Objects

Todd W. Neller

Motivation for Object

- Organizing strategy: Store related things together.
  - Kitchen: store dishes together
  - Office: store letter-writing items (e.g. envelopes, stamps, return address labels) together
- **Objects** are bundles of related code and data.
- Objects are one way programmers “keep their rooms clean”.

Parallel Arrays

- Imagine non-object oriented code modeling a partly-empty deck of cards:
  - Partially filled array of card ranks:

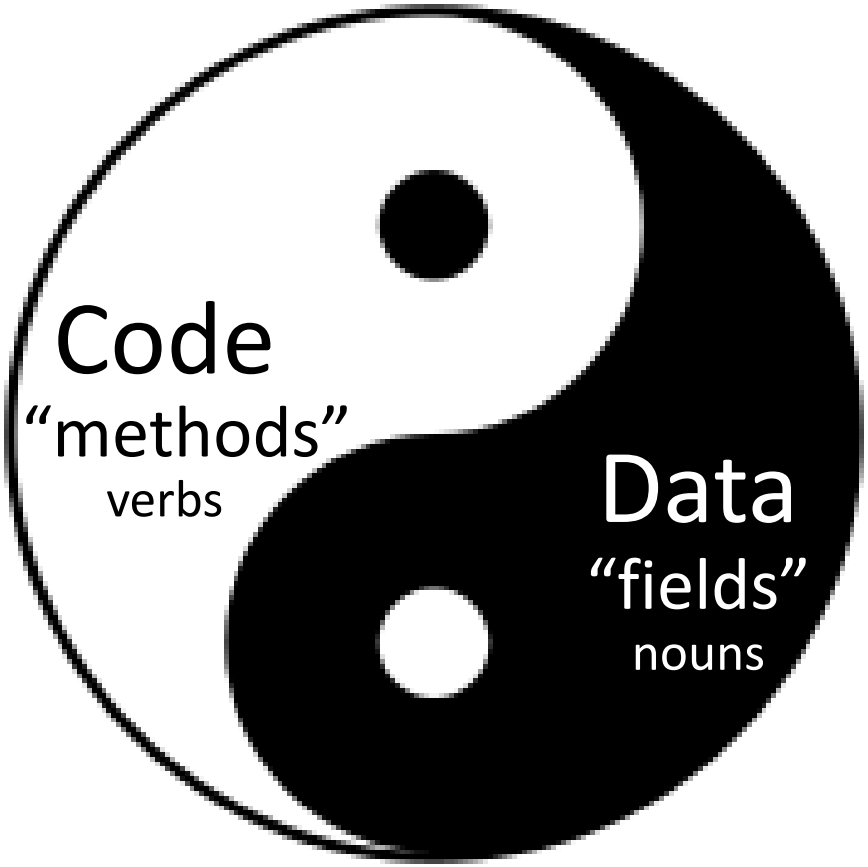
Q	7	…	4	null

  - Partially filled array of card suits:

C	D	…	S	null

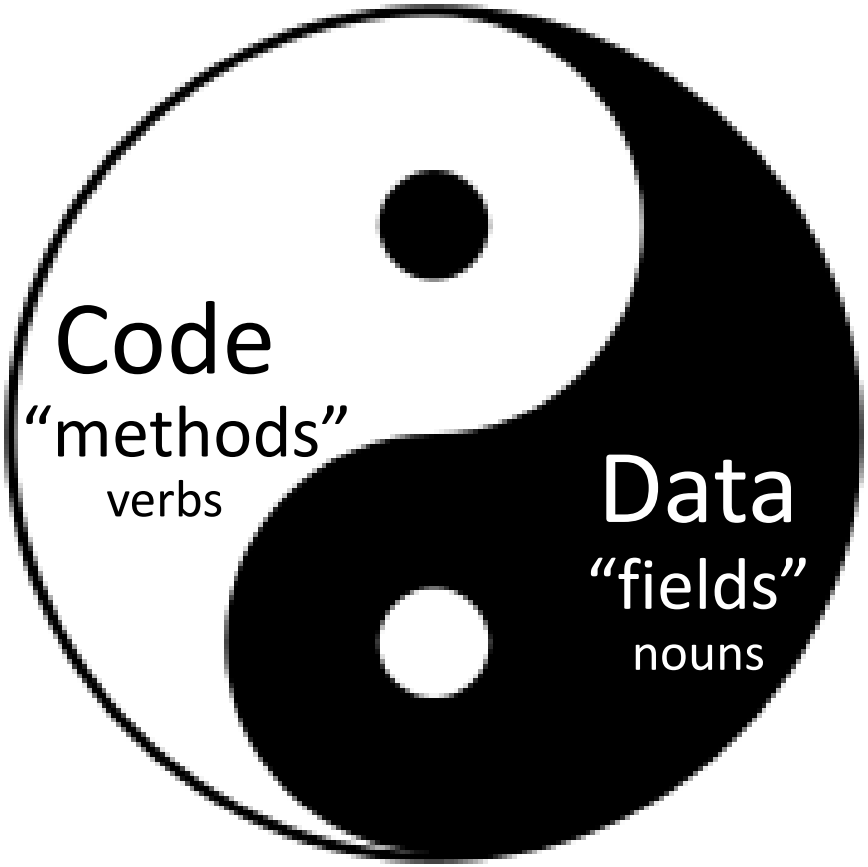
  - Parallel arrays refer to different attributes of a single item at the same index.
- Address parallel arrays for first name, middle name, last name, street address, city, state, zip code, etc.

