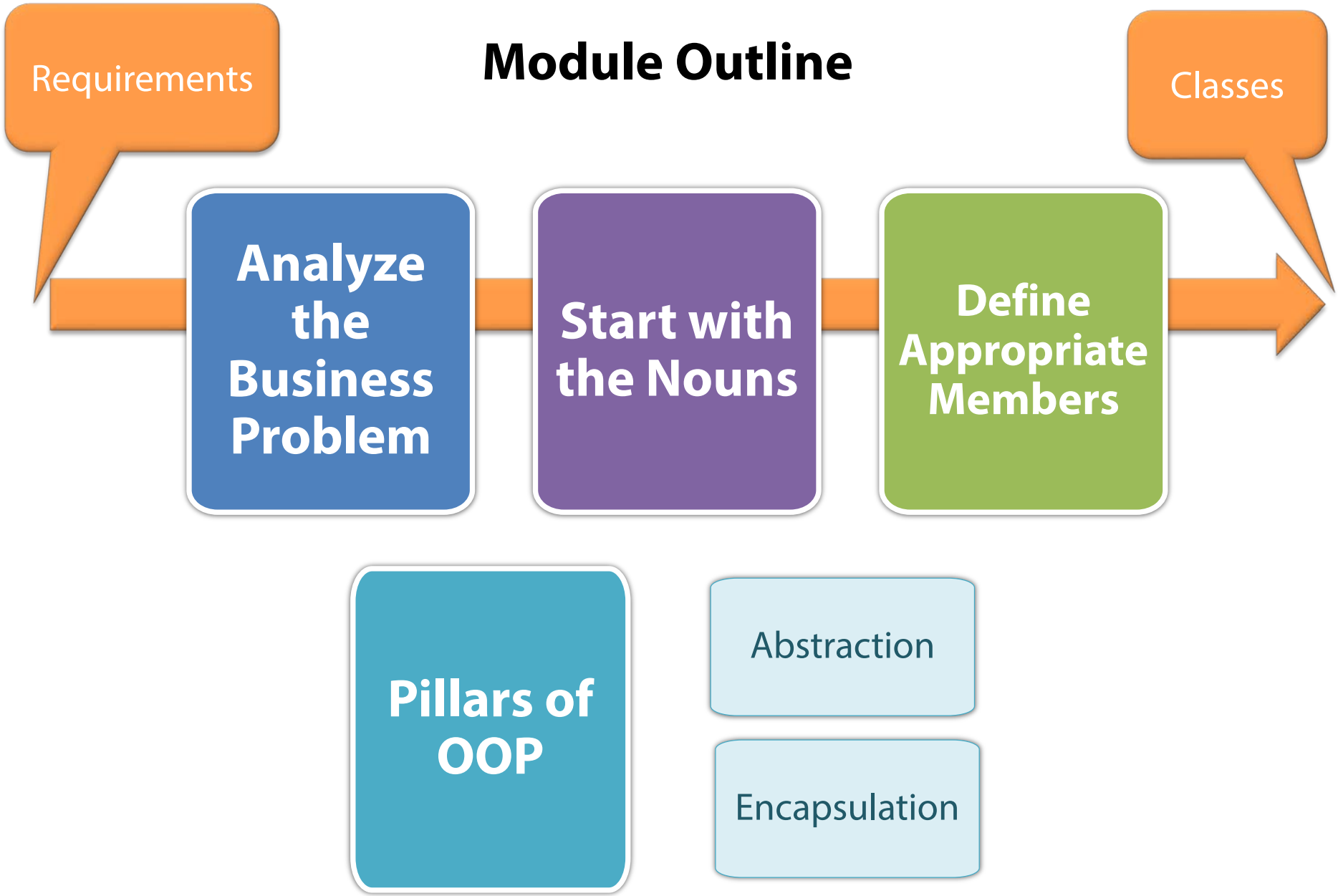
Identifying Classes from Requirements

Object-Oriented Programming Fundamentals in C#

Deborah Kurata
http://msmvps.com/blogs/deborahk/
@DeborahKurata
deborahk@insteptech.com

pluralsight
hardcore dev and IT training

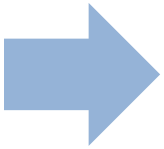
Module Outline



Business Requirements

Object-Oriented Programming (OOP)

