UML (CONT.)

RECALL

- Diagrams to depict project structure
  - Show classes, interfaces, etc.
  - Show relationships between them
- Classes: rectangle with 3 sections
  1. Class name
  2. Instance variables
  3. Methods

RECALL

- Instance variables

```
vis name : type [= default_value]
```

- Methods

```
vis name(param_name1 : type,
param_name2 : type, ...) : return_type
```

DEPICTING A CLASS (CONT.)

- Abstract classes: italicize class name
- Interface: `<<interface>>` placed above name
- Static methods/variables: underline

DEPICTING RELATIONSHIPS: GENERALIZATION

- Relationship between general thing and more specific kind of it
- "is-a" relationship indicated through inheritance
- Use solid line with open arrowhead pointing from child to parent

DEPICTING RELATIONSHIPS: REALIZATION

- When one thing specifies contract another must carry out
- aka, interface implemented by a class
- Use dashed line with open arrowhead pointing from class to interface

DEPICTING RELATIONSHIPS: ASSOCIATION

- When one object "has-a" different object
- `A` has-a `B` if `B` is type of field(s) in `A`
- Example: Book class has instance variable that is Publisher
- Use a solid, directed line

DEPICTING RELATIONSHIPS: ASSOCIATION

- Two more forms of association
  - Aggregation (solid line, open diamond)
  - Composition (solid line, closed diamond)
- We won't use these (we'll just use the general association), but book might

DEPICTING RELATIONSHIPS: DEPENDENCY

- indicates a "uses" relationship
- Examples: `A` uses `B` if
  - `A` has method(s) with local variable of type `B`
  - `A` has method(s) with parameter of type `B`
  - `A` has method(s) with return type `B`
  - `A` has method(s) that invoke methods in `B`
- Use a dashed, directed line