Object



Bank Account Object

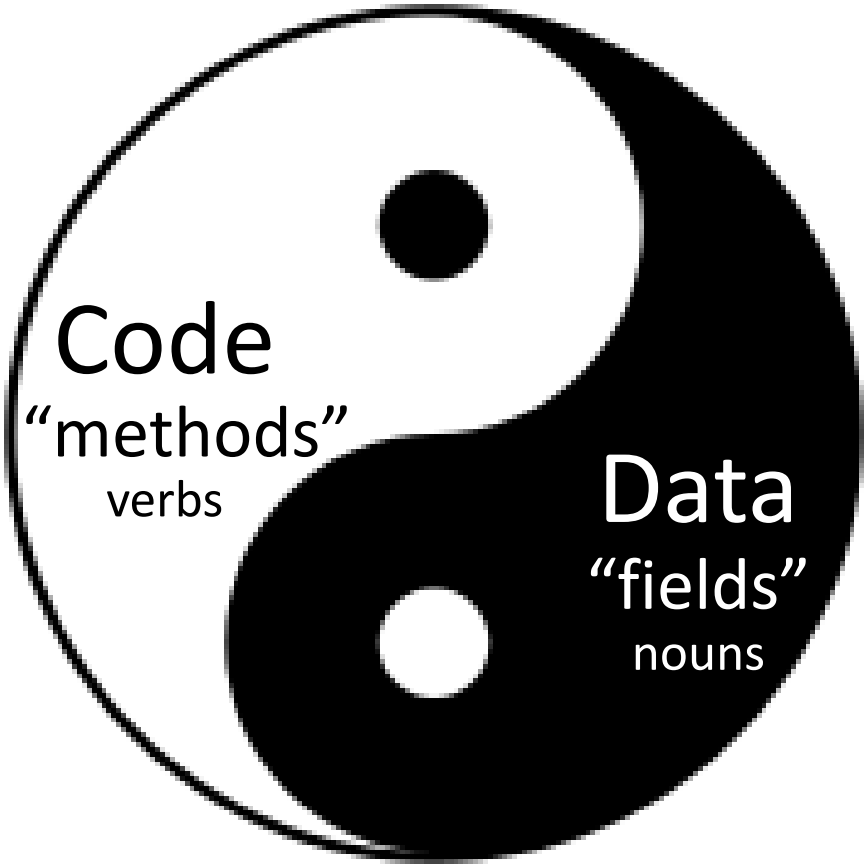
Get balance
Deposit
Withdraw
Transfer
…



Balance
Account #
Name
SSN
Interest rate
…

Card Object

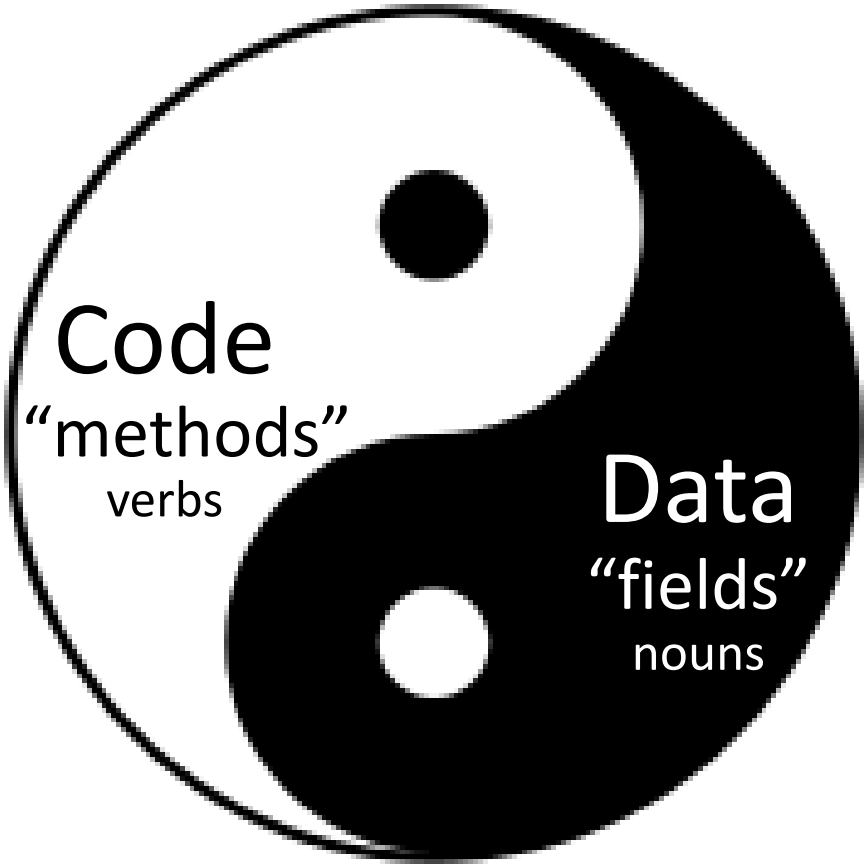
Get rank
Get suit
Get color
Get card name
Compare to
…