Identifying Classes



- Represents business entities
- Defines properties (data)
- Defines methods (actions/behavior)

Separating Responsibilities

Establishing Relationships

Extracting Commonality

Acme Customer Management System

Manage business, residential, government, and educator types of customers

- Customer's name (Last name, first name)
- Email address
- Home and work addresses

Manage both our current products and Consolidated Systems' products

- Product name
- Description
- Current price

Accept orders from customers either online or through our call center

- Customer
- Order date
- Shipping address
- Products and quantities ordered

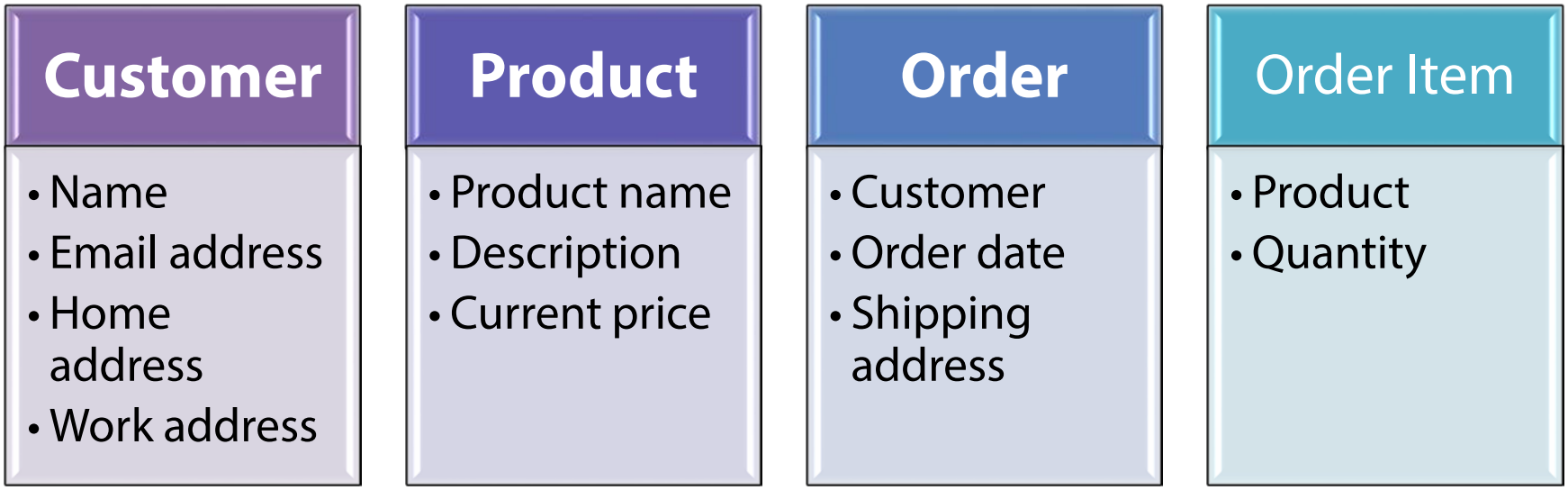
Start with the Nouns

Manage business, residential, government, and educator types of customers	Customer
Manage both our current products and Consolidated Systems' products	Product
Accept orders from customers either online or through our call center	Order

Define Appropriate Members

Customer	Product	Order
• Name • Email address • Home address • Work address	• Product name • Description • Current price	• Customer • Order date • Shipping address • Product • Quantity

Define Appropriate Members



Define Appropriate Members

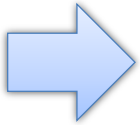
Customer	Product	Order	Order Item
• Name • Email address • Home address • Work address • Validate() • Retrieve() • Save()	• Product name • Description • Current price • Validate() • Retrieve() • Save()	• Customer • Order date • Shipping address • Validate() • Retrieve() • Save()	• Product • Quantity • Validate() • Retrieve() • Save()

