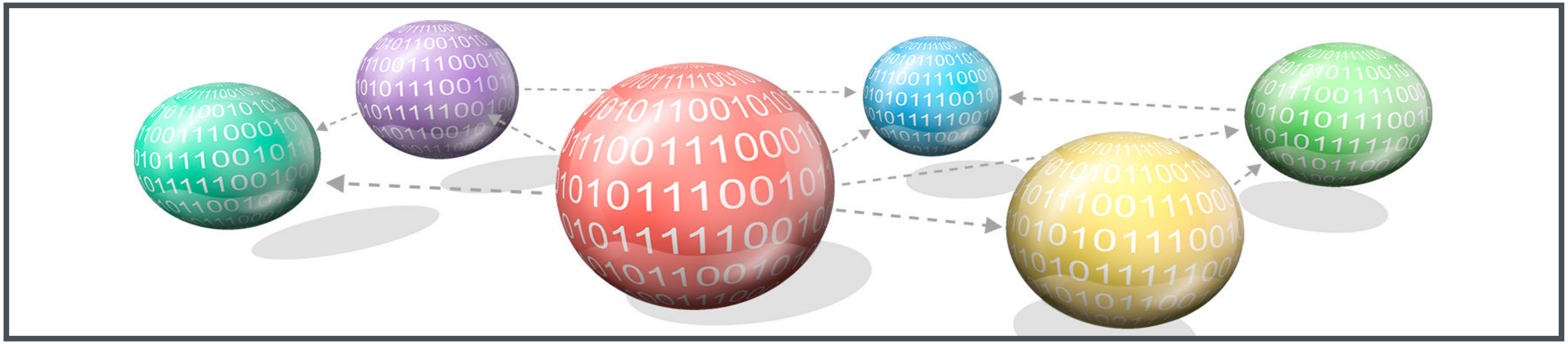
NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

Introduction

SC/CE/CZ2002 Object Oriented Design & Programming

Dr Shen Zhiqi & Dr Li Fang



Contact Information

First-half: Dr Shen Zhiqi

- Office: N4-2B-43; Phone: 6790-6250
- Email: zqshen@ntu.edu.sg
- Consultation: (Thursday 4:30pm-5:30pm)
- Appointment via email

Second-half: Dr Li Fang

- Office: N4-2B-42; Phone: 6790-6106
- Email: asfli@ntu.edu.sg
- Consultation: ()
- Appointment via email

Course Objectives

- **Pre-requisites:**
  - CE1007 and CZ1007 - Data Structures
  - Knowledge of C Programming Language

- **Objectives:**
  - Explain and articulate Object-Oriented Methodology and Concepts.
  - Develop OO program using OO Language.
    1. Use OO design principles to design a program.
    2. Implement an OO design using OO Language.

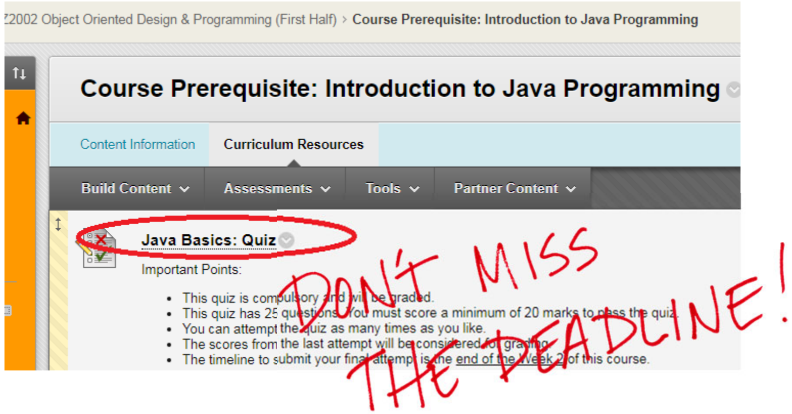
Java C++

Course Structure

- Lectures (from Week 1)
- Online MCQ Java Quiz (Week 1-2)
- Tutorials (Week 4 onwards, 9 Tutorials
- Labs (Week 3-odd/ Week 4-even onwards)
  - 5 labs in total; 2 hours each (refer to NTULearn for new schedule)
  - First 2 labs to familiarize with Java syntax knowing C syntax
  - 3rd and 4th lab on OO concepts and usage (4th lab is also for assignment discussion)
  - 5th lab on C++

Course Assessment Components

- Final Exam: No
- Quizzes: 50% of final marks
  - 5% Online Quiz on Basic Java Syntax
    - Before end of week 2 to complete it
  - 45% Physical
    - One in Week 7
    - One in Week 13
- Coursework: 50% of final marks
  - 10% Lab Completion
    - Attendance is also part of assessment marks
  - 40% Group Assignment
    - Publish after recess
    - Group sign-up will be released later
      - ➢ The students are to form their groups (4-5 students per group).
      - ➢ The sites probably will be ready after week 3



Z2002 Object Oriented Design & Programming (First Half) › Course Prerequisite: Introduction to Java Programming
Content Information
Curriculum Resources
Build Content
Assessments
Tools
Partner Content

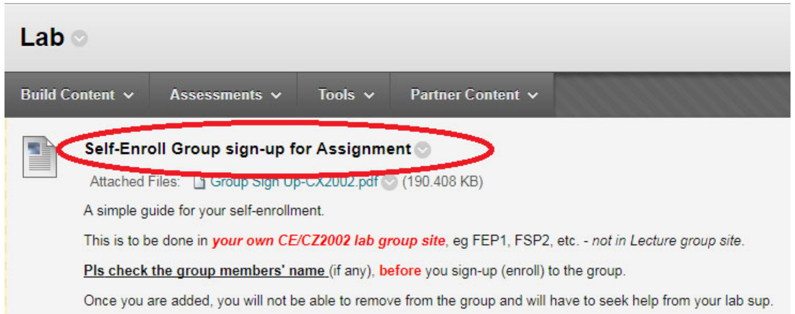
Course Prerequisite: Introduction to Java Programming

Java Basics: Quiz

Important Points:

- This quiz is compulsory and will be graded.
- This quiz has 25 questions. You must score a minimum of 20 marks to pass the quiz.
- You can attempt the quiz as many times as you like.
- The scores from the last attempt will be considered for grading.
- The timeline to submit your final attempt is the end of the Week 2 of this course.

DON'T MISS THE DEADLINE!



Lab

Self-Enroll Group sign-up for Assignment

Attached Files: Group Sign Up-CX2002.pdf (190.408 KB)

A simple guide for your self-enrollment.

This is to be done in ***your own CE/CZ2002 lab group site***, eg FEP1, FSP2, etc. - *not in Lecture group site*.

