


Comp 170 / Isom 370 – Introduction to Object Oriented Programming

Dr. William L. Honig
Associate Professor
Department of Computer Science



KEY Concept



What is this
**OBJECT
ORIENTED**
Thing???



Objects



A Football in
Your Hand,
about to be
thrown

Brad Pitt

Your pink
iPOD,
loaded with
357
downloads

A small white
dog named
Dolly

Angelina Jolie

The Oak tree
outside CS
on east lawn
on LSC

Your bed in
your room

- Real world things; often physical
- Want or need to use inside software
- Can interact with others
- Usually have "lifetimes"

© 2006 William L. Honig, Ph.D. Introduction to Programming

3



1.16 Software Engineering Case Study (Cont.)

4

• Objects

- Reusable software components that model real-world items
- Look all around you
 - People, animals, plants, cars, etc.
- Attributes
 - Size, shape, color, weight, etc.
- Behaviors
 - Babies cry, crawl, sleep, etc.



© 2005 Pearson Education, Inc. All rights reserved.

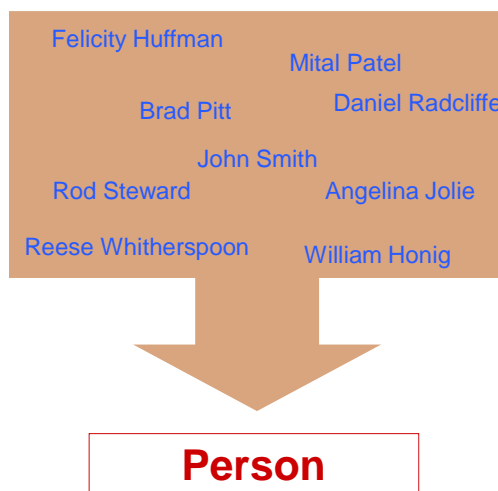
Object, aka...



- The object concept is related to
 - Instance
 - Class Instance
 - Variable
 - Attribute value (or set of attribute values)
 - Field value (or set of field values)



Class



- Abstract a bunch of objects of the same kind
- Model related objects (need not be identical objects)
- Used inside a computer program
- A “Factory” for objects



Class, aka...



- The class concept is also known as
 - Type
 - Abstract Type
- The class concept also involves common
 - Field or attribute **definitions** (types)
 - Behaviors
 - Methods
 - Operations



More Class.....



- Object Oriented Programming
 - Class used to model something from the real world
 - Class = representation of all possible objects of a certain kind
 - Curiosity – How to computers do those things????
- Classes are ******NOT****** specific things
 - A specific person object has a name = Angelina Jolie
 - The class Person includes the concept of name in general. All possible Persons have a name.
- Angelina Jolie is an instance of the class Person
- A single class can have 0, 1, or many instances



Why Object Technology?



“The value of OT (OOA&D, OOP) primarily lies in its ability to handle complex problems and create comprehensible, manageable systems that can scale up to increasing complexity, and that are easily adaptable—if designed skillfully.”

Craig Larman

1. Elegantly tackle complexity & create easy adaptability.

2. . . .

9. Productivity

10. Reuse

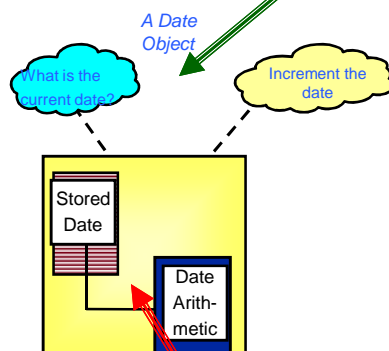
The productivity is realized in the maintenance or modification phases of a system—often with profoundly faster changes, if the system was designed skillfully.

Dr. William L. Honig
Copyright 2002

COMP 272 Spring 2002

What is an Object?

A public, external view that can be seen and operated on by others



Examples:

My Birthday
Assignment Due Date
Tomorrow

A private, hidden, secret way that things are done "inside" the object

- > An object is a software bundle of related variables and functions.
- > Software objects often used to model real world objects.
- > Objects are typically the nouns you use to describe a system or process or how something is to work.
- > Objects can store information – the objects current state.
- > Objects define or implement the functions they can perform.