Define Appropriate Members

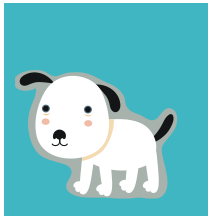
Customer	Product	Order	Order Item
• Name • Email address • Home address • Work address • Validate() • Retrieve() • Save()	• Product name • Description • Current price • Validate() • Retrieve() • Save()	• Customer • Order date • Shipping address • Validate() • Retrieve() • Save()	• Product • Quantity • Purchase price • Validate() • Retrieve() • Save()

Abstraction

Manage business, residential, government, and educator types of customers



Customer



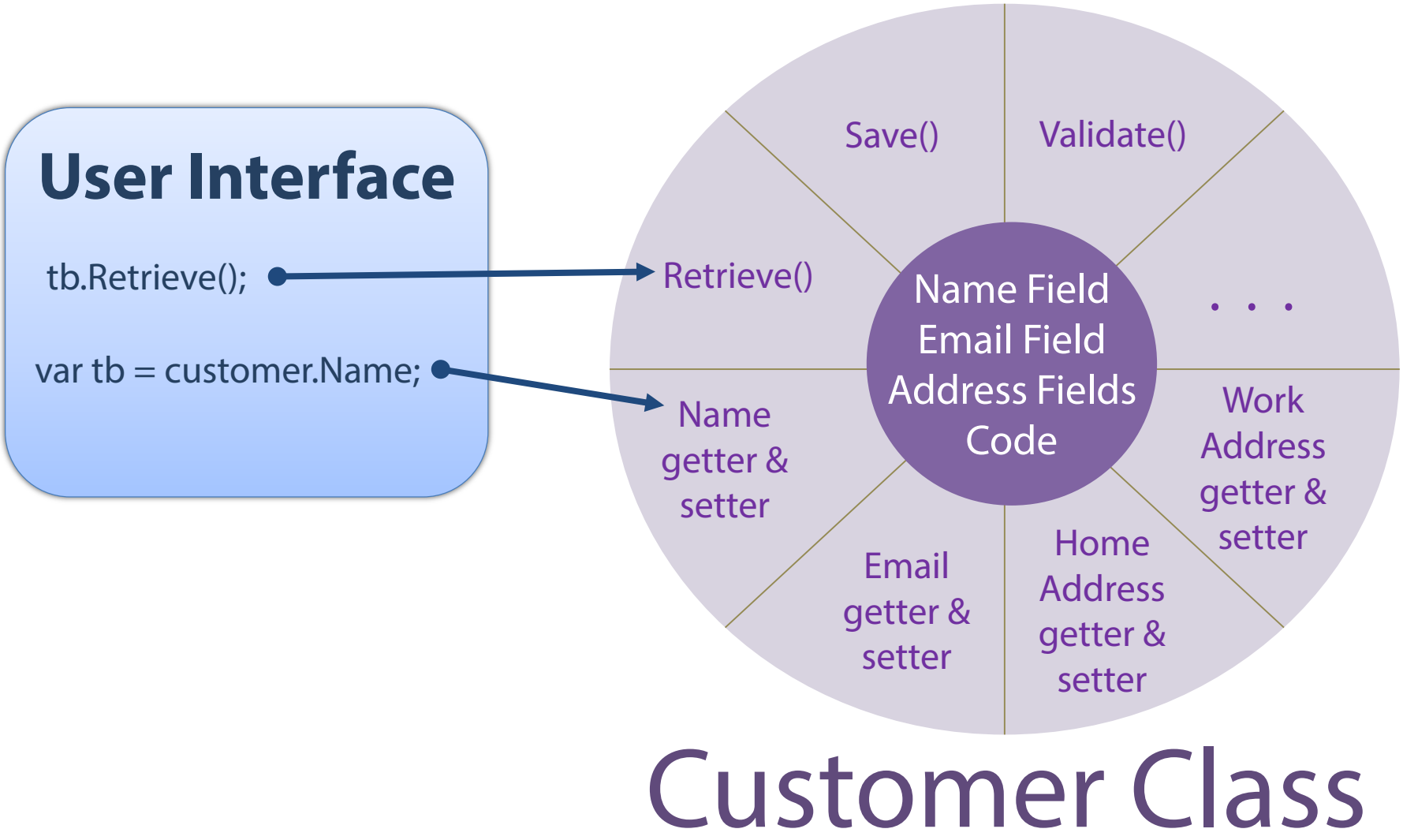
Joe Smith
Joe@aol.com
123 Main St.

