

## Interfaces

Sec. 12.1 contains this material. Corresponding lectures on ProgramLive CD is an alternative way to learn the material.

### Top finalists from a real-life "Dilbert quotes contest"

As of tomorrow, employees will be able to access the building only using individual security cards. Pictures will be taken next Wednesday and employees will receive their cards in two weeks." (Fred Dales, Microsoft)

I need an exact list of specific unknown problems we might encounter. (Lykes Lines Shipping)

Email is not to be used to pass on information or data. It should be used only for company business. (Accounting manager, Electric Boat Company)

This project is so important, we can't let things that are more important interfere with it. (Advertising/Marketing manager, United Parcel Service)

Doing it right is no excuse for not meeting the schedule. (Plant manager, Delco Corporation)

1

## Interface: used to enforce that a class overrides certain methods

/\*\* Comparable provides a method for comparing two objects \*/

```
public interface Comparable {
```

```
    /** = a negative integer if this object < c,
```

```
        = 0 if this object = c,
```

```
        = a positive integer if this object > c.
```

```
    Throw a ClassCastException if c cannot be cast to the
```

```
    class of this object. */
```

```
    public int compareTo(Object c),
```

```
}
```

Body replaced by ;

Every class that "implements" Comparable must override compareTo

2

```
public class Date implements Comparable {
    public int year; /** year. */
    public int month; /** month, in the range 1..12. */
    public int day; /** The day of the month. */
    /** = -1, 0, or +1 if this date comes before, is same as, or comes after c.
     * Throw a ClassCastException if c cannot be cast to Date.*/
    public int compareTo(Object c) {
        if (!(c instanceof Date))
            throw new ClassCastException("argument is not a Date");
        Date d= (Date) c;
        if (year == d.year && month == d.month && day == d.day)
            return 0;
        if (year < d.year || (year == d.year && month < d.month) ||
            (year == d.year && month == d.month && day < d.day))
            return -1;
        return 1;
    }
}
```

class will contain other methods

3

## You can use an interface as a type

```
/** Swap b[i] and b[j] to put larger in b[j]
 * Use Comparable.compareTo to compare */
public static void swap(Comparable [] b, int i, int j) {
```

```
    if (b[j].compareTo(b[i]) < -1) {
        Comparable temp= b[i];
        b[i]= b[j];
        b[j]= temp;
    }
}
```

Polymorphism: the quality or state of existing in or assuming different forms

polygon  
polygamy  
polychord  
polydemic  
polytechnic

This parametric polymorphism allows us to use swap to do its job on any array whose elements implement Comparable.

4

## Interface java.util.Comparable

/\*\* Comparable requires method compareTo and method equals\*/

```
public interface Comparable {
```

```
    /** = a negative integer if this object < c,
```

```
        = 0 if this object = c,
```

```
        = a positive integer if this object > c.
```

```
    Throw a ClassCastException if c cannot be cast to the class of this object. */
```

```
    int compareTo(Object c);
```

```
}
```

Classes that implement Comparable  
Boolean  
Byte  
Double  
Integer  
...  
String  
BigDecimal  
BigInteger  
Calendar  
Time  
Timestamp  
...

5

## Form of an interface declaration

```
/** comment*/
```

```
public interface <interface-name> {
```

```
    /** method spec for function*/
```

```
    int compareTo(...);
```

```
    /** method spec for procedure */
```

```
    void compareTo(...);
```

```
    /** explanation of constant x*/
```

```
    int x= 7;
```

```
}
```

Methods are implicitly public. You can put the modifier on if you wish.

Every field is implicitly public, static, and final. You can put these modifiers on them if you wish.

6

A class can implement several interfaces

```
/** comment*/  
public class C implements Inter1, Inter2, Inter3 {  
    ...  
}
```

The class must override all methods declared in interfaces Inter1, Inter2, and Inter3.