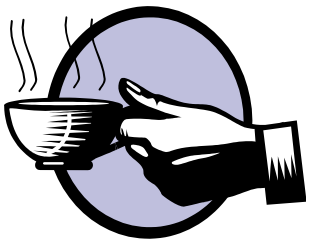
COP2800 – Java Programming



INSTRUCTOR: **Prof. Greg Ballinger**
OFFICE: **6151**
TELEPHONE: **(305) 237-2879**
EMAIL: **gballing@mdc.edu**
WEBSITE: **http://faculty.mdc.edu/gballing**

INSTRUCTOR OFFICE HOURS: **Posted at http://faculty.mymdc.net/faculty_member_selector.asp (follow the link at http://www.mdc.edu/current/)**

COMPUTER ACCESS:

You will need to use a computer with internet access to complete assignments. Study Center in Room 9103 provides computers access for CIS students. You will be asked to provide a current MDC ID and validated schedule when you enter the Study Center. The hours of operation are on the door of Room 9103 and on the CIS Department web page located at http://www.cis.kendall.mdc.edu/About/Center.htm. Broken computers and "internet down" are not valid excuses! You can use the CIS Study Center computers.

COURSE DESCRIPTION

This course is an intermediate programming course using the Java computer language. Students are required to code, compile and execute programs. Object oriented programming techniques as they are applied in event driven programming will be presented. Practical examples of object oriented programming for the World Wide Web will be studied. Prerequisites: COP 1220. Laboratory fee (3hr. lecture; 2hr lab). Detailed course competencies are online at http://faculty.mdc.edu/gballing/Java/COP2800_Comp.pdf .

TEXTBOOKS & MATERIALS

Required:

1) *Big Java 3e with WileyPLUS* by Cay Horstmann. Publisher: John Wiley & Sons, Inc.
   (WileyPLUS provides access to *required* online features)

Purchase options:

a. Purchase printed book at the MDC bookstore (includes WileyPLUS) ISBN: **9780470132111**
or
b. Purchase WileyPLUS only at the MDC bookstore (includes eText) ISBN: **9780470111130**
or
c. Purchase WileyPLUS separately ONLINE (includes eText)
Note: Online purchase must be made at an URL specific for your class – to be provided

2) USB Drive (Flash Drive, Thumb drive) or several 3½" Floppy Disks.

Highly Recommended:

- Home computer with high speed internet access

E-MAIL AND PHONE MESSAGES

Be sure to include your name (not just your e-mail address!) and your class in every email and phone message. I try to respond within 2 business days to all e-mail. Remember that e-mail is NOT guaranteed delivery. If you do not receive a response, it may be necessary to leave a phone message. Use the email at the top of this syllabus, *do NOT use the ANGEL email.*

ATTENDANCE

This is a participatory course and you must be in class to succeed. Roll will be taken at the *start* of each class – if you are late you will be counted absent. It is the student's responsibility to obtain the information missed during an absence.

TEXTBOOK DATAFILES AND INSTRUCTOR FILES

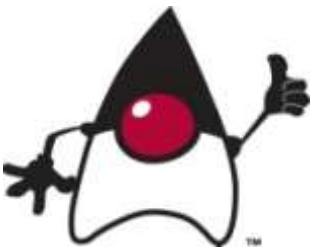
Some files may be stored on the CIS network in the classroom and in the CIS Study Center. Instructor files (data, sample code, handouts) may be found on the INSTRUCTOR share in the BALLINGER directory. *If you need a file from campus, be sure to make a copy of it before leaving campus.*

READING and TUTORIAL ASSIGNMENTS

Students will be responsible for all material covered in class and all chapters assigned in the textbook. Readings should be completed before class begins. While some class time will be spent working on the tutorial exercises in the book, ***you are responsible for completing each of the assigned tutorials***.

GRADING:

Grading Criteria	number	percent of total
Major Assignments	5 – 10	30%
Minor Assignments & Quizzes	10 -- 20	20%
Exams	2 – 3	20%
Final Project	1	15%
Final exam	1	15%
Total		100%



Final Grading Scale	A	B	C	D
Percentage cut-off	90%	80%	70%	60%

- Students must make up a missed exam within one week or zero (0) points will be assigned as the exam grade. Contact instructor during office hours to make the arrangements. Make arrangements in advance as there will be limited opportunities following the test.

GUIDELINES FOR PROGRAMMING ASSIGNMENTS

Assignments are due at the beginning of class on the due date. *Late assignments will lose 10% of their grade and will only be accepted up to one week past their due date*. Each assignment will have specific submission guidelines. Most assignments will be submitted via the internet to either the ANGEL or WileyPLUS web sites.

All coding assignments must follow professional coding style guidelines. See the Coding Style Guide.

