Week six

Help

Objects, classes and their terminology — Object-oriented programming

- All data entities in Python are *objects*. Compound objects consist of multiple pieces of data.
- New types of compound objects can be defined using the keyword `class`. For example, a class for card decks could be created using the definition statement `class Deck:`
- The data for objects in the class are contained in user-defined *fields*. For example, the Deck class could include a `cards` field that corresponds to a list of cards.
- Functions defined inside the class definition are *methods*. These method create and manipulate the object and the data in corresponding fields. For example, the Deck class might include a `shuffle` method that shuffles a Deck object.
- Lecture examples - Character, Ball
- More examples - Class Structure, Class Errors

Create and working with objects — Object-oriented programming

- The class initializer `__init__` generates instances of class objects. In Python, this initializer can be called via the expression `class_name(...)`. For example, an instance of the `Deck` class can be created via the statement `my_deck = Deck(...)`.
- The first parameter to class methods is, by convention, always named `self`. This name refers to the object being acted on by the method.
- Class fields for an object are defined/modified in class methods via `self.class_field = ...` For example, `self.cards = []` would assign an empty list to the `cards` field in a class method for a Deck object.
- Class methods can be applied to a class object via `class_object.class_method(...)`. For example, a `shuffle` method could be applied to a deck via the statement `my_deck.shuffle()`.
- Objects for user-defined classes are mutable. In particular, modification of a field in an object via a class method mutates the object.
- Lecture examples - Character, Diffusing Particles Blackjack Template
- More examples - Class Structure, Class Errors, Bubbles, Flowers

Tiled images — Tiled images

- A tiled image is a single image that consists of a set of smaller images laid out in a regular grid.
- Tield images are useful since loading a single large image is faster than loading many small images.
- Small images in the tiled image can be drawn by specifying the appropriate source rectangle for `draw_image` using the size of the small image and its position in the grid.
- Lecture examples - Tiled Images
- More examples - Bunny Emotions

Programming Tips — Week 6, Prime Numbers (while loops)

Created Wed 5 Sep 2012 10:49 PM CEST

Last Modified Mon 3 Jun 2013 4:17 AM CEST