

# **Operation Contracts**

CSE 3311 & 5324

Christoph Csallner

University of Texas at Arlington (UTA)

# Motivation

- Operation contract = (pre- and) **postcondition of a step in a use-case scenario**
- CL, Chapter 11
- Operation contracts
  - **Part of use-case modeling**
- An operation contract
  - Details one step of a use-case scenario
  - **Describes system-internal state change**

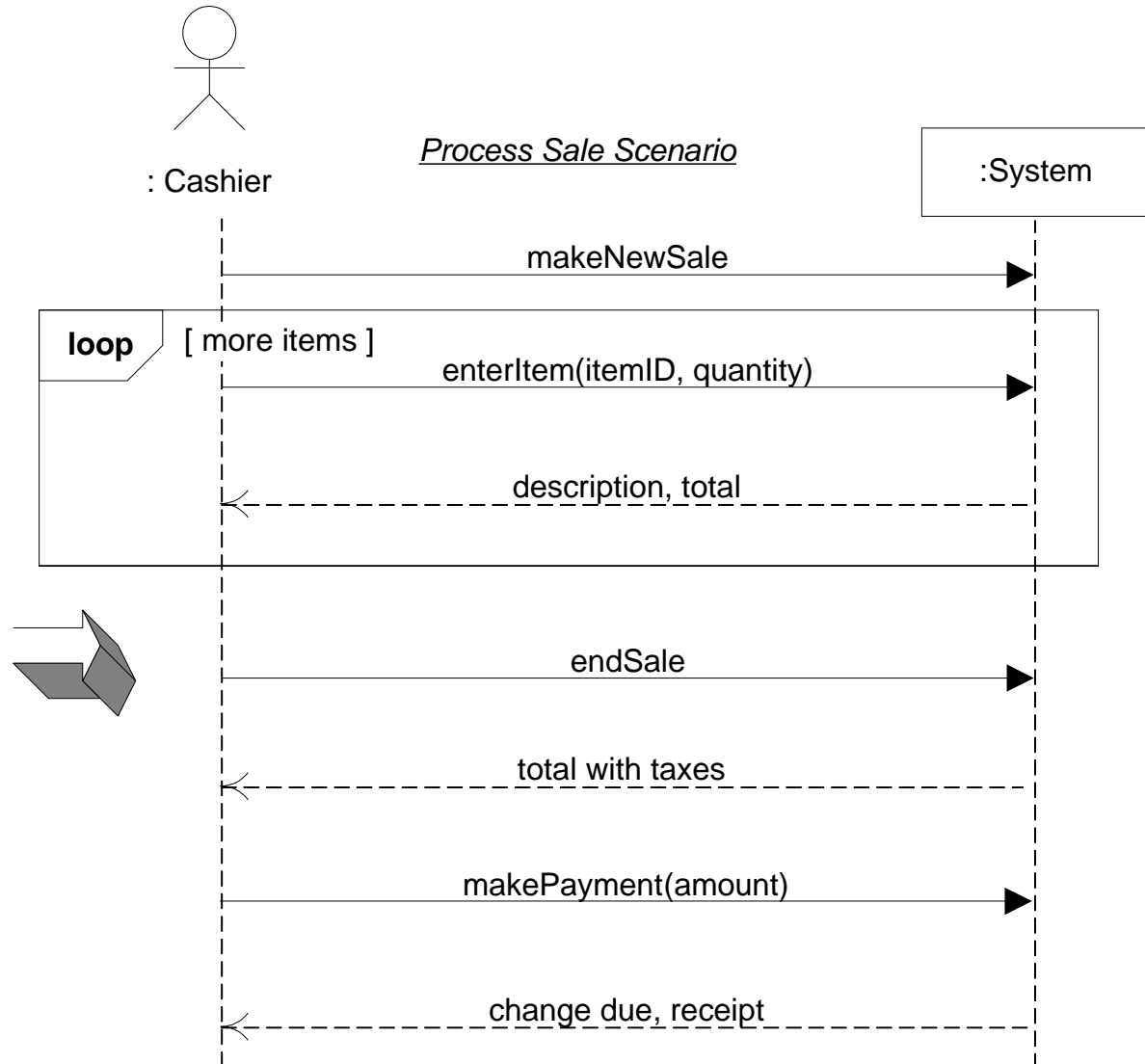
# Review

- Use-case = Text story
  - Main success scenario
  - Error scenarios
- Scenario = Sequence of steps or actions
  - Ping-pong between primary actor and system
  - (Action by primary actor, [system response])\*
- Treat system as a single black box
- System Sequence Diagram
  - Rephrases each action by the primary actor as a method signature
  - **Method signature = system operation**

# Review: System Sequence Diagram

## Simple cash-only *Process Sale* scenario:

1. Customer arrives at a POS checkout with goods and/or services to purchase.
2. Cashier starts a new sale.
3. Cashier enters item identifier.
4. System records sale line item and presents item description, price, and running total.  
Cashier repeats steps 3-4 until indicates done.
5. System presents total with taxes calculated.
6. Cashier tells Customer the total, and asks for payment.
7. Customer pays and System handles payment.
- ...