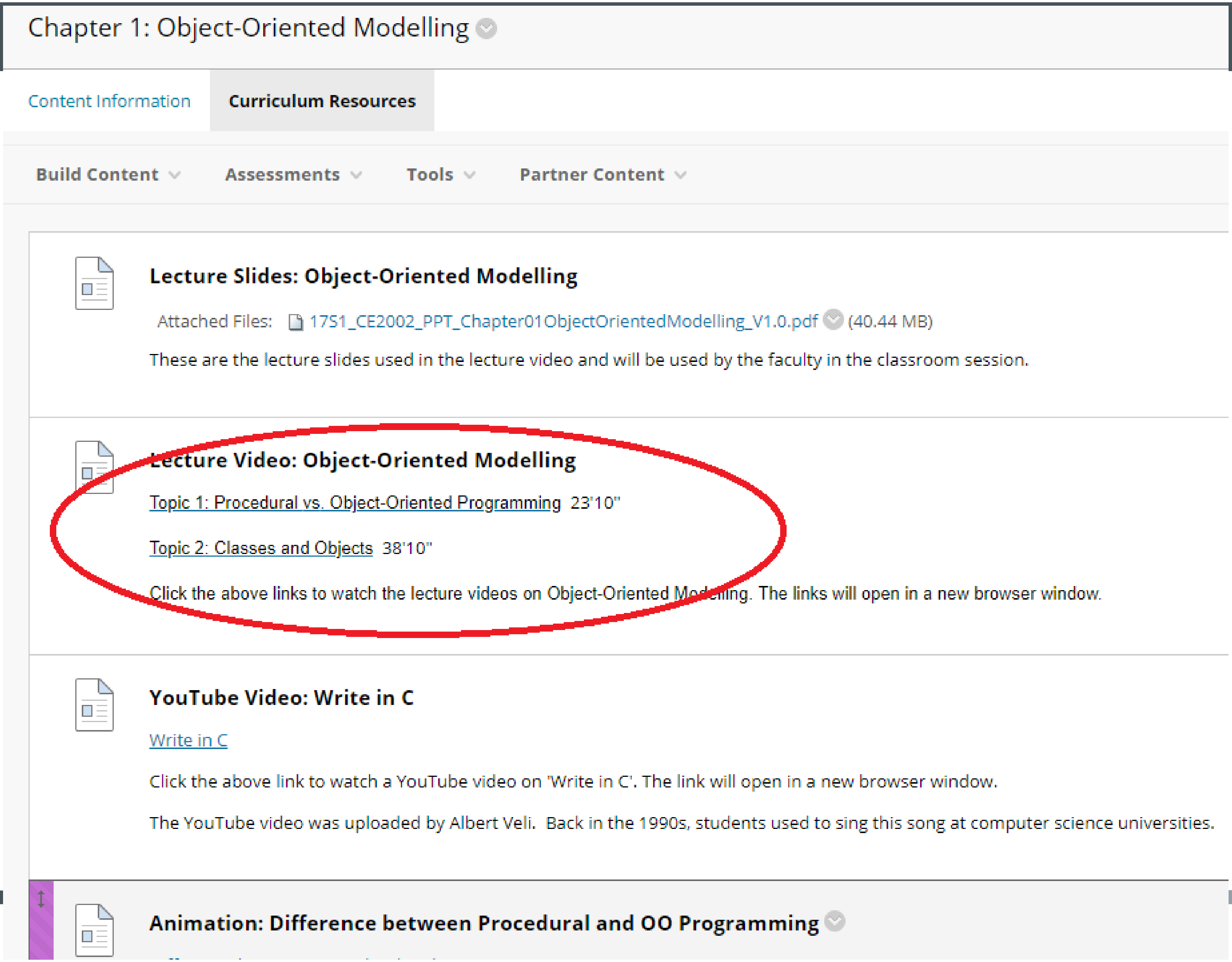
Pls check the group members' name (if any), **before** you sign-up (enroll) to the group.

Once you are added, you will not be able to remove from the group and will have to seek help from your lab sup.

Technology-Enhanced Learning (TEL)

- Own pace
- Own time
- Own place
- More interaction and examples in face-to-face sessions

- Technology-Enhanced Learning (TEL) courses are conducted with a mix of online and face-to-face sessions.
- Participants access and study course materials online before meeting classmates and their professors in face-to-face discussions and brainstorming sessions.



Chapter 1: Object-Oriented Modelling

Lecture Slides: Object-Oriented Modelling

Attached Files: 17S1_CE2002_PPT_Chapter01ObjectOrientedModelling_V1.0.pdf (40.44 MB)

These are the lecture slides used in the lecture video and will be used by the faculty in the classroom session.

Lecture Video: Object-Oriented Modelling

Topic 1: Procedural vs. Object-Oriented Programming 23'10"

Topic 2: Classes and Objects 38'10"

Click the above links to watch the lecture videos on Object-Oriented Modelling. The links will open in a new browser window.

YouTube Video: Write in C

Write in C

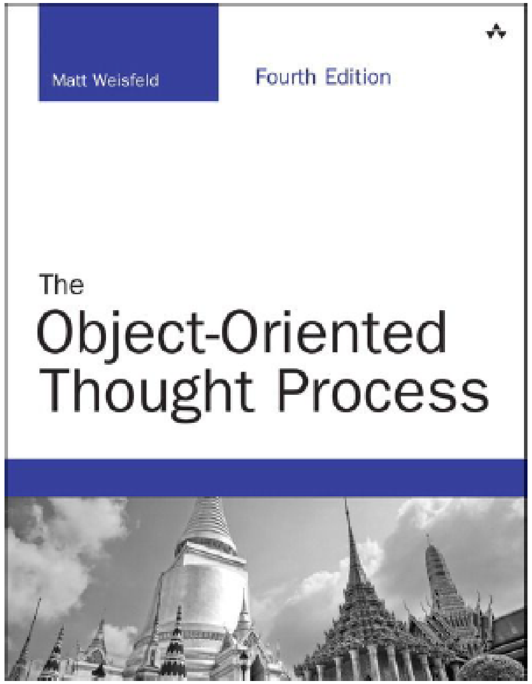
Click the above link to watch a YouTube video on 'Write in C'. The link will open in a new browser window.

The YouTube video was uploaded by Albert Veli. Back in the 1990s, students used to sing this song at computer science universities.

Animation: Difference between Procedural and OO Programming

Course Reference Books

- Text: Matt Weisfeld, The Object-Oriented Thought Process, 4th Edition, Pearson/ Addison Wesley.
- Call No. QA76.64.W427 2013, Lee Wee Nam Library
- Electronic copy, 3rd Edition, search title on the library website



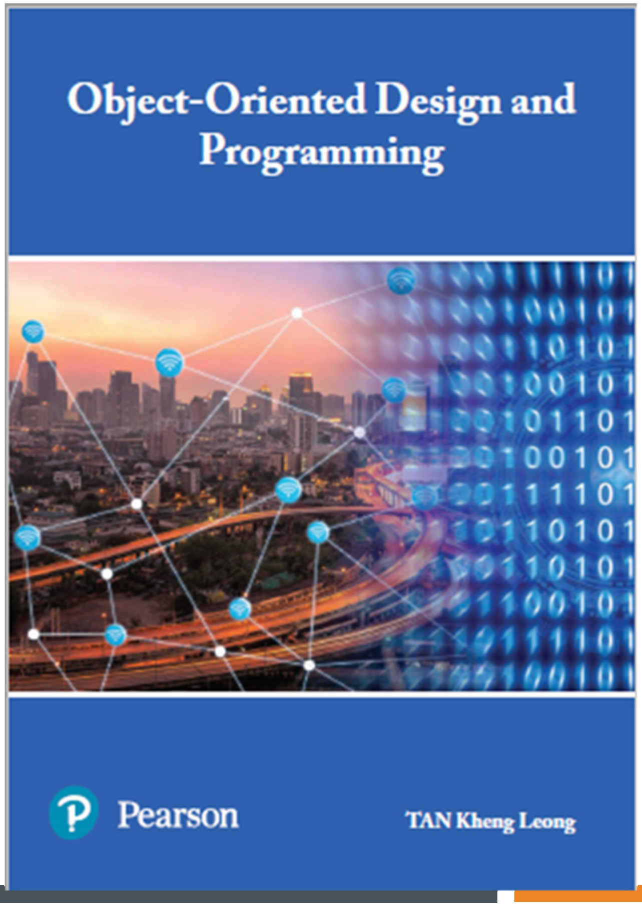
Pearson

Object-Oriented Thought Process, The, 4/E

Author	: Weisfeld
Publisher	: Pearson
ISBN	: 9780321861276

Available now at Booklink NTU.

