

same/different

Is my

```
Cake ( icing = "Chocolate", layers=3, base="Chocolate")
```

the same as yours?

```
Cake ( icing = "Chocolate", layers = 1, base="Chocolate chip")
```

Does it matter?

Yes

Telling two objects apart is important for

- Testing
- Various applications
 - Do we already have this information?
 - Do two items have the same subproperties?
 - Have we already calculated what to do next, for this input?

sameas

| |
|--------------------------|
| Cake |
| String icing |
| int layers |
| String base |
| boolean sameAs(Cake c) |

```
class Cake {  
    String icing, base;  
    int layers;  
    Cake( String icing, int layers, String base ) {  
        this.icing = icing;  
        this.layers = layers;  
        this.base = base;  
    }  
    // To determine if this Cake is like c  
    boolean sameAs( Cake c ) {  
        // ... this.icing ... this.layers ... this.base ...  
    }  
}
```

sameas

| |
|--------------------------|
| Cake |
| String icing |
| int layers |
| String base |
| boolean sameAs(Cake c) |

```
class Cake {  
    String icing, base;  
    int layers;  
    Cake( String icing, int layers, String base ) {  
        this.icing = icing;  
        this.layers = layers;  
        this.base = base;  
    }  
    // To determine if this Cake is like c  
    boolean sameAs( Cake c ) {  
        // ... this.icing ... this.layers ... this.base ...  
        return this.icing.equals(c.icing)  
            && this.layers == c.layers  
            && this.base.equals(c.base);  
    }  
}
```

More complicated sameAs

| |
|--------------------------|
| Cake |
| String icing |
| int layers |
| String base |
| boolean sameAs(Cake c) |

| |
|----------------------------------|
| Lunchbox |
| Swandwich mainCourse |
| Beverage drink |
| Cake dessert |
| boolean sameAs(Lunchbox other) |

How to write sameAs for Lunchbox

equals

Testing equality very important

Method to compare two Objects built in to Java

| |
|----------------------------|
| Object |
| boolean equals(Object o) |
| String toString() |

| |
|----------------------------|
| String |
| boolean equals(Object o) |
| String toString() |
| ... |

Can we use equals to compare Lunchboxes?

Too general

Object -> Lunchbox ?

Object -> Sandwich ?

Object -> Cake ?

```
Object o = "Howdy";  
((String) o).equalsIgnoreCase("howdy")
```

casts: For those times when we know more than Java knows

casts

- queue.push(new Lunchbox(...))
- ((Lunchbox) queue.pop()).mainCourse
- List names = new Larger("Francis", new Empty())
- ((Larger) names).first

A little dangerous

```
Object o = new Lunchbox( ... )  
((String) o).equals(" Uh oh ")
```

?

Error when running the program

Living dangerously

Lets write equals(Object o) for Lunchbox

equals vs ==

- What do we expect new Object().equals(new Object()) to produce?
 - returns false
- What do we expect new Object() == new Object() to produce?
- Also false

Why?

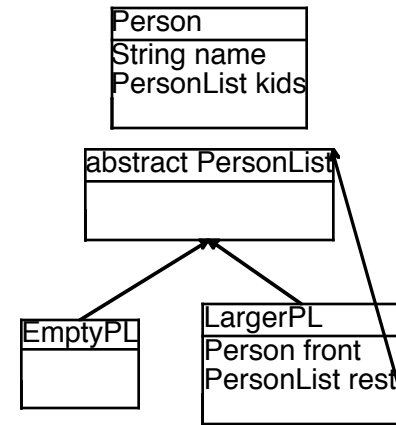
== on Objects

- Object o = new Object();
- o == o -> true
- == tests whether two references came from the same new
- Default .equals uses ==

Families

- A family tree
 - We started from the child and went to the parent
 - Let's start with the parent this time

Descendents



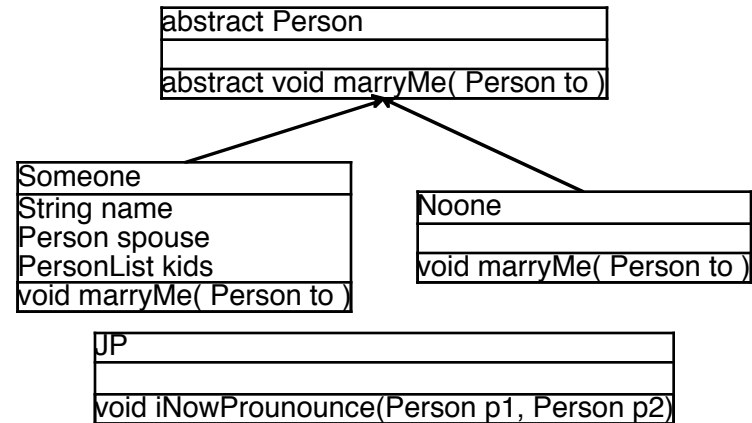
Descendents

`new Person("Jackson", new EmptyPL())`

- A Person with 3 kids

I now pronounce you ...

- People get married, and their spouse is part of the family
- Let's ammend our model (allowing for single people)



To DrScheme

Let's have another baby ...

