

CSI31 Lecture 23

Topics: *Chapter 10. Defining Classes*

10.1 Quick overview of objects

10.3 Defining new classes

10.1 Quick overview of objects

recall lecture 13:

OO objects

contain data (know staff) (*instance variables*)

have operations (can do staff) (*methods*)

- in other words, it is a sort of *active* data type.

Objects *interact* by sending each other messages (requests for an object to perform one of its operations).

Every object is an *instance* of some class.

New objects are created by invoking a *constructor*.

Example: `circ=Circle(Point(100,100),20)`

The parameters to the constructor are used to initialize some of the instance variables (*center, radius* for Circle). Then we can manipulate the object (*draw, move, erase*).

10.3 Defining new classes

Class definition has the following form:

```
class <class-name>:  
    <method-definitions>
```

`__init__`, `roll`,
`getValue`, `setValue`
– are the methods of the
class `MSDie`

Let's define a class for a die:

- die has n sides (set it up during the *initialization*)
- when we roll a die it gets some value (method *roll*)

```
from random import randrange
```

```
class MSDie:
```

```
    def __init__(self, sides):  
        self.sides = sides  
        self.value = 1
```

```
    def roll(self):  
        self.value = randrange(1, self.sides+1)
```

```
    def getValue(self):  
        return self.value
```

```
    def setValue(self, value):  
        self.value = value
```

first parameter of each
method – a reference to
the object it works on

