OOP via Python: Session 05

Stephen Leach, Nov 2021

X-Ray Vision

- The art of identifying latent objects and their methods in procedural code
- In procedural code, data is passive
- Functions create, interrogate and update data
- OOP reorganises code so that data and the code that operates on it are grouped into classes

Demo: range_extract.py

Convert Generic Structures into Custom Classes

- In Python we make heavy use of tuples, lists and dicts
- These generic classes are very richly supported by a variety of functions & methods
- And they require no additional coding
- But they can be overused - and they are a hint that a useful class is concealed

Demo: Convert Tuples into a Custom Class

- Replace generic tuples in range_extract.py with RangeOfPages

Extract-and-shift

- Find a chunk of code that spends all its time fussing with objects of class X
- Extract the chunk of code into its own function/method
- Shift that method into class X

Demo: Shift Constructors

- Extract the calls to the new constructor and shift into the class as a @staticmethod

Gather buddy variables

- A group of variables that only occur together are 'buddies'
- They are a "smell" that indicates that they represent a meaningful object
- Create a 'naked' class that captures those variables
- Then extract-and-shift

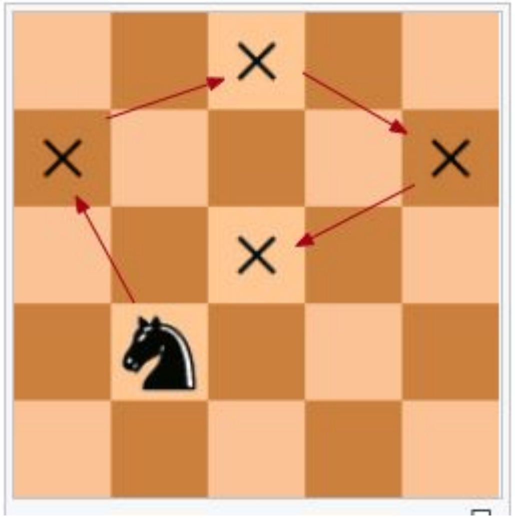
Demo: Gathering Buddy Variables & Start Shifting

```
class Cursor:
   def __init__(self, lst):
      self._i = 0
      self._lenlst = len(lst)
      self._lst = lst
```

Demo: Knight's Tour

The Knight's Tour Problem

- An oldie but a goodie
- https://en.wikipedia.org/wiki/Knight%27s_tour
- Visit every square on the board without landing on the same square twice?



Demo: Gather Buddy Variables

```
class Board:
   def __init__(self, board, boardsize):
      self._board = board
      self._boardsize = boardsize
```

Demo: Extract-and-Shifting

- Set about hiding the implementation of Board

Demo: Refactoring

- Merge min and accessibility
- Turn knightsmoves into an iterator
- Eliminate the deepcopy

Demo: Organise

- Core vs Non-Core
- Public vs Non-Public

Also

- Collate function parameters into parameter-objects
- Converting switches into method dispatch
- Convert entire functions into objects + series of calls