Abstraction

Abstraction

- Simplifying reality
- Ignoring extraneous details
- Focusing on what is important for a purpose

Encapsulation



Encapsulation

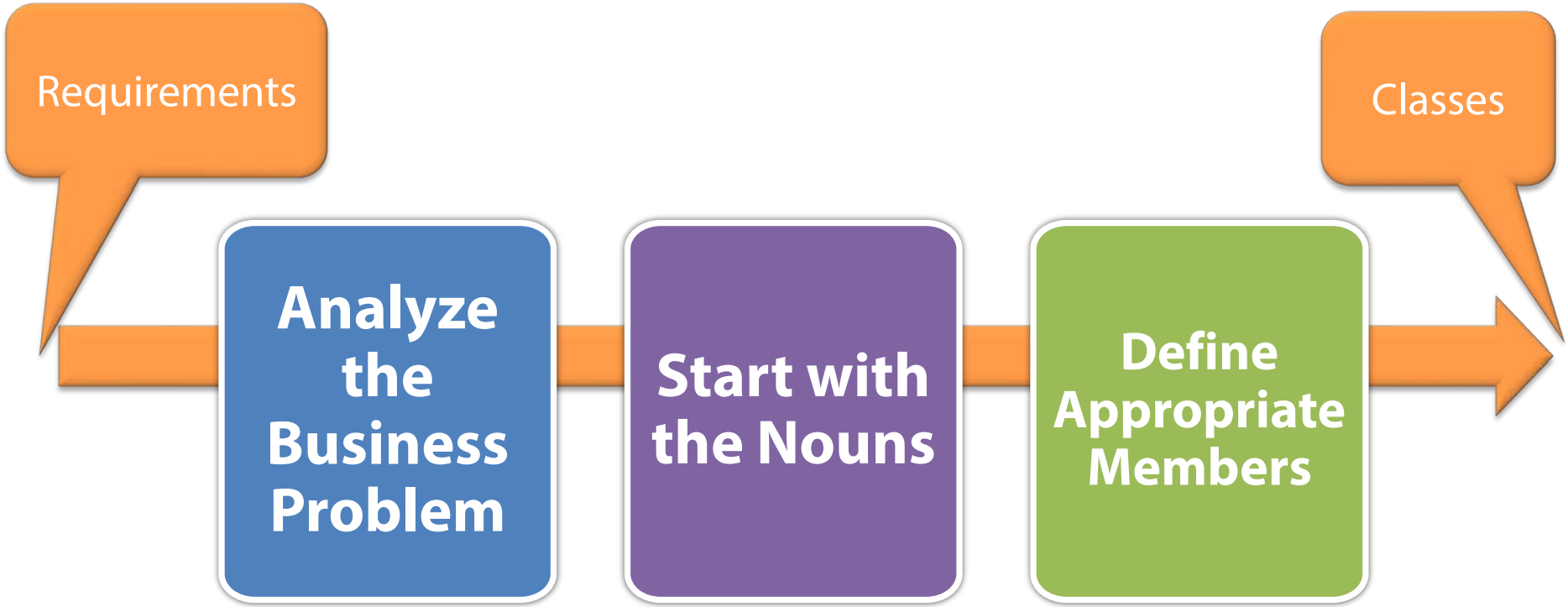
Data hiding

- Protects the data
- Allows for authorization before getting the data
- Allows for validation before setting the data

