Lecture 18: object-oriented programming

Review of Object-Oriented Programming in Python

The primary characteristics associated with object-oriented programming are

- inheritance;
- encapsulation; and
- polymorphism

Inheritance

```
class Shape:

class Rectangle(Shape):

class Square(Rectangle):
```

Encapsulation

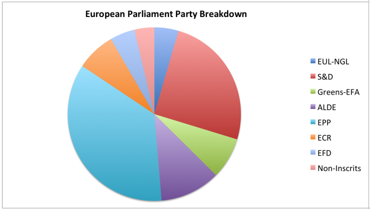
```
class Shape:

class Rectangle(Shape):

  def __init__(self, width, height):
    self._width = width
    self._height = height
```

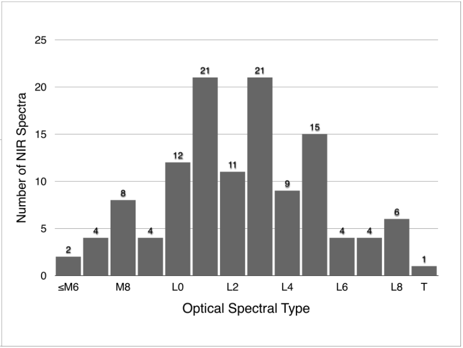
Polymorphism

```
class Shape:
    def area(self):
        pass


class Rectangle(Shape):

    def area(self):
        return self._width * self._height

class Square(Rectangle):

    def __init__(self, side):
        super().__init__(side, side)
```

```
>>> shape = Rectangle(10,20)
>>> shape.area()
200
>>> shape = Square(10)
>>> shape.area()
100
```

Modelling Charts





A Chart class

```
class Chart:

    def __init__(self, title):
        self._title = title

    def title(self):
        return self._title

    def __str__(self):
        return "{}".format(self._title)
```

A Histogram class

```
class Histogram(Chart):

    def __init__(self, bins, title):
        self._bins = bins
        self._counts = [0]*len(self._bins)
        super().__init__(title)

    def _index(self, bin):
        return self._bins.index(bin)

    def add_to_bin(self, bin, count):
        self._counts[self._index(bin)] += count

    def count(self, bin):
        return self._counts[self._index(bin)]

    def __str__(self):
        h = " ".join(["{}:{}".format(x,y) for (x,y) in zip(self._bins, self._counts)])
        return "[{}] {}".format(super().__str__(), h)

    @staticmethod
    def percentage(count, total):
        return count / total
```

Usage

```
>>> h = Histogram(["Intro", "Data Structures", "Algorithms", "Operating Systems"], "CS Course Enrollments")
>>> print(h)
[CS Course Enrollments] Intro:0 Data Structures:0 Algorithms:0 Operating Systems:0
>>> h.add_to_bin("Intro", 10)
>>> print(h)
[CS Course Enrollments] Intro:10 Data Structures:0 Algorithms:0 Operating Systems:0
>>> h.count("Intro")
10
>>> h.add_to_bin("Operating Systems", 30)
>>> print(h)
[CS Course Enrollments] Intro:10 Data Structures:0 Algorithms:0 Operating Systems:30
```