Rank
Suit
Color

Card Deck Object

Shuffle
Add card
Draw card
…



Card sequence

Object-Oriented Design Advice

One simple, general approach to object-oriented (O-O) design is to:

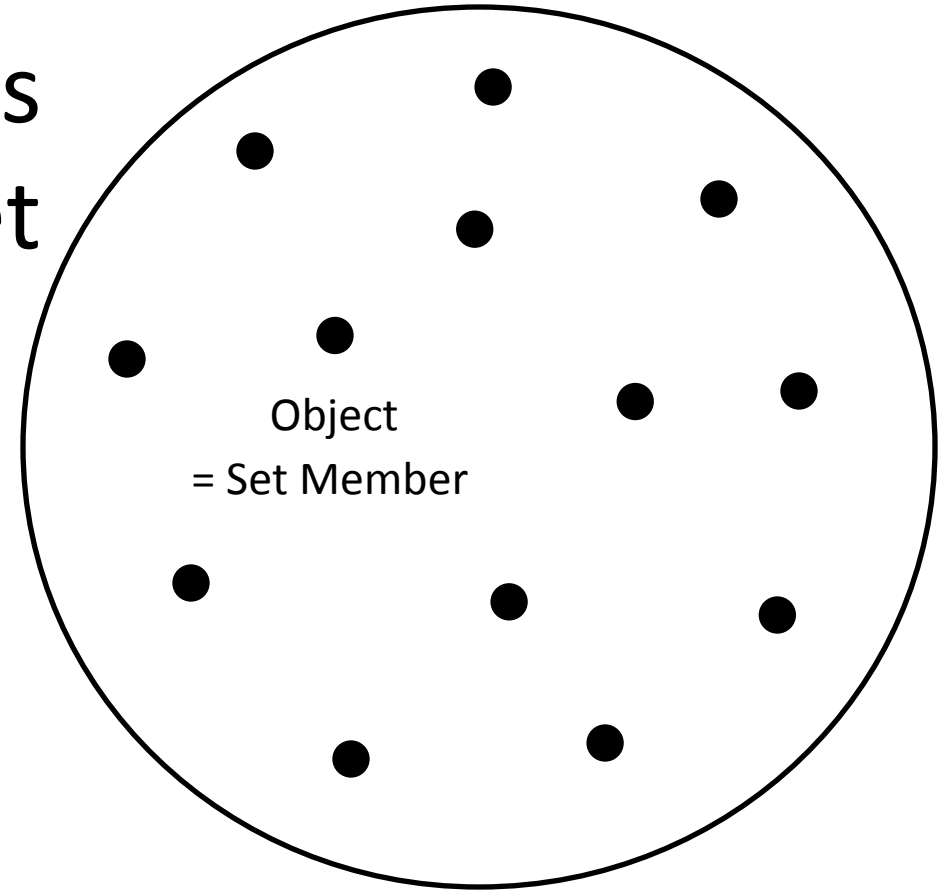
1. Get a text description of the problem.
2. Copy the nouns out and group them by relevance.
3. Draw lines between directly related nouns (e.g. "__ has a __", "__ is a __").
4. Thinking of this noun graph as a flight map, "hubs" (nouns that relate to many other nouns) are central organizing concepts that should likely be objects. Other nouns are likely fields.
5. Verbs in the description are likely methods to be defined as part of the most relevant object and taking other relevant objects as parameters.

Test-Driven Development (TDD)

- Why is test code beneficial?
  - Quality assurance
  - Focused guidance of development
  - Reusable after code revisions, additions, etc., to ensure backwards compatibility
- Test-Driven Development (TDD) is the practice of writing test code first and then implementing so as to pass the tests.
- Common problem: Test code that is not written first is sometimes never written!

Classes

Class
= Set



Remember: **Class** is to **Object** as **Set** is to **Member**.

Static versus Non-static

- Static – “associated with the **class**”
- Non-static – “associated with the **object**”
- Examples of static fields and methods:
  - Math.PI, Math.E, Integer.MAX_VALUE, System.out
  - Math.sin, Math.max, Integer.parseInt
- Examples of non-static methods:
  - (new Scanner(System.in)).readLine();
  - (new StringBuilder(“test”)).toString();