Inheritance, Polymorphism, and Overriding

Weiss, from the beginning of chapter 4 through section 4.1.8 (pages 109–121)
extend property

• class \(D\) extends class B
  – An object of D has all the fields and methods specified by B.
  – The D object may have additional fields and methods.
  – B’s fields and methods do not have to be re-written and re-defined.
Type-compatibility

- class D is type-compatible with class B.
  - Student class extends Person class.
    Student s = new Student();
    Person p = s;

  - Student class re-defines toString() method of Person class.
    System.out.println( p.toString() );
Type-compatibility (cont.)

```java
public static boolean isOlder ( Person p1, Person p2 ) {
    return p1.getAge() > p2.getAge();
}

isOlder ( p, p )
isOlder ( s, s )
isOlder ( p, s )
isOlder ( s, p )
```
Compatibility of array types

- Arrays are type-compatible (covariant array types).

```java
Person [ ] arr = new Employee [ 5 ];
arr [ 0 ] = new Student( ... );
```
Super method and final class

• Invoke super in the constructor of a class that is derived from another class

```java
public Derived() {
    super();
}
```

• A final class is not extended.
• The methods of the final class are not overridden.
• Accessor and mutator methods should be declared as final.