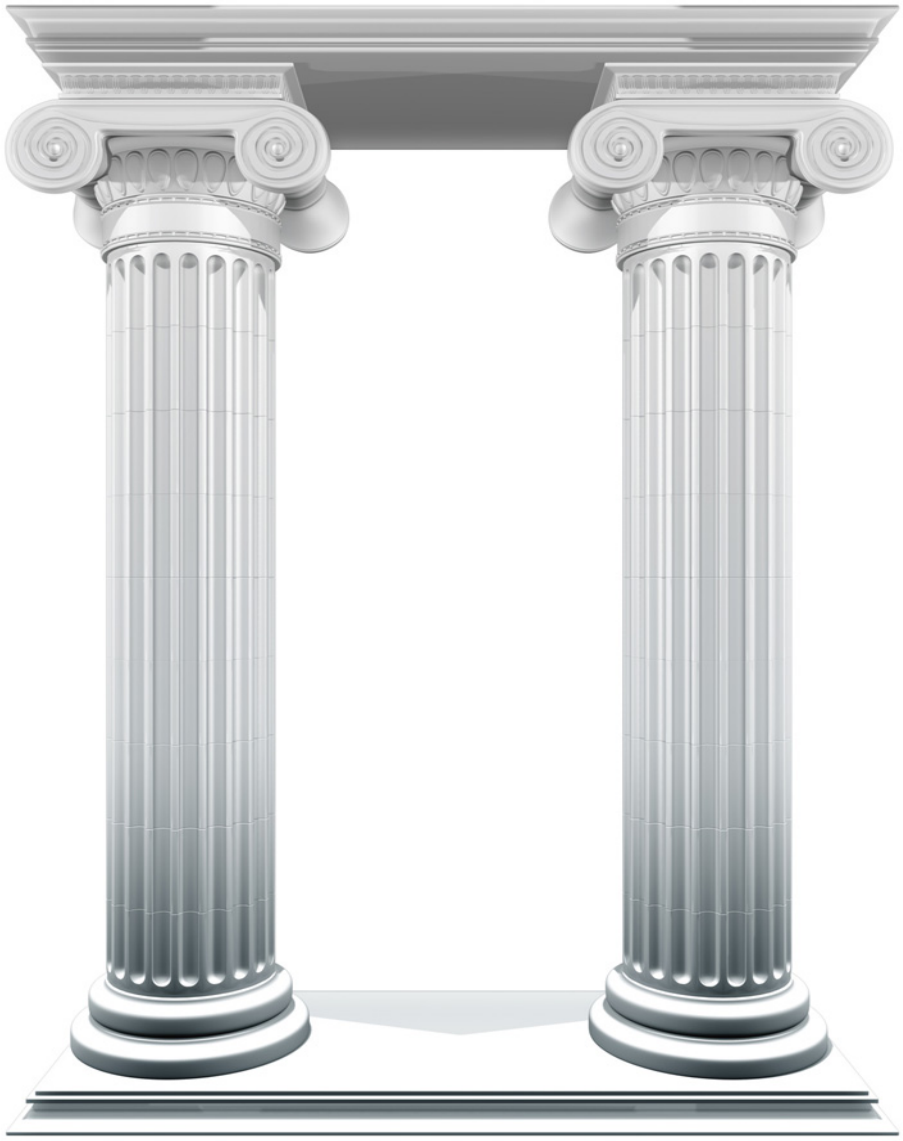
Implementation hiding

- Helps manage complexity
- Only the class needs to understand the implementation
- Implementation can be changed without impacting the application

Summary

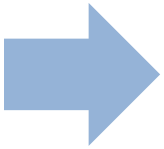

Requirements
Analyze the Business Problem
Start with the Nouns
Define Appropriate Members
Classes

Four Pillars of OOP



Object-Oriented Programming (OOP)

Identifying Classes



- Represents business entities
- Defines properties (data)
- Defines methods (actions/behavior)

Separating Responsibilities

Establishing Relationships

Extracting Commonality