>> t[1] = 50
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
```

THE UNIVERSITY OF
MELBOURNE

Mutability

- Mutable data types
  - list
  - set
  - dict
- Immutable data types
  - bool
  - int
  - float
  - str
  - tuple

Iteration

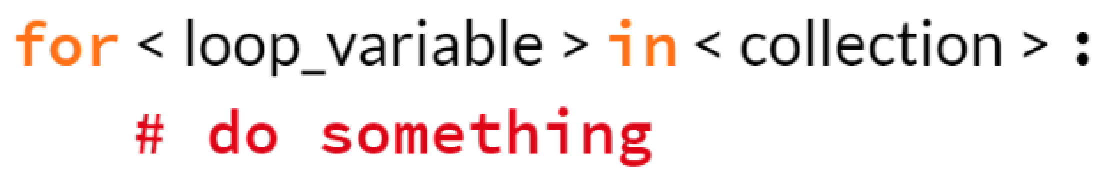
- The process of executing a block of code repeatedly (often with a slight difference each time)
- Allows us to avoid writing the same instructions many times over

- In Python, iteration is achieved through for loops and while loops

for Loop

- Iterates over a collection of items

- Skeleton

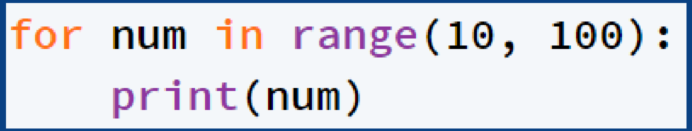


```
for < loop_variable > in < collection > :
    # do something
```

- Use when
  - You know in advance how many times you want to iterate

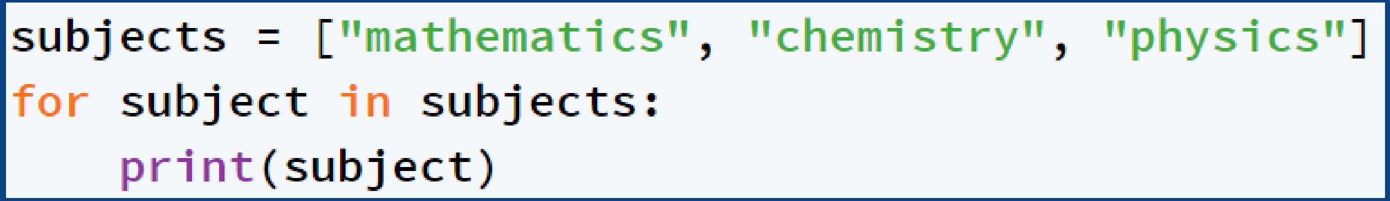
for Loop

- E.g.
  - Print the numbers from 10 to 99



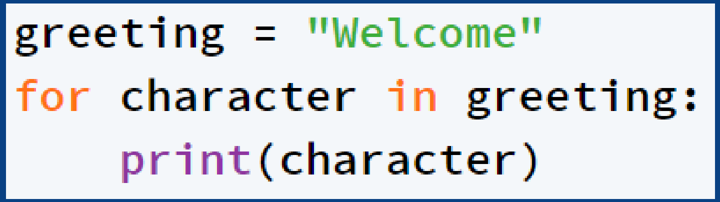
```python
for num in range(10, 100):
    print(num)
```

  - Print the elements in the list (or tuple)



```python
subjects = ["mathematics", "chemistry", "physics"]
for subject in subjects:
    print(subject)
```

  - Print the characters in a string



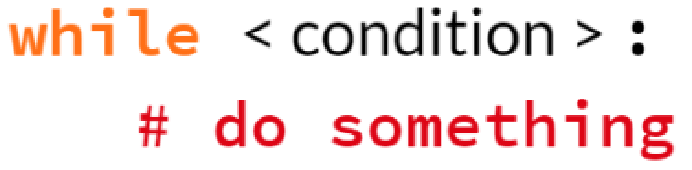
```python
greeting = "Welcome"
for character in greeting:
    print(character)
```

THE UNIVERSITY OF
MELBOURNE

while Loop

- Runs as long as some condition is True

- Skeleton



```
while < condition > :
    # do something
```

- Use when
  - The number of iterations is unknown ahead of time
- Example of use
  - Keep asking the user to guess the correct number between 0 and 10

THE UNIVERSITY OF
MELBOURNE

for vs. while

- We can convert any for loop into a while loop by making use of some additional variables
- Most while loops can be converted into for loops, however not all
- E.g. a while loop that repeats itself indefinitely until an action is made by the user (guesses the correct number)

THE UNIVERSITY OF
MELBOURNE

Exercises