Introduction to Java II

Overview

- Today we will cover Java identifiers, classes and objects.
- We will also review the object-oriented concept of inheritance.
- We will apply what we learn in an example Java program that will be compiled and run from the Eclipse IDE and command-line.

Introduction to Java

- In the Java programming language
  - a program is made up of one or more *classes*
  - a class contains one or more *methods*
  - a method contains program *statements*
- These terms will be explored in detail throughout the course
- A Java application always contains a method called main

Introduction to Java

Identifiers

- Identifiers are the words a programmer uses in a program
- Java is case sensitive
  - Total, total, and TOTAL are different identifiers
- We use special identifiers called reserved words that already have a predefined meaning in the language.
  - A reserved word cannot be used in any other way

Introduction to Java

Reserved Words

abstract	else	interface	switch
assert	enum	long	synchronized
boolean	extends	native	this
break	false	new	throw
byte	final	null	throws
case	finally	package	transient
catch	float	private	true
char	for	protected	try
class	goto	public	void
const	if	return	volatile
continue	implements	short	while
default	import	static	
do	instanceof	strictfp	
double	int	super	

Introduction to Java – Classes

- An object is defined by a *class*
- A class is the blueprint of an object. The class uses methods to define the behaviors of the object
- The class that contains the **main** method of a Java program represents the entire program
- A class represents a concept, and an object represents the embodiment of that concept
- Multiple objects can be created

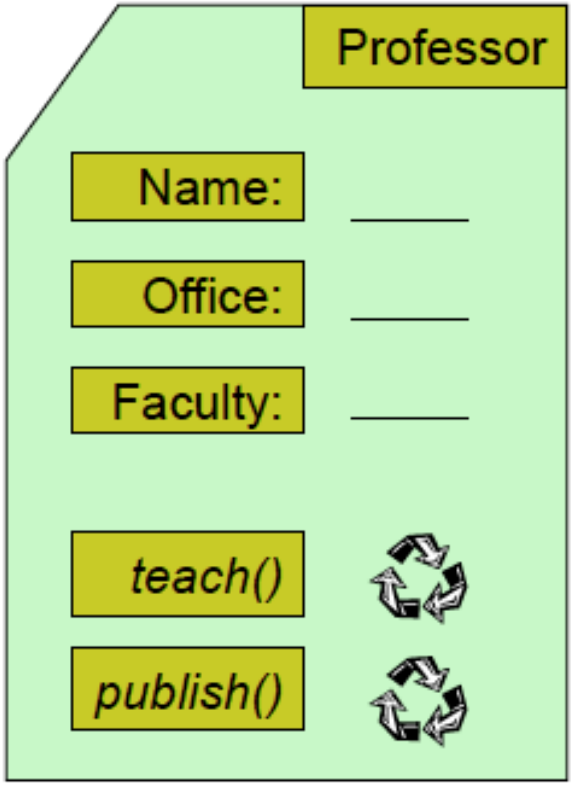
Introduction to Java – Classes

Classes:

Blueprint of the "layout" of objects.
- Members (data)
- Methods (code)

Writing classes involves
- Declare the members
- Declare the methods
- Write code to implement the methods

Classes are created when you write your Java code.



