Object-oriented programming

Computing & Information Sciences

W. H. Bell

Classes and objects

University of Strathclyde Glasgow

- Class can contain:
  - Data members.
  - Member functions.
- An object is an instance of a class.
  - Similar to a variable being an instance of an integer.
  - Objects are normally mutable.

Single data member

```
class MyClass:
    def __init__(self, name):
        self.name = name

m1 = MyClass("A new object")
m2 = MyClass("A new object")
m1.name = "Updated name"
print("m1.name = " + m1.name)
print("m2.name = " + m2.name)
```

Output

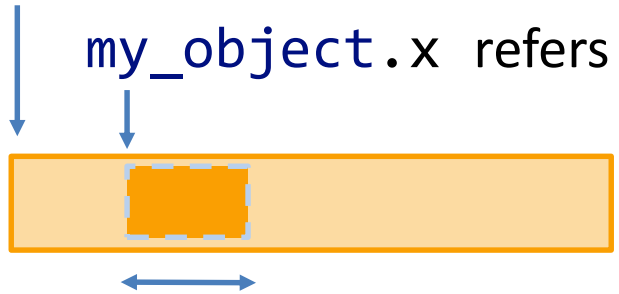
```
m1.name = Updated name
m2.name = A new object
```

Objects and data members

```
my_object = MyClass()      # Create object and assign it.
print(my_object.x)         # Use data member and print it.
```

`my_object` refers to start of memory

`my_object.x` refers to position within object



Size of memory set by type of data member

Size of memory set by Class

Single member function

```
class MyClass:
    def __init__(self):
        self.name = "MyClass"

    def full_name(self):
        return self.name + " is an example"

m = MyClass()
m.name = "New name"
print("full_name = " + m.full_name())
```

Output

```
full_name = New name is an example
```

Data classes

University of Strathclyde Glasgow

- Read/write to input/output.
  - Often referred to as the “data model”.
- Member functions to return transient data.
  - Calculate simple features from stored values.
  - E.g. angle from coordinates.
- Limit functionality of member functions.

Algorithm classes

University of Strathclyde Glasgow

- Collection of functions.
  - Perform operations on input data – data objects.
  - May contain configuration settings.

- A class might not be needed.
  - Avoid classes with collection of static functions (Java).
  - Python module might be a better choice than a class.

Inheritance

University of Strathclyde Glasgow

- Classes can inherit from another class.
  - Data members and functions become part of derived class.
- A derived class can directly use member functions and data if they are public or protected.
  - Private member functions and data cannot be directly accessed by derived classes.
- Only use inheritance if absolutely necessary.

Inheritance

```
class MapPosition:
    def __init__(self):
        self.latitude = 0.
        self.longitude = 0.

class InclinedPosition(MapPosition):
    def __init__(self):
        self.elevation = 0.

m = MapPosition()   # Create a position
m.latitude = 13.0
m.longitude = -10.0
p = InclinedPosition() # Create an inclined position
p.latitude = 55.860916
p.longitude = -4.251433
p.elevation = 16
```

University of Strathclyde Glasgow