



Object-Oriented Programming

Unit 02



Section Goals

- To understand:
 - Benefits of object-oriented programming
 - Basic concepts and terminology of object-oriented programming
 - The Java object model

Benefits of Object Technology

- Faster application development at a lower cost
- Decreased maintenance time
- Less complicated and faster customization
- Higher quality code

What is Object-Oriented Programming?

- A set of implementation techniques that:
 - Can be done in any programming language
 - May be very difficult to do in some programming languages
- A strong reflection of software engineering
 - Abstract data types
 - Information hiding (encapsulation)
- A way of encouraging code reuse
 - Produces more malleable systems
- A way of keeping the programmer in touch with the problem
 - Real world objects and actions match program objects and actions

Objects

- Objects are:
 - Are building blocks for systems
 - Contain data that can be used or modified
 - Bundle of variables and related methods
- An object possesses:
 - Identity
 - A means of distinguishing it from other objects
 - State
 - What the object remembers
 - Interface
 - Messages the object responds to
 - Behavior
 - What the object can do

Object Example

- The car shown in the figure can be considered an object. It has an ID (“1”), state (its color, for instance, and other characteristics), an interface (a steering wheel and brakes, for example) and behavior (the way it responds when the steering wheel is turned or the brakes are applied).
- Many texts regard an object as possessing only two characteristics – state and behavior. When considered this way, identity is part of the state, and the interface is included in the behavior.

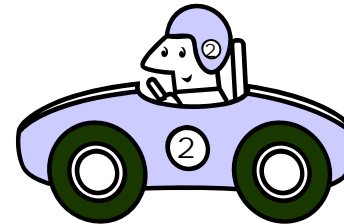
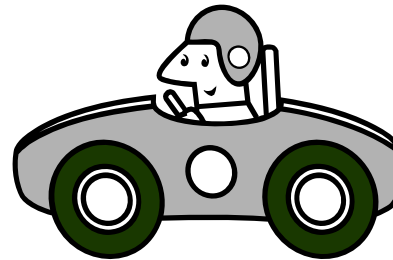


Classes

- A class
 - Defines the characteristics and variables common to all objects of that class
- Objects of the same class are similar with respect to:
 - Interface
 - Behavior
 - State
- Used to instantiate (create an instance of) specific objects
- Provide the ability of reusability
 - Car manufacturers use the same blueprint to build many cars over and over

Class Example

- The car at the top of the figure represents a class – notice that the ID and color (and presumably other state details) are not known, but the interface and behavior are.
- Below the “class” car are two objects which provide concrete installations of the class



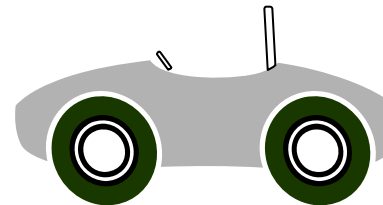
Message and Object Communication

- Objects communicate via messages
- Messages in Java correspond to method calls (invocations)
- Three components comprise a message:
 1. The object to whom the message is addressed (Your Car)
 2. The name of the method to perform (changeGears)
 3. Any parameters needed by the method (lower gear)



Object Messaging Example

- By itself the car is incapable of activity. The car is only useful when it is interacted with by another object
- Object 1 sends a message to object 2 telling it to perform a certain action
- In other words, the driver presses the car's gas pedal to accelerate.



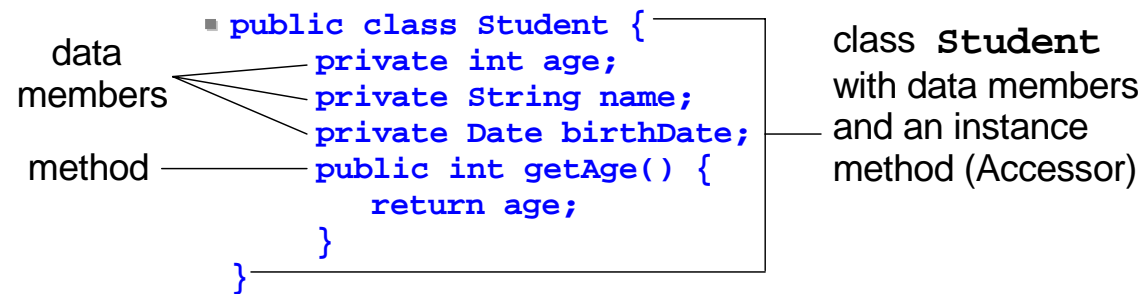
Accessing State

- State information can be accessed two ways:
 - Using messages:
 - Eliminates the dependence on implementation
 - Allows the developer to hide the details of the underlying implementation
 - “Accessor” messages are usually used instead of accessing state directly
 - Example: `getSpeed()` may simply access a state value called “speed,” or it could hide (or later be replaced by) a calculation to obtain the same value

OO in Java

- Language elements:
 - Class-based object-oriented programming language with inheritance
 - A class is a template that defines how an object will look and behave once instantiated
- Java supports both instance and class (static) variables and methods
- Nearly everything is an object
 - They are accessed via references
 - Their behavior can be exposed via public methods
 - They are instantiated using the `new` construct

Java Classes



What You Have Learned

- How objects store information and interact with each other
- The benefits of object technology
- How object-oriented programming relates to Java

Homework #2

- 1. What does the Java Virtual Machine (JVM) do?
- 2. What is **bytecode**?
- 3. What is the difference between a **program** and a **component**?
- 4. What are the differences between an **application**, an **applet** and a **servlet**?
- 5. p.32 for the self review exercises hand in exercises 1.2 and 1.3 yes I know the answers are given to you but you have to know this stuff and typing it out will help you remember.
- 6. Type the program into a text editor on P.37
 - send me the compiled class file

Homework #2

7. Do the exercise starting on p.17 the ATM example from the book

Example from me doing it:

The first time I tried to run the program, I received the following exception:

```
C:\$downloads\iup\deitel small Java how to program\examples\ch01\ATM>java ATMCaseStudy
Exception in thread "main" java.lang.NoClassDefFoundError: ATMCaseStudy
```

To fix this exception, I issued the following command:

```
C:\$downloads\iup\deitel small Java how to program\examples\ch01\ATM>set
CLASSPATH=%CLASSPATH%;C:\$downloads\iup\deitel small Java how to program\examples\ch01\ATM
```

The second attempt at running the program worked:

```
C:\$downloads\iup\deitel small Java how to program\examples\ch01\ATM>java ATMCaseStudy
```

Welcome!

Please enter your account number: 12345

Enter your PIN: 54321

Main menu:

1 - View my balance

2 - Withdraw cash

3 - Deposit funds

4 - Exit

Enter a choice: 1

Balance Information:

- Available balance: \$1,000.00

- Total balance: \$1,200.00

Main menu:

1 - View my balance

2 - Withdraw cash

3 - Deposit funds

4 - Exit

Enter a choice:

CAPTURE A SCREEN SHOT OF ANY PART OF THIS SMALL APPLICATION RUNNING AND HAND IT IN

Homework #2

8. Create a Java Project. Explore the Eclipse IDE. Fill in the following to create a program that prints out "Hello, World".

```
public class HelloWorld{  
  
    public static void main(String[] args){  
  
        _____;  
    }  
}
```

Test your answer in Eclipse.

CAPTURE A SCREEN SHOT OF ANY PART OF THIS SMALL APPLICATION RUNNING AND HAND IT IN

9. What is a client?

10. What characteristics do objects possess?