# Mapping Designs to Code

Larman, Chapter 20
CSE432
Object Oriented Software Engineering

## 00 development is iterative

- OOA/D artifacts feed into implementation model in a traceable manner
- Some tools generate partial code from UML
- But programming not trivial generation!
- Programmers make changes as the work out the details
- Therefore, Expect and plan for change and deviation from design during programming

# Mapping Designs to Code

- Write source code for:
  - Class and interface definitions
  - Method definitions
- Work from OOA/D artifacts
  - Create class definitions for Domain Class Diagrams (DCDs)
  - Create methods from Interaction diagrams

## From DCD to Java class

```
public class SalesLineItem
{
  private int quantity;

  private ProductDescription description;

  public SalesLineItem(ProductDescription desc, int qty) { ... }

  public Money getSubtotal() { ... }
}
```

SalesLineItem quantity: Integero

getSubtotal(): Money

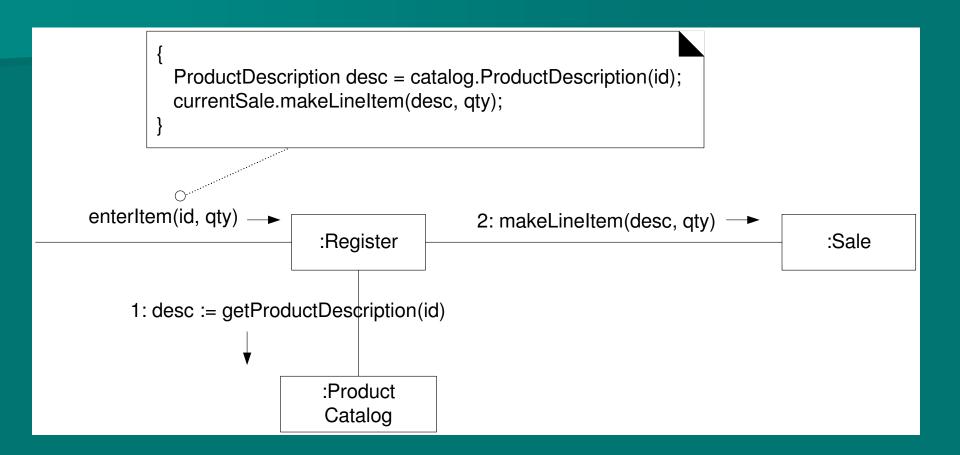
description

Product Description

description : Text price : Money itemID : ItemID

...

#### From Interaction diagram to method



### Collection classes

What collection class has been added to the design and why?

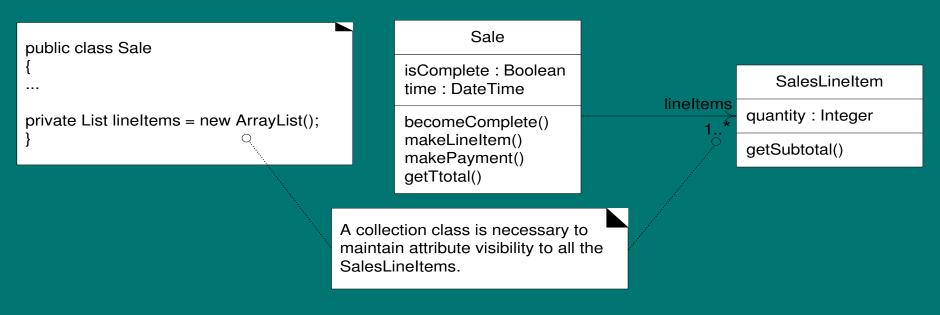


Fig. 20.5

# **Exception handling**

- Why is it wise to consider large-scale exception handling strategies during design modeling?
- In UML, exceptions can be inserted as property strings of messages

Why implement from least-coupled to most-coupled?