# Terminology

□ <b>UML</b>	<b>vs.</b>	<b>Object-oriented languages</b>
– Operation	--	
– Method	--	
– Event	--	

# Terminology

## □ UML

- Operation
- Method
- Event

## vs. Object-oriented languages

- Method signature: Name, param types
- Method body: Code / implementation
- Method invocation: Method call

# Operation Contract Template

- Name of operation + parameters
- Use-cases this operation occurs in
- Precondition
  - **State of objects in domain model before operation**
- Postcondition
  - **State of objects in domain model after operation**
- Pre- and post-conditions may be expressed in
  - Natural language: Quick and informal
  - Object Constraint Language (OCL) or other formal lang.
  - Hybrid

# Possible State Changes

- ▣ Contents of sets (recall: Set shown as box)
  - Who exists?
  - Objects may be added to / deleted from a set
- ▣ Contents of relations (Relation = set of tuples)
  - Who is associated with whom?
  - Tuples may be added to / deleted from a relation
- ▣ State of object attributes
  - Value of attributes may change



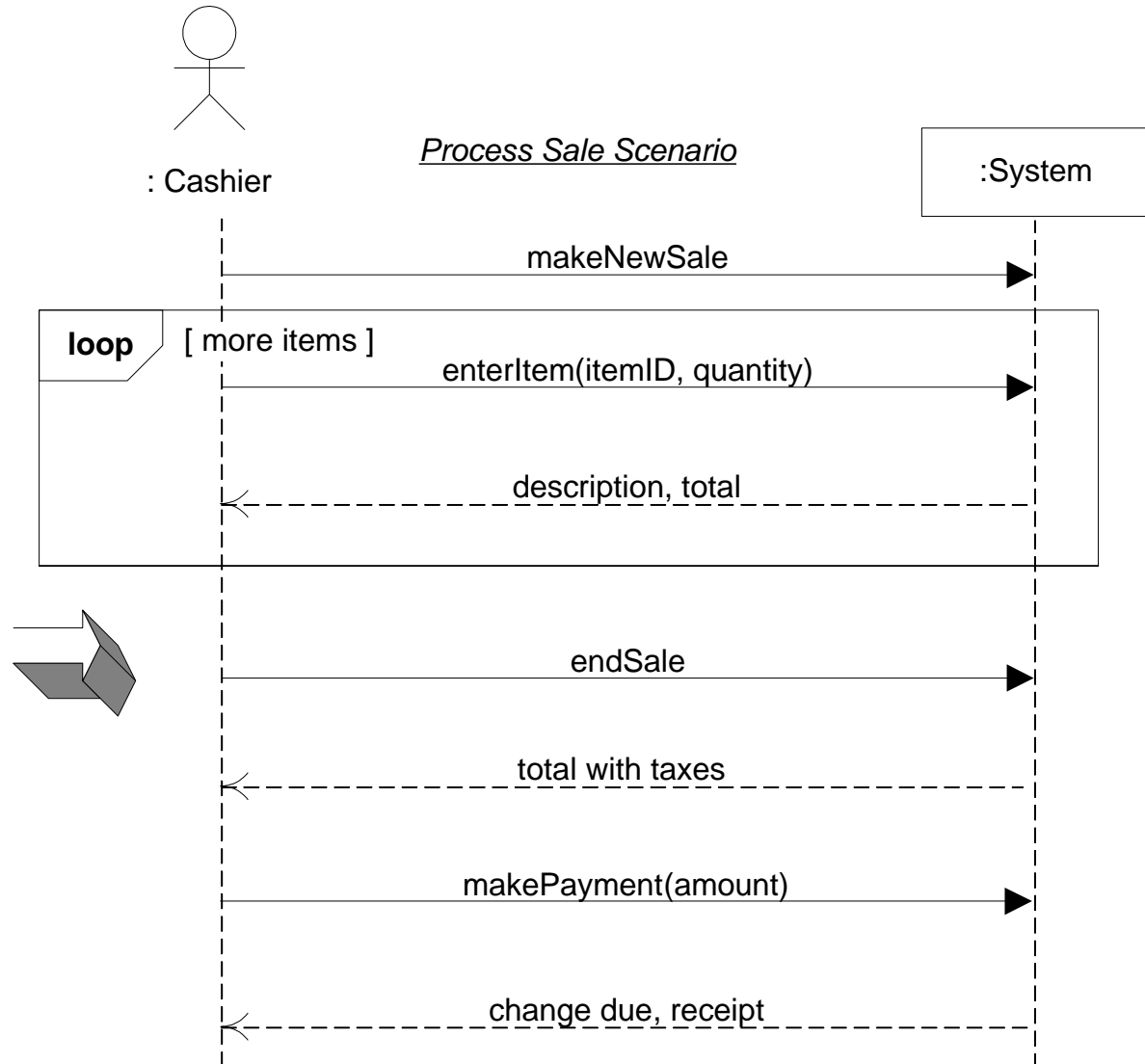
# Guideline

- Write post-condition in past tense
  - Observation about state change that arose from an operation

# Recall the System Sequence Diagram

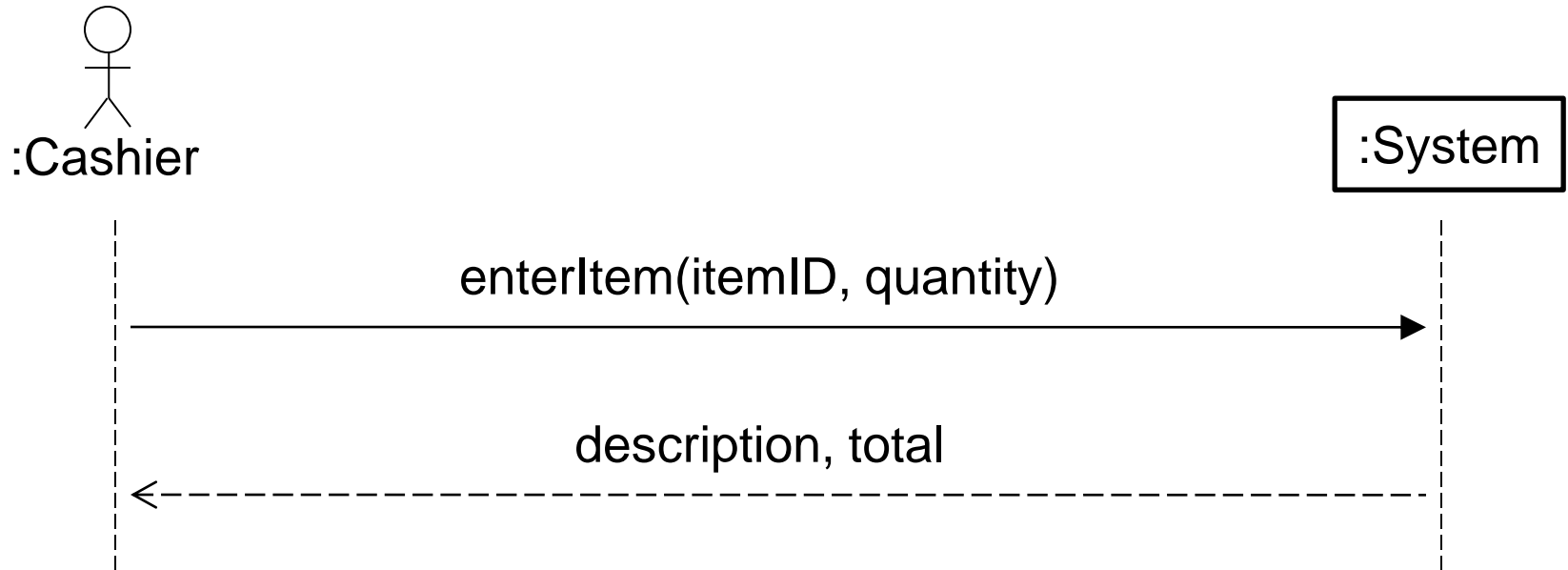
## Simple cash-only *Process Sale* scenario:

1. Customer arrives at a POS checkout with goods and/or services to purchase.
  2. Cashier starts a new sale.
  3. Cashier enters item identifier.
  4. System records sale line item and presents item description, price, and running total.
- Cashier repeats steps 3-4 until indicates done.
5. System presents total with taxes calculated.
  6. Cashier tells Customer the total, and asks for payment.
  7. Customer pays and System handles payment.
- ...