CZ2002 OBJECT ORIENTED DESIGN & PROGRAMMING



Object-Oriented Design and Programming

Pearson

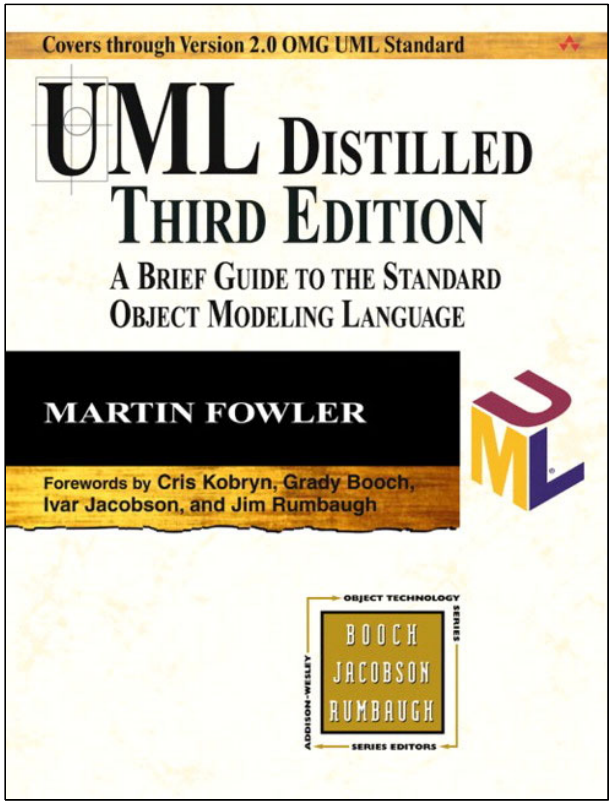
TAN Kheng Leong

- OBJECT ORIENTED DESIGN & PROGRAMMING

Author	: NTU - TAN
Publisher	: Pearson
ISBN	: 9789813133815

Books available at Booklink NTU

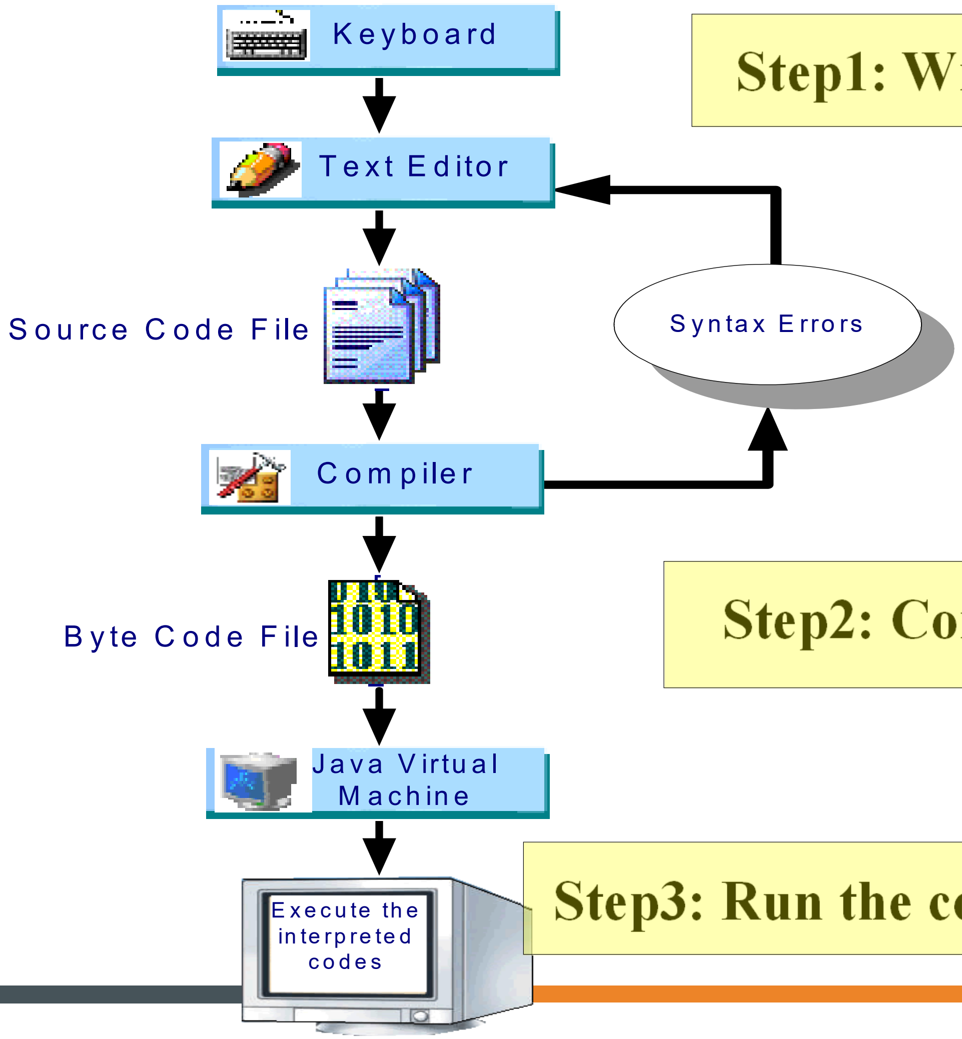
CZ2002 OBJECT ORIENTED DESIGN & PROGRAMMING


Covers through Version 2.0 OMG UML Standard
UML Distilled
Third Edition
A Brief Guide to the Standard Object Modeling Language
Martin Fowler
Forewords by Cris Kobryn, Grady Booch, Ivar Jacobson, and Jim Rumbaugh
Object Technology Series
Booch Jacobson Rumbaugh
Series Editors
Addison-Wesley

- UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3/E

Author	: Fowler
Publisher	: Pearson
ISBN	: 9780321193681

Books available at Booklink NTU



Keyboard

Text Editor

Source Code File

Syntax Errors

Compiler

Byte Code File

Java Virtual Machine

Execute the interpreted codes

Step1: Write the Java source code

Step2: Compile the Java source code

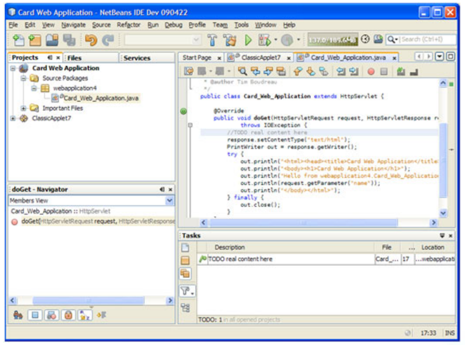
Step3: Run the compiled Java code or class file

