

Objectives

- Wrap up defining classes
- `__lt__`, `__eq__` method
- Helper methods
- Creating classes - discussion
- Broader Issue: CS and Cancer

Mar 25, 2016

Sprengle - CSCI111

1

Review

- Where do we define the data that is needed to represent every object of a class?
 - How do we access that data?
- What method do you define for the object's constructor?
- How do we create a new method?
- What method do you define to be called by `print`?
 - What does this method look like and what does it do?

Mar 25, 2016

Sprengle - CSCI111

2

Implementing Deck Functionality

- Deck API, Implementation

Mar 25, 2016

Sprengle - CSCI111

3

`__LT__` and `__EQ__` METHODS

Mar 25, 2016

Sprengle - CSCI111

4

`__eq__`: Compare Objects of Same Type

- Header: `def __eq__(self, other)`
- Returns
 - True if `self` is equivalent to `other`
 - False otherwise
- Can now use objects in comparison expressions
 - `==`

How would you determine if two Card objects are equivalent?

Mar 25, 2016

Sprengle - CSCI111

5

`__lt__`: Compare Objects of Same Type

- Header: `def __lt__(self, other)`
- Returns
 - True if `self < other`
 - False otherwise
- Can now use objects in comparison expressions
 - `<`, `sort`

Rule of Thumb: define `__lt__` if there is a "natural" way to sort the objects

How do you compare two Card objects?

Mar 25, 2016

Sprengle - CSCI111

6

Comparing Objects of the Same Type

```
def __eq__(self, other):
    """ Compares Card objects by their ranks and suits """
    if type(self) != type(other):
        return False

    return self._rank == other._rank and \
           self._suit == other._suit

# Could compare by black jack or rummy value

def __lt__(self, other):
    """ Compares Card objects by their ranks """
    if type(self) != type(other):
        return False

    return self._rank < other._rank
```

Mar 25, 2016

Sprengle - CSCI111

card.py

7

Frequency Object

Alternative way to do sorting of FrequencyObjects

```
def __lt__(self, other):
    """Compares this object with other, which is
    also a FrequencyObject. Used when using the
    list's sort method."""
    return self._count < other._count
```

Question: is this the "natural" way to sort FrequencyObjects?

Mar 25, 2016

Sprengle - CSCI111

8

HELPER METHODS

Mar 25, 2016

Sprengle - CSCI111

9

Helper Methods

- Part of the class
- **Not** part of the API
- Make your code easier but others outside the class shouldn't use
- Convention: method name begins with "_"

Mar 25, 2016

Sprengle - CSCI111

10

Example Helper Methods

- Only *loosely* enforces that other can't use
 - Doesn't show up in `help`
 - Does show up in `dir`

Helper Method:

```
def _isFaceCard(self):
    if self._rank > 10 and self._rank < 14:
        return True
    return False

def rummyValue(self):
    if self._isFaceCard():
        return 10
    elif self._rank == 10:
        return 10
    elif self._rank == 14:
        return 15
    else:
        return 5
```

In use:

Mar 25, 2016

card2.py

Sprengle - CSC

Summary: Designing Classes

- What does the object/class represent?
- How to model/represent the class's *data*?
 - Instance variable
 - Data type
- What *functionality* should objects of the class have?
 - How will others want to use the class?
 - Put into methods for others to call (API)

Mar 25, 2016

Sprengle - CSCI111

12

Refactoring Bug Class

- What is a bug's data?
- What methods should a Bug object implement?

Mar 25, 2016

Sprengle - CSC111

bug.py

13

Benefits of Classes

- Package/group related data into one object
 - Deck can have list of Card objects rather than a list of ranks and a list of suits
- Reuse code
 - Card class used in war.py and deck.py
- Provide interface, can change underlying implementation without affecting calling code

Mar 25, 2016

Sprengle - CSC111

14

Changing Implementations

- Same API, different implementations

```
def __init__(self, rank, suit):
    self._rank = rank
    self._suit = suit

def getRank(self):
    return self._rank

def getSuit(self):
    return self._suit

def __init__(self, rank, suit):
    self._cardid = rank
    if suit == "clubs":
        self._cardid += 13
    elif suit == "hearts":
        self._cardid += 26
    elif suit == "diamonds":
        self._cardid += 39

def getRank(self):
    return (self._cardid - 2) % 13 + 2

def getSuit(self):
    suits = ["spades", "clubs", "hearts", "diamonds"]
    whichsuit = (self._cardid - 2) / 13
    return suits[whichsuit]
```

Tradeoff: Saving information (memory); Computing information

Mar 25, 2016

Sprengle - CSC111

card_byid.py

15

Considerations for Using Classes

Only use class if you're using most of its functionality/information

Since you don't know implementation, may inadvertently duplicate code

- Redo something done by class
- Could have efficiency penalties
- But time saved reusing code is usually worth it

Mar 25, 2016

Sprengle - CSC111

16

Broader Issue Discussion

- Relay Week!
- David Patterson
 - Former president of the ACM

"My introduction to the topic came when I was trapped in an elevator at the Grace Murray Hopper conference last October. Some conference attendees in the elevator recognized me as ACM's President, so I became an instant and captive recipient of feedback regarding our organization."

Mar 25, 2016

Sprengle - CSC111

17

Broader Issue Discussion

- Relay Week!
- David Patterson
 - Former president of the ACM

That's me!

"My introduction to the topic came when I was trapped in an elevator at the Grace Murray Hopper conference last October. Some conference attendees in the elevator recognized me as ACM's President, so I became an instant and captive recipient of feedback regarding our organization."

Mar 25, 2016

Sprengle - CSC111

18

Broader Issue Groups

Alice Ethiopia Holly Michael Ryan	Austin F Austin J Collin George Taylor	Chris Corson Lily Margaret Rachel R	Abdur Alicia Bennett Clark Josie	Allie Gunnar Rachel B Viktor
Andrew JT Max Stuart	Emily Eric Honor Madhav			

Mar 25, 2016

Sprengle - CSCI111

19

Broader Issue Discussion

- What is Big Data?
- How should we balance protecting patient privacy with sharing data?
- Any surprising finds in the W&L name data?

Mar 25, 2016

Sprengle - CSCI111

20

Looking Ahead

- Lab 10: Facebook

Mar 25, 2016

Sprengle - CSCI111

21