Examples

- > A date (for calendar or schedule)
- > A document
- > An author of a document

Dr. William L. Honig
Copyright 2002

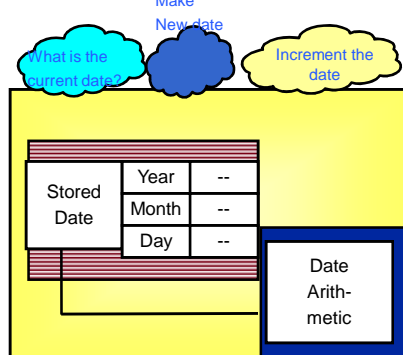
COMP 272 Spring 2002

Implementing Objects in Software (Key Concept #1)



Software Factory to Manufacture Date Objects

- * Knows what is shared by all Dates
- * Knows what to make unique to each Date



- * Software implementations will need many objects of the same type.
- * All these objects have common operation and behavior.
- * The objects may come and go at different times, independently of each other.
 - > It makes sense to associate them all together
 - > More efficient (computer storage)
 - > Especially if we think something might change
 - > Especially if we wish others to use the objects without knowing about their private, inside operation

Dr. William L. Honig
Copyright 2002

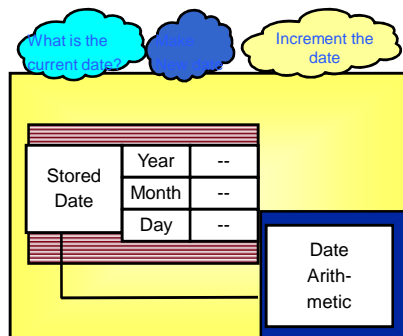
COMP 272 Spring 2002

The Factory in Operation (Key Concept #1)



Create a Date Object – Assignment Due Date

Create another Date Object –



- * All date objects created by the factory
- * So they all act the same way
- * Make New Date creates the unique parts or content for each object
- * All Date objects stay associated with the factory that made them

What is this date?
1949 | Oct | 1

Assignment Due Date
2001 | Sept | 3

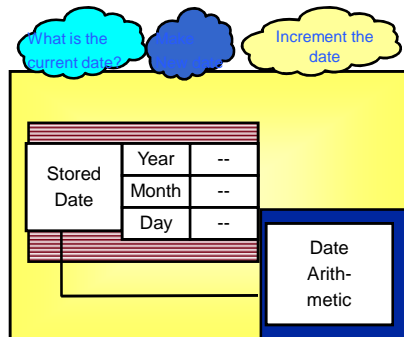
Dr. William L. Honig
Copyright 2002

COMP 272 Spring 2002

Object Class and Object Instance (Key Concept #1)



The Factory is called a **CLASS**



Each specific **OBJECT** or Instance (of the class) is created by the factory

What is this date?

| | | |
|------|-----|---|
| 1949 | Oct | 1 |
|------|-----|---|

Assignment Due Date

| | | |
|------|------|---|
| 2001 | Sept | 3 |
|------|------|---|

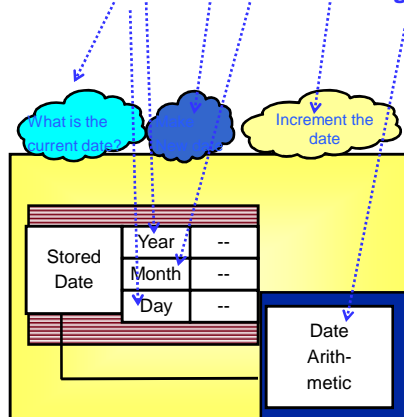
Dr. William L. Honig
Copyright 2002

COMP 272 Spring 2002

So, Why Do I Care? Object Oriented Programming



Must Write Code for all these things



Would it not be easier just to write and use a good data type?

```
enum type_month {jan,feb,mar,
apr,may,jun,jul,aug,sep,
oct,nov,dec
```

```
};
struct date {
int year;
type_month month;
int day
```

What is this date?

| | | |
|------|-----|---|
| 1949 | Oct | 1 |
|------|-----|---|

Assignment Due Date

| | | |
|------|------|---|
| 2001 | Sept | 3 |
|------|------|---|

Dr. William L. Honig
Copyright 2002

COMP 272 Spring 2002