A Simple Java Program

```
//This program prints Welcome to Java!
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```

Java Development tools

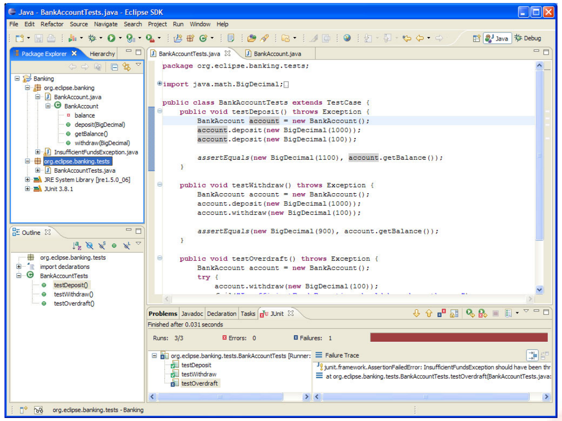
- You can use a text editor, to create Java Programs and to compile and run the programs from the command windows.
- You can use a Java development tool
  - NetBeans, Eclipse
  - Integrated development environment(IDE)
  - quick, effective
  - Self-study tutorials
    - Java Fundamentals I- Introduction to NetBeans IDE, Part 1
    - https://www.youtube.com/watch?v=Hv2yvXTVTVo
    - Eclipse IDE Tutorial:
    - https://www.youtube.com/watch?v=23tAK5zdQ9c



Card Web Application - NetBeans IDE Dev 090422
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects
Files
Services
Start Page
ClassicApplet7
Card_Web_Application.java
doGet - Navigator
Tasks

```
 * @author Tim Boudreau
 */
public class Card_Web_Application extends HttpServlet {

    @Override
    public void doGet(HttpServletRequest request, HttpServletResponse r
            throws IOException {
        //TODO real content here
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        try {
            out.println("<html><head><title>Card Web Application</title
            out.println("<body><h1>Card Web Application</h1>");
            out.println("Hello from webapplication4.Card_Web_Applicatio
            out.println(request.getParameter("name"));
            out.println("</body></html>");
        } finally {
            out.close();
        }
```

	Description	File	...	Location
	TODO real content here	Card_...	17	...webapplicati



```
package org.eclipse.banking.tests;

import java.math.BigDecimal;

public class BankAccountTests extends TestCase {
    public void testDeposit() throws Exception {
        BankAccount account = new BankAccount();
        account.deposit(new BigDecimal(1000));
        account.deposit(new BigDecimal(100));

        assertEquals(new BigDecimal(1100), account.getBalance());
    }

    public void testWithdraw() throws Exception {
        BankAccount account = new BankAccount();
        account.deposit(new BigDecimal(1000));
        account.withdraw(new BigDecimal(100));

        assertEquals(new BigDecimal(900), account.getBalance());
    }

    public void testOverdraft() throws Exception {
        BankAccount account = new BankAccount();
        try {
            account.withdraw(new BigDecimal(100));
```

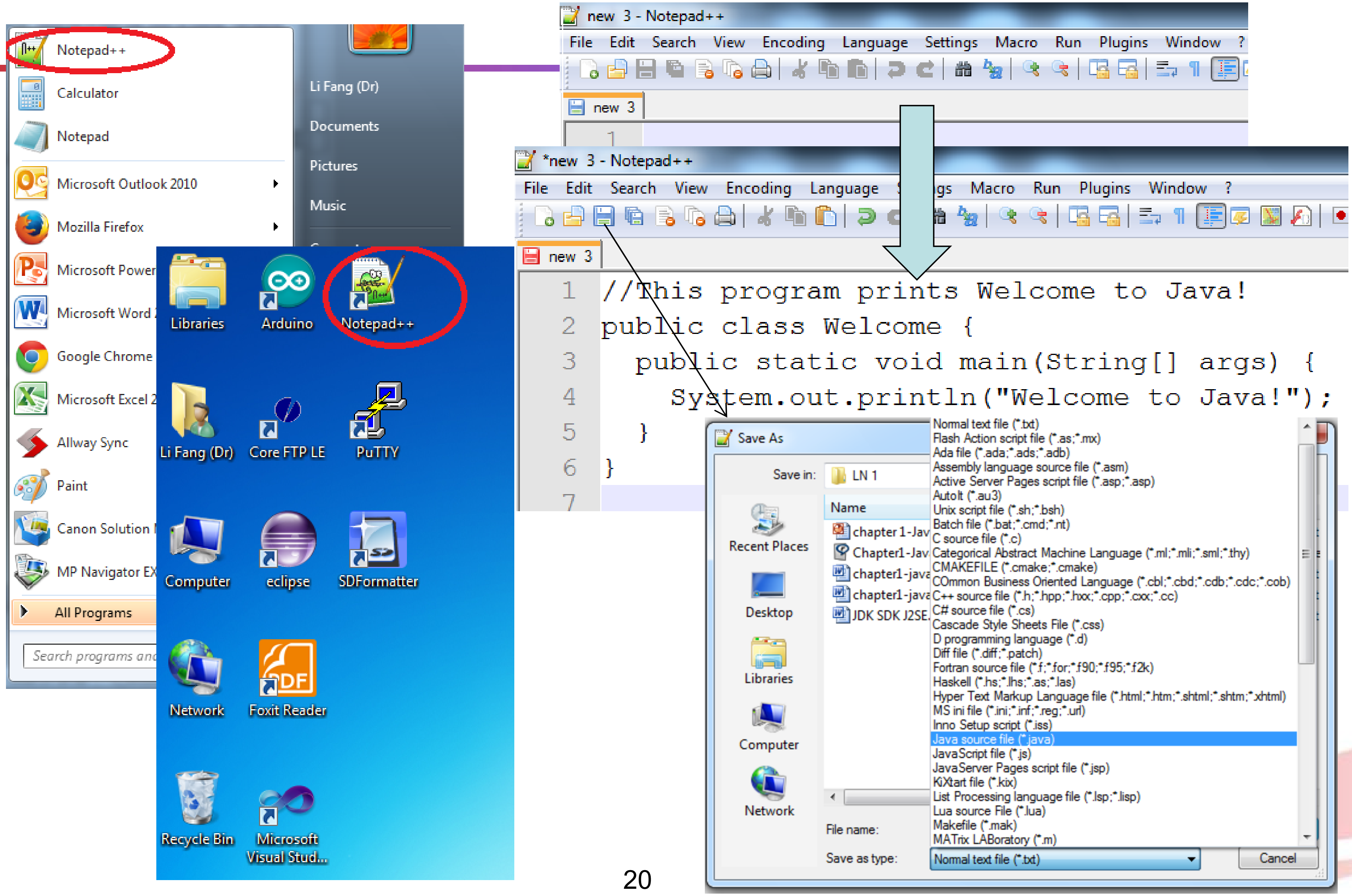
Finished after 0.031 seconds

Runs: 3/3 Errors: 0 Failures: 1

org.eclipse.banking.tests.BankAccountTests [Runner:
testDeposit
testWithdraw
testOverdraft

junit.framework.AssertionFailedError: InsufficientFundsException should have been thr
at org.eclipse.banking.tests.BankAccountTests.testOverdraft(BankAccountTests.java:

Step 1:Creating and Editing Using NotePad++

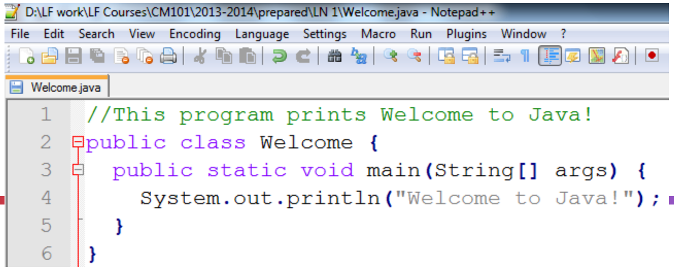


