

CMPT 213

Object Oriented Design in Java

<http://www.cs.sfu.ca/CourseCentral/213/bobbyc/>

- Course Introduction

Course Information

Introduction

- People
 - Instructor: Bobby Chan (bobbyc@cs.sfu.ca)
 - TA: Tirth Patel (tirthp@sfu.ca)
 - TA: Aditi Shrivastava (ashrivas@sfu.ca)
- Webpage
www.cs.sfu.ca/CourseCentral/213/bobbyc/
- Recommended Textbook:
Object-Oriented Design & Patterns, 2nd ed. Cay Horstmann
On 24H reserve.

3

Introduction

- Learning outcomes

Goals:

- Learn to write good quality object-oriented programs in Java.
- Three main components: ([Ex 0-0](#))
 1. Tools
 2. Design
 3. Standards

4

Introduction

- Prerequisites

From courses \leq CMPT 225, it's assumed that you are proficient at the basic concepts of programming.

- Simple Object Oriented Environment
- Using IDEs
- Python, C, C++
- Experience with the [develop \rightarrow test \rightarrow debug] cycle

5

Introduction

- Tips to do well

- Read materials before class
- Take notes on examples
- Ask questions during class
- Do assignments pace yourselves, they are very lengthy ...
 - Do them by yourself, if you need assistance, please see me or the TA. That's why we're here!
- Do not multitask

6

Introduction

- Quick Introduction

Motivation questions: (Ex 0-1)

1. Why do we write comments?
2. Why **Java** when we already know C++?
3. What is (good/bad) **code smell**?

7

What is Java?

Features of the Java Programming Language

- **Java**

- Originally Developed by James Gosling at Sun Microsystems (acquired by Oracle)
- First public implementation as Java 1.0 in 1995
- An object-oriented language
- General-purpose

[https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))

9

Features of the Java Programming Language (cont'd.)

- **Java (cont'd.)**

- Can be run on a wide variety of computers
- Does not execute instructions on the computer directly
- Runs on a “hypothetical computer”
- Many other programming languages also on this machine

- **Source code**

- Programming statements written in high-level programming language

10

Features of the Java Programming Language (cont'd.)

- **Development environment**
 - A set of tools used to write programs, NetBeans, Eclipse
- **Bytecode**
 - Statements saved in a file
 - A binary program into which the Java compiler converts source code
- **Java interpreter**
 - Checks bytecode and communicates with the operating system
 - Executes bytecode instructions line by line within the Java Virtual Machine

11

Java Program Types

- **Applets**
 - Programs embedded in a Web page
- **Java applications**
 - Called Java stand-alone programs
 - **Console applications**
 - Support character output
 - **Windowed applications**
 - Menus
 - Toolbars
 - Dialog boxes

12

Compiling a Java Class

- Save the class in a file with exactly the same name and .java extension
 - For public classes, class name and filename must match exactly
- Compiling a Java class
 - Compile the source code into bytecode
 - Translate the bytecode into executable statements
 - Using a Java interpreter
 - Type `javac First.java`
- Compilation outcomes
 - `javac` is an unrecognized command
 - Program language error messages
 - No messages indicating successful completion
- Run the application from the command line
 - `java First`
- Shows the application's output in the command window