# Encapsulation: the most important design pattern of all…

* Hide all but essential properties and methods by making them private (best) or protected (partially hidden, but visible within same package)
* Combine a cohesive set of behaviors in one unit (a class, e.g.)
* Encapsulate the things that vary into separate components  
    
    
  Design Principle:  
    
  Identify the aspects of your application that vary and separate them from what stays the same.

# Benefits…

* Improves ease of use
* Improves flexibility and reuse potential.
* Can you think of others???

# We can Encapsulate Many of the Duck Behaviors…

* Because they vary from duck to duck
* Because a component will be easy to change vs. changing many methods inherited from Interfaces
* Using encapsulated components enables us to change the behavior of a duck dynamically at runtime
* Can you think of others????