```
//This program prints Welcome to Java!
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```

Save as type: Java source file (*.java)

Notepad++
Calculator
Notepad
Microsoft Outlook 2010
Mozilla Firefox
Microsoft Power
Microsoft Word
Google Chrome
Microsoft Excel
Allway Sync
Paint
Canon Solution
MP Navigator EX
All Programs
Search programs and
Li Fang (Dr)
Documents
Pictures
Music
Libraries
Arduino
Notepad++
Li Fang (Dr)
Core FTP LE
PuTTY
Computer
eclipse
SDFormatter
Network
Foxit Reader
Recycle Bin
Microsoft Visual Stud...
new 3 - Notepad++
*new 3 - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
new 3
Save As
Save in: LN 1
Name
chapter1-Jav
Chapter1-Jav
chapter1-java
chapter1-java
JDK SDK J2SE
Recent Places
Desktop
Libraries
Computer
Network
File name:
Save as type:
Normal text file (*.txt)
Cancel
Normal text file (*.txt)
Flash Action script file (*.as;*.mx)
Ada file (*.ada;*.ads;*.adb)
Assembly language source file (*.asm)
Active Server Pages script file (*.asp;*.asp)
AutoIt (*.au3)
Unix script file (*.sh;*.bsh)
Batch file (*.bat;*.cmd;*.nt)
C source file (*.c)
Categorical Abstract Machine Language (*.ml;*.mli;*.sml;*.thy)
CMAKEFILE (*.cmake;*.cmake)
COmmon Business Oriented Language (*.cbl;*.cbd;*.cdb;*.cdc;*.cob)
C++ source file (*.h;*.hpp;*.hxx;*.cpp;*.cxx;*.cc)
C# source file (*.cs)
Cascade Style Sheets File (*.css)
D programming language (*.d)
Diff file (*.diff;*.patch)
Fortran source file (*.f;*.for;*.f90;*.f95;*.f2k)
Haskell (*.hs;*.lhs;*.as;*.las)
Hyper Text Markup Language file (*.html;*.htm;*.shtml;*.shtm;*.xhtml)
MS ini file (*.ini;*.inf;*.reg;*.url)
Inno Setup script (*.iss)
Java source file (*.java)
JavaScript file (*.js)
JavaServer Pages script file (*.jsp)
KiXtart file (*.kix)
List Processing language file (*.lsp;*.lisp)
Lua source File (*.lua)
Makefile (*.mak)
MATrix LABoratory (*.m)

Creating, Compiling, and Running Programs


D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1\Welcome.java - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
Welcome.java

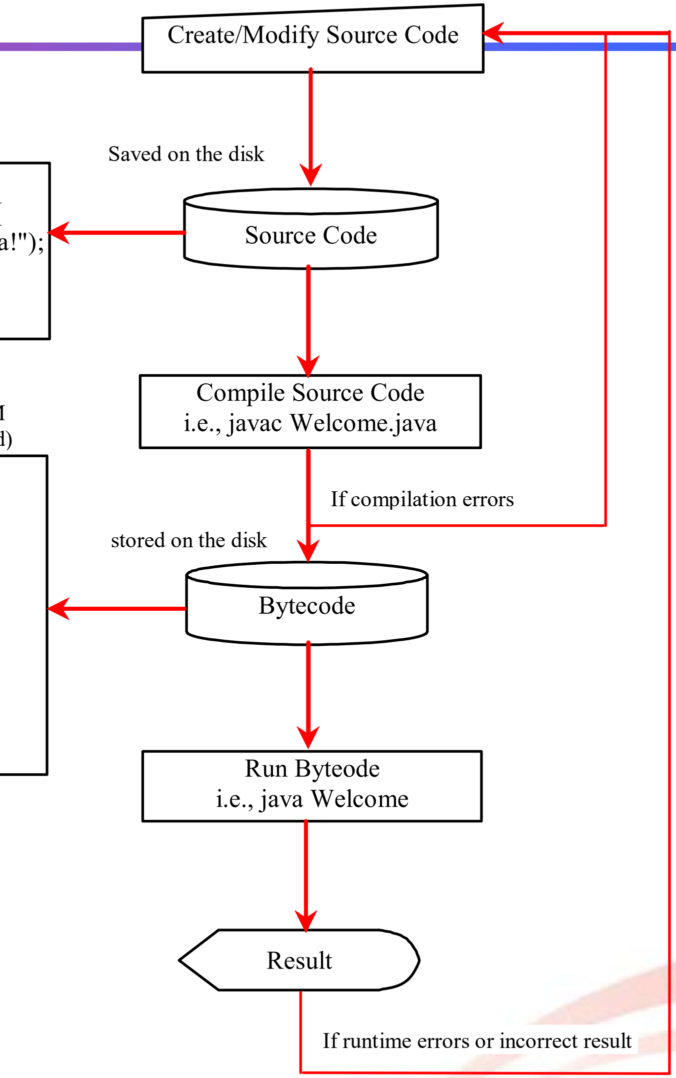
```
//This program prints Welcome to Java!
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```

Source code (developed by the programmer)

```
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```

Byte code (generated by the compiler for JVM to read and interpret, not for you to understand)

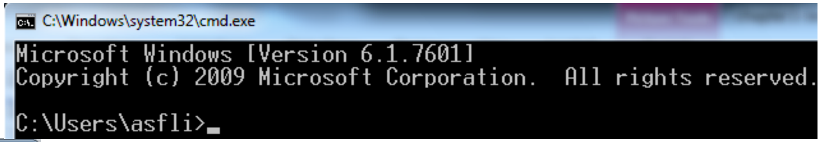
```
…
Method Welcome()
  0 aload_0
  …

Method void main(java.lang.String[])
  0 getstatic #2 …
  3 ldc #3 <String "Welcome to
Java!">
  5 invokevirtual #4 …
  8 return
```