Introduction to Java – Objects

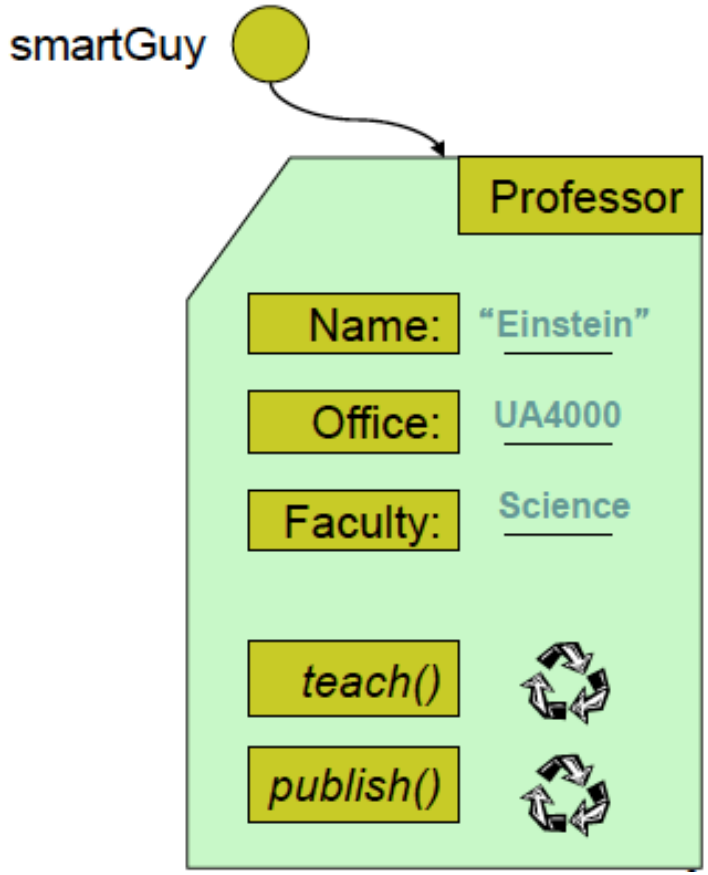
- Java code can create objects by
  - **Directly**: invoke constructors that are available to a certain class.
  - **Indirectly**: invoke a method which will in turn create an object and return that object.
- The only way to access Java objects is by variables which are pointers to the object.

Introduction to Java – Objects

Objects:

Variables of a class containing
- members (data)
- Methods (code)

Objects are created when you run your program. This is called "run-time".



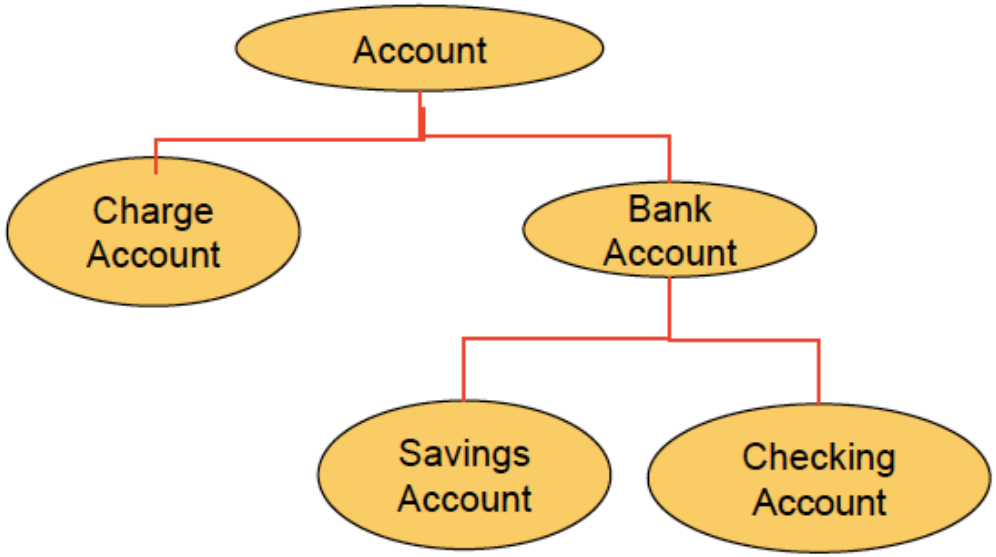
Introduction to Java

- Let's now look at an example!

> See FirstProgram.java and Triangle.java for a first example of a Java program and Java classes.

OO Principles - Inheritance

- One class can be used to derive another via inheritance
- Classes can be organized into hierarchies



Introduction to Java II

Summary

- Today we covered Java identifiers, classes and objects and reviewed inheritance.
- We also learned how to compile and run a Java program from the Eclipse IDE and command-line.

Next time

- We will talk about Java data types, variables and expressions