Accessor methods

Why bad?
- Provide external access to implementation detail
  - Abusing can violate encapsulation principle
- Return value type must match implementation
  - More points of failure if implementation is changed

When to use?
- When you absolutely have to
- When the system is small, very few set/get, little data flow

How to use?
- Have to use accessor
  - Return objects that implement interfaces
    - Return type is consistent, with reasonable behavior
    - Implementation is hidden
- Don't have to use accessor
  - Tell, Don't Ask
    - Tell other objects to do the task themselves
    - Don't ask about their states and do the task youself
    - Build static model, where relationship ~ data flow