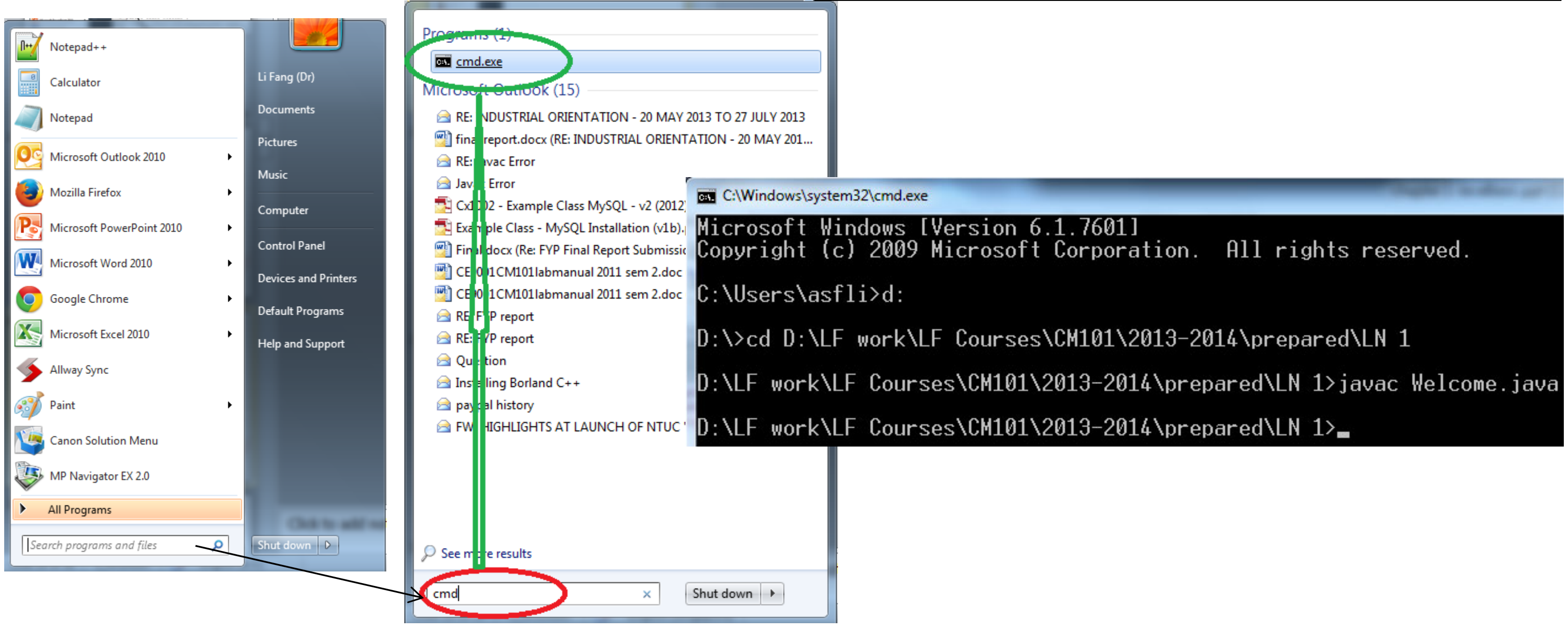
Step 2: To compile your program

Type JDK Command:

javac Welcome.java



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\asfli>_
```



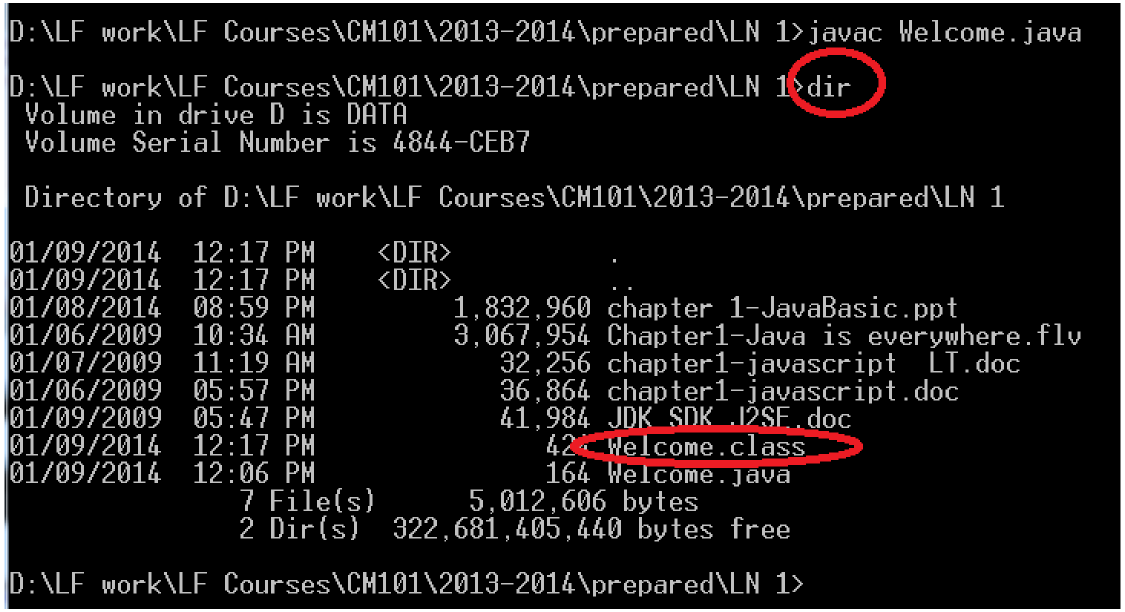
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\asfli>d:

D:\>cd D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>javac Welcome.java

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>_
```

where javac is JDK Java compiler.

- •Compiler translates the source program into Java bytecode.
- •The compiler saves the bytecode into the file **Welcome.class.**

Step 3: To run the byte code with the Java interpreter

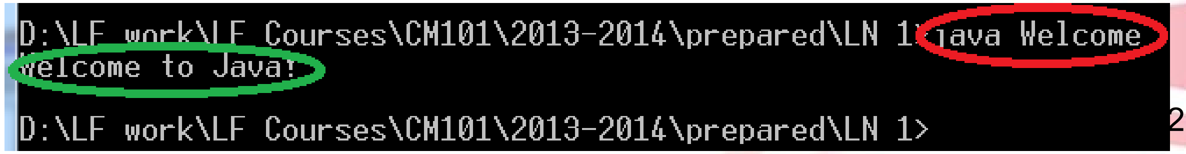
Successful compilation will create the bytecode class file: **Welcome**.class



```
D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>javac Welcome.java

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>dir
 Volume in drive D is DATA
 Volume Serial Number is 4844-CEB7

 Directory of D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1

01/09/2014  12:17 PM    <DIR>          .
01/09/2014  12:17 PM    <DIR>          ..
01/08/2014  08:59 PM         1,832,960 chapter 1-JavaBasic.ppt
01/06/2009  10:34 AM         3,067,954 Chapter1-Java is everywhere.flv
01/07/2009  11:19 AM            32,256 chapter1-javascript  LT.doc
01/06/2009  05:57 PM            36,864 chapter1-javascript.doc
01/09/2009  05:47 PM            41,984 JDK SDK J2SE.doc
01/09/2014  12:17 PM               424 Welcome.class
01/09/2014  12:06 PM               164 Welcome.java
               7 File(s)      5,012,606 bytes
               2 Dir(s)  322,681,405,440 bytes free

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>
```

Type JDK Command : **java Welcome**

- ➢The class file (bytecode) is loaded into memory and interpreted by the Java Virtual Machine (JVM)



```
D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>java Welcome
Welcome to Java!

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>
```