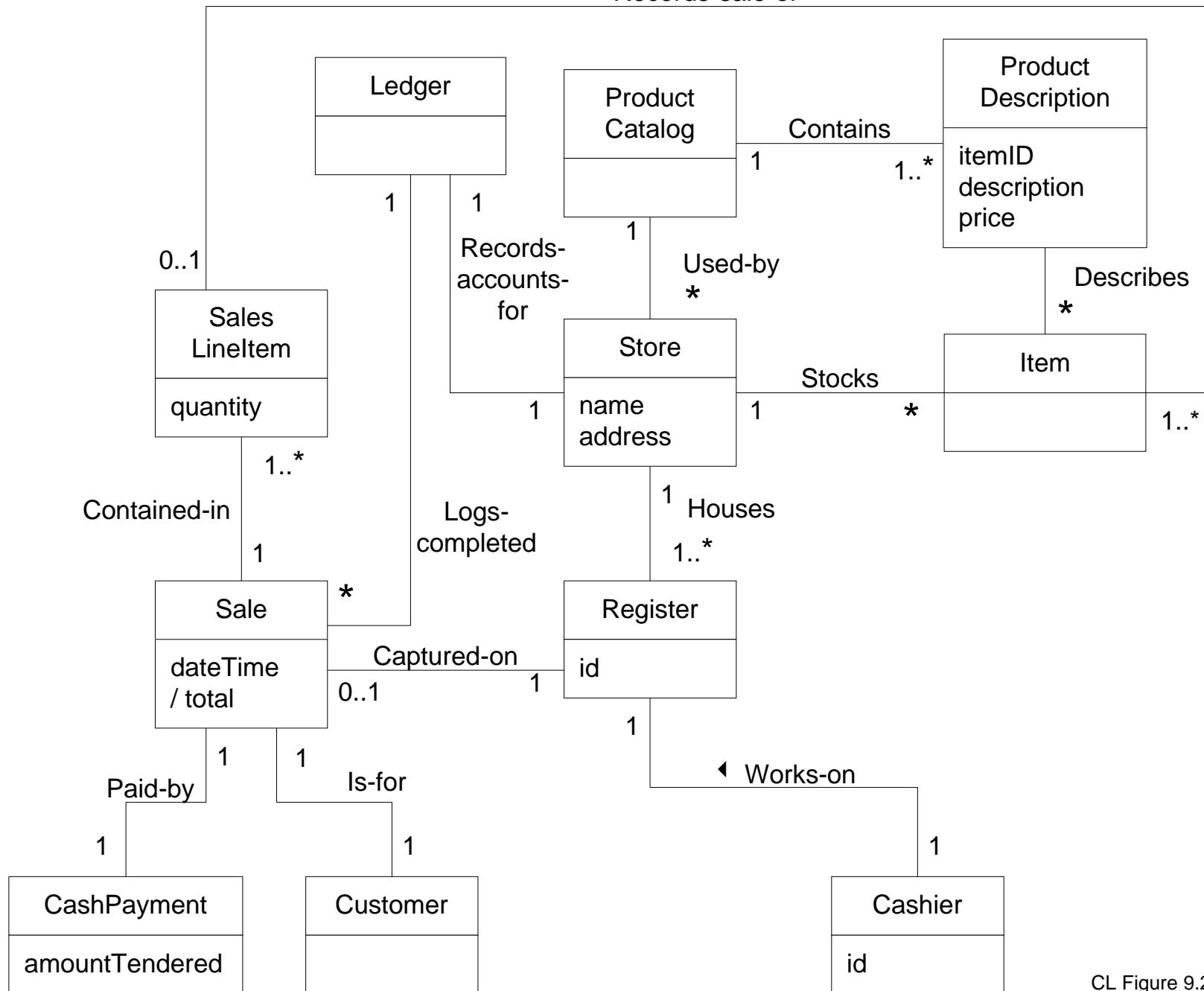


# POS Example

- Excerpt of system sequence diagram:



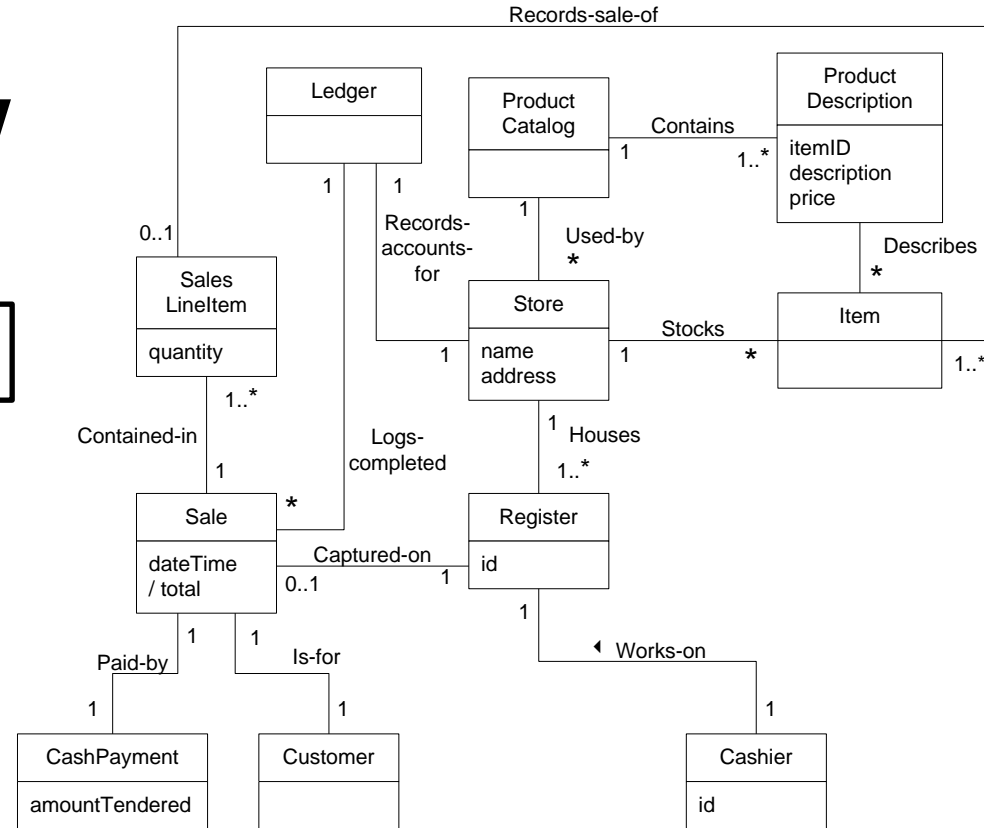
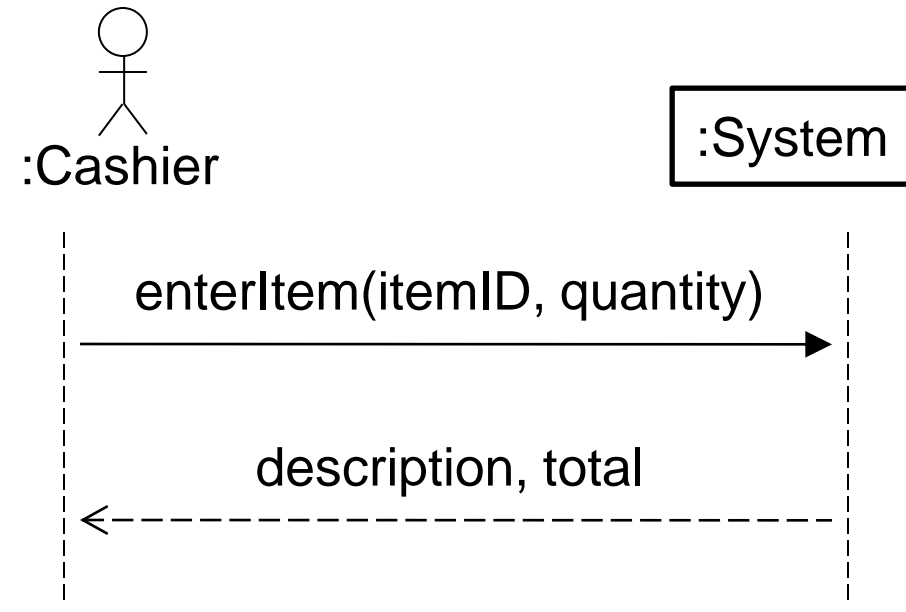
# Records-sale-of



# Example Operation Contract

- enterItem(itemId, quantity)
- In use-cases: Process Sale
- Precondition
  - There is a sale underway
- Postcondition
  - A **SalesLineItem** instance **sli** was created
  - **sli** was associated with the current **Sale**
  - **sli.quantity** became quantity
  - **sli** was associated with a **ProductDescription**, based on **itemId** match
- (Bold = part of the domain model)

# Example Overview



- Postcondition of **enterItem(itemID, quantity)**:
  - A **SalesLineltem** instance **sli** was created
  - **sli** was associated with the current **Sale**
  - **sli.quantity** became quantity
  - **sli** was associated with a **ProductDescription**, based on **itemID** match



# **IN-CLASS EXERCISE: OPERATION CONTRACTS**

# Create Operation Contracts

- Get together with your team
- Add operation contracts to a main success scenario
  1. Pick the main success scenario of a use-case you defined for your project inception
  2. Add post-conditions to each step of the main success scenario
- Be prepared to present your results