

CSC148 fall 2013

Introduction to computer science week 1

Danny Heap

heap@cs.toronto.edu

BA4270 (behind elevators)

<http://www.cdf.toronto.edu/~heap/148/F13/>

416-978-5899

September 11, 2013

python infested by objects

"word"

"word"



Here are some built-in objects to fool around with:

```
>>> w1 = "words"  
>>> w2 = "swords"[1:]  
>>> w1 is w2  
False
```

ids] don't match !

same values,
different
objects.

```
>>> import turtle  
>>> t = turtle.Turtle()  
>>> t.pos()  
(0.00,0.00)  
>>> t.forward(100)
```

turtles know
where they are

vandalizing existing classes

this is **deeply wrong**, except for teaching purposes...

```
>>> from turtle import Turtle
>>> t1 = Turtle()
>>> t1.pos()
(0.00,0.00)
>>> t1.forward(100)
>>> t1.pos()
(100.00,0.00)
>>> t1.neck
```

attributes of
class Turtle
are also attributes
of instances

```
Traceback (most recent call last):
```

```
File "<stdin>", line 1, in <module>
```

```
AttributeError: 'Turtle' object has no attribute 'neck'
```

```
>>> Turtle.neck = "very reptilian"
```

```
>>> t2 = Turtle()
```

```
>>> t2.neck
```

```
'very reptilian'
```

t2.neck
also works



Design a new class

Somewhere in the real world there is a description of points in two-dimensional space:

In two dimensions, a point is two numbers (coordinates) that are treated collectively as a single object. Points are often written in parentheses with a comma separating the coordinates. For example, (0, 0) represents the origin, and (x, y) represents the point x units to the right and y units up from the origin. Some of the typical operations that one associates with points might be calculating the distance of a point from the origin, or from another point, or finding a midpoint of two points, or asking if a point falls within a given rectangle or circle.

Find the most important noun (good candidate for a class...), its most important attributes, and operations that sort of noun should support.

build class Point...

in that **deeply wrong**, but **informative**, way

don't ever do this in assignment!

```
>>> from math import sqrt
>>> class Point(object):
...     pass
...
>>> def initialize(point, x, y):
...     point.x = x
...     point.y = y
...
>>> def distance(point):
...     return sqrt(point.x**2 + point.y**2)
...
>>> Point.__init__ = initialize
>>> Point.distance = distance
>>> p2 = Point(12, 5)
>>> p2.distance()
13.0
>>>
```

methods added to all turtles first parameter not used (implicit) self point



build class Point...

...properly!...

← do this

```
from math import sqrt
```

```
class Point(object):
```

```
    """Two dimensional point  
    """
```

```
    def __init__(self: "Point",  
                  x: "horizontal coordinate",  
                  y: "vertical coordinate") -> "None":
```

```
        """Initialize this point
```

```
        >>> p = Point(3, 4)
```

```
        """
```

```
        self.x = x
```

```
        self.y = y
```

```
# and so on
```

float
float