Trace a Program Execution

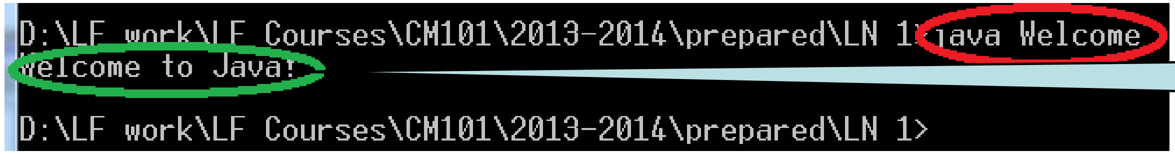
Enter main method

```
//This program prints Welcome to Jav
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```

Trace a Program Execution

Execute statement

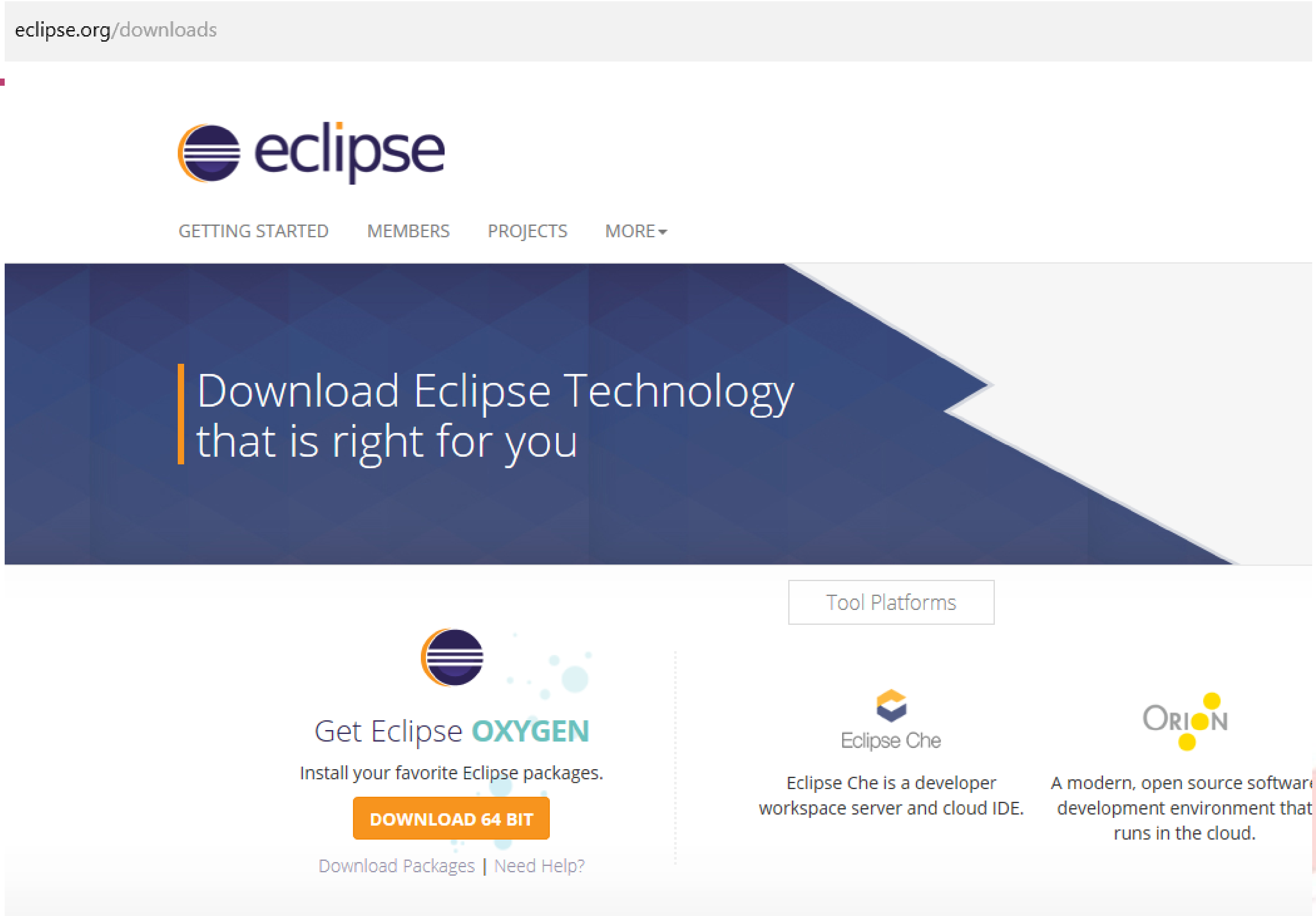
```
//This program prints Welcome to Java!
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to Java!");
  }
}
```



```
D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>java Welcome
Welcome to Java!

D:\LF work\LF Courses\CM101\2013-2014\prepared\LN 1>
```

print a message to the console

Download eclipse



eclipse.org/downloads

Download Eclipse Technology that is right for you

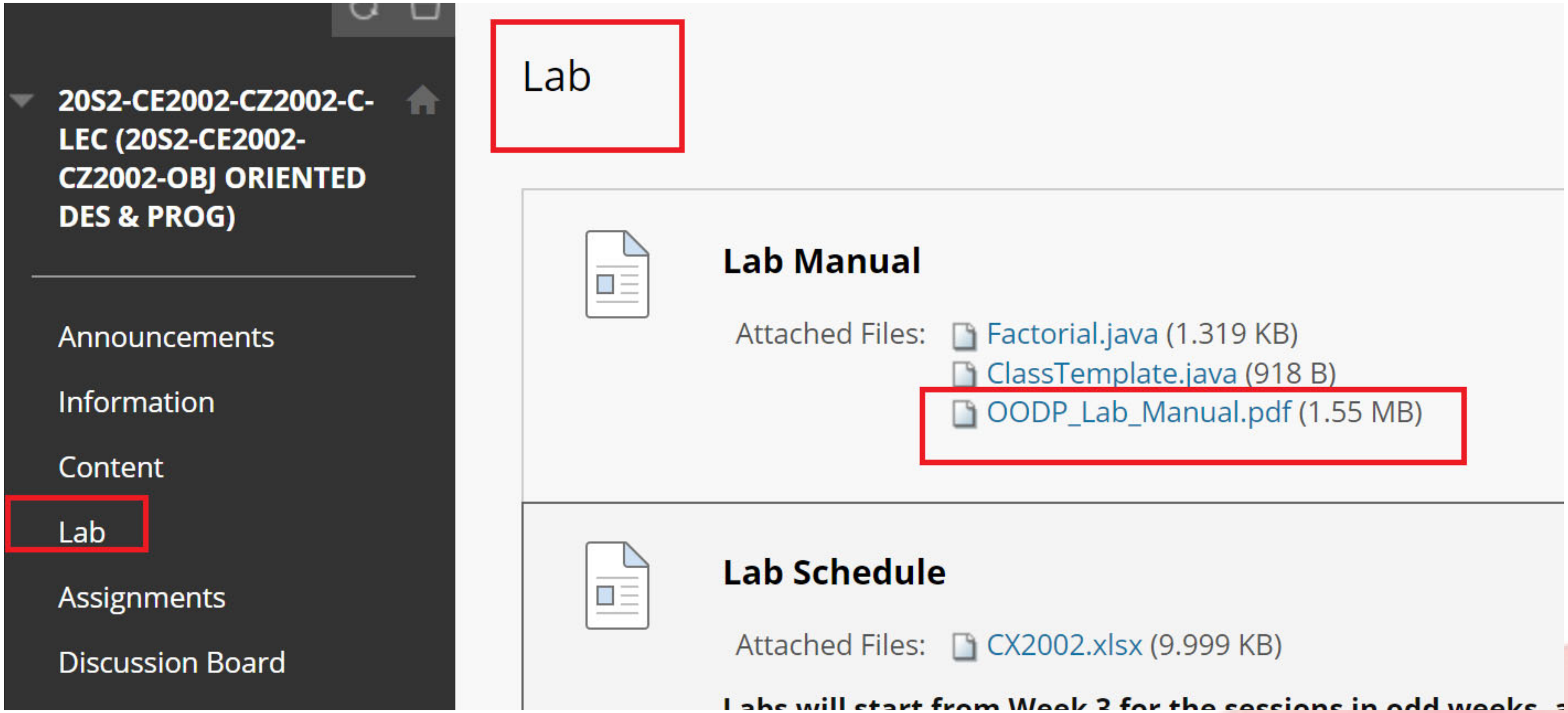
Get Eclipse OXYGEN

Install your favorite Eclipse packages.

Eclipse Che is a developer workspace server and cloud IDE.

A modern, open source software development environment that runs in the cloud.

Eclipse IDE quick start： Lab manual



20S2-CE2002-CZ2002-C-LEC (20S2-CE2002-CZ2002-OBJ ORIENTED DES & PROG)
Announcements
Information
Content
Lab
Assignments
Discussion Board

Lab

Lab Manual

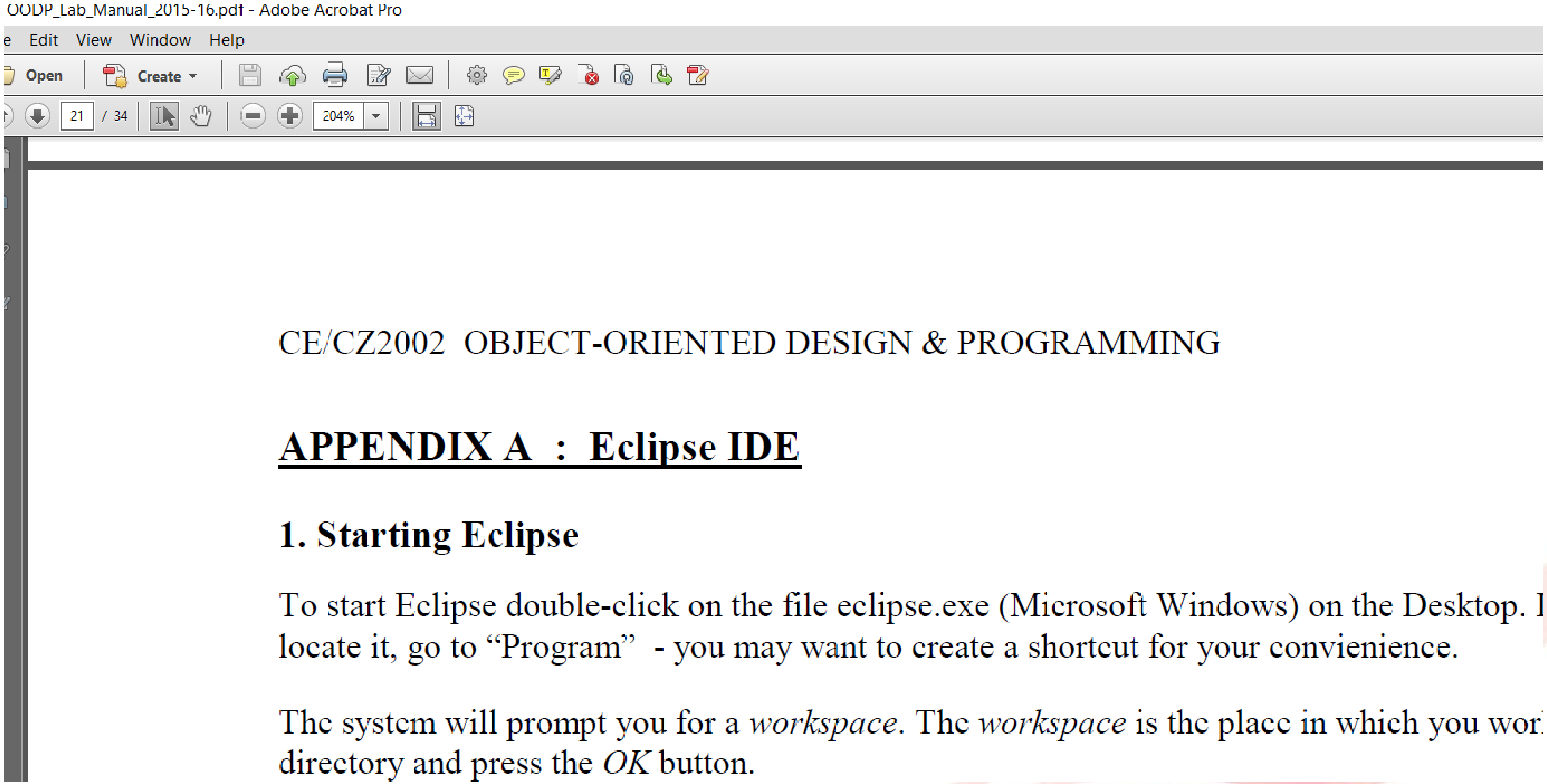
Attached Files: Factorial.java (1.319 KB)
ClassTemplate.java (918 B)
OODP_Lab_Manual.pdf (1.55 MB)

Lab Schedule

Attached Files: CX2002.xlsx (9.999 KB)

Labs will start from Week 3 for the sessions in odd weeks

Eclipse IDE quick start: Lab manual



CE/CZ2002 OBJECT-ORIENTED DESIGN & PROGRAMMING

APPENDIX A : Eclipse IDE

1. Starting Eclipse

To start Eclipse double-click on the file eclipse.exe (Microsoft Windows) on the Desktop. I locate it, go to “Program” - you may want to create a shortcut for your convienience.

The system will prompt you for a *workspace*. The *workspace* is the place in which you wor directory and press the *OK* button.

NTU COPYRIGHT STATEMENT

All course materials, including but not limited to, lecture slides, handout and recordings, are for your own educational purposes only. **All the contents of the materials are protected by copyright, trademark or other forms of proprietary rights.**

All rights, title and interest in the materials are owned by, licensed to or controlled by the University, unless otherwise expressly stated. **The materials shall not be uploaded, reproduced, distributed, republished or transmitted in any form or by any means, in whole or in part, without written approval from the University.**

You are also not allowed to take any photograph, film, audio record or other means of capturing images or voice of any contents during lecture(s) and/or tutorial(s) and reproduce, distribute and/or transmit any form or by any means, in whole or in part, without the written permission from the University.

Appropriate action(s) will be taken against you including but not limited to disciplinary proceeding and/or legal action if you are found to have committed any of the above or infringed the University's copyright.