While I encourage discussion about assignments, *ALL CODE MUST BE YOUR OWN* unless otherwise specified. Since it is impossible to determine who the author was and who was the plagiarizer, all assignments submitted with duplicate code will receive a failing grade.

USEFUL WEBSITES

Sun Java Technology: http://java.sun.com/

ANGEL: http://mdc.angellearning.com/

WileyPLUS: http://www.wileyplus.com/

BlueJ IDE website: http://www.bluej.org/

The Java Tutorial: http://java.sun.com/docs/books/tutorial/index.html

USEFUL BOOKS

- Kathy Sierra and Bert Bates, **Head First Java (2nd Edition)** , O'Reilly. A highly regarded alternative textbook. Good alternative explanations and illustrations; a limited preview is available at books.google.com.
- Peter van der Linden. **Just Java™ 2 (6th Edition)**. Prentice Hall. Good reference and source of alternative explanations.
- Ken Arnold, James Gosling, and David Holmes. **The Java™ Programming Language**. Addison-Wesley . Good description of language design; interesting reading but not a good reference for syntax.
- Steve McConnell, **Code Complete (2nd Edition)**, Microsoft Press. This is a book that every professional programmer should read. McConnell's claim "*... this handbook will help you to create higher-quality software and to do your work more quickly and with fewer problems.*" is absolutely true.

WITHDRAWAL AND INCOMPLETE

All students that are listed on the final grade report will receive a final grade. Incomplete grades will be given only if a) you are up-to-date in class AND b) you have a passing grade AND c) you have an emergency or life change that occurred after the semester began that can be verified and is beyond your control. It is your responsibility to determine the official drop dates for your courses. See http://www.mdc.edu/academic_calendar/.

"Patience and perseverance have a magical effect before which difficulties disappear and obstacles vanish."
John Quincy Adams

Projected Schedule

Module	Title/Activities	Module Begins	Assignments Due	Chapters
1	**Tools for Programming** Syllabus Quiz Java Technology Quiz Role of Variables Quiz Assignment 1 - Checking your JDK Installation Assignment 2 - ISBN Part 1	1/6	1/13	1 (not needed for assign ments!)
2	**Using objects and exploring variables** WileyPLUS Assignment Zero Module 2 Quiz Using API Documentation quiz Programming Exercise 1 – Perimeter *(req)* Assignment 3 - ISBN Part 2 Begins	1/13	1/20	2
3	**Exploring objects and writing methods** Programming Exercise 2 - Die Simulator *(opt)* Programming Assignment 1 - Graphics Applet Assignment 3 - ISBN Part 2 Due	1/20	1/27	2
4	**Defining and modifying objects** Programming Exercise 3 - Product class *(opt)* Programming Exercise 4 - Bug class *(opt)* Programming Assignment 2 - Olympic Rings ***Exam_1***	1/27	2/10	3
5	**Data types & control Structures** Programming Exercise 5 – Counter (req) Programming Exercise 6 – DataSet (req) Programming Assignment 4 – Debugging Assignment 4 - ISBN Part 3 Assignment 4 - Individual Process Report	2/10	2/17	4, 5 & 6
6	**Object Interaction, Unit Testing and Delivery** Programming Assignment 5 – Clocks Module 6 Quiz – Testing	2/17	2/24	Pages 47-50, 102- 104, 212-215, 319-321, and 372-374
7	**Managing multiple objects and values** Array Quiz Programming Exercise 7 - Array Output *(opt)* Programming Exercise 8 - Passing Arrays *(opt)* ArrayList Quiz Programming Exercise 9 - Purse *(req)* Programming Exercise 10 - Purse Transfer *(req)* Programming Assignment 6 - Game of Life	2/24	3/10	7
8	**Designing High Quality Classes** Programming Assignment 7 - Library Manager Professional Development Assignment ***Exam 2***	3/10	3/17	8

9	**Code Inheritance and Polymorphism** Programming Exercise 11 - Square *(opt)* Programming Assignment 8 - Media Collection	**3/17**	**3/24**	**10**
10	**Design Inheritance with Java interfaces** Programming Exercise 12 - WordMeasurer *(req)*	**3/24**	**3/31**	**9**
11	**Event handling and Building GUIs** Programming Exercise 13 - Two Buttons *(opt)* Programming Assignment 9 - Animation	**3/31**	**4/14**	**Chapter 9, pages 408 - 429, Chapter 10, pages 478-489, Chapter 18 - all**
12	**Exception Handling and File I/O** Programming Exercise 14 - Catch Exception *(opt)* Programming Assignment 10 - Link Extraction	**4/14**	**4/21**	**Chapter 11 and Chapter 19 section 19.1: pages 820-822, and section 19.4: pages 833 - 838.**
13	**Recursion and A Closing Look at Good Design** Programming Assignment 11- List Files Completed Final Project Uploaded ***Final Exam 4/30***	**4/21**	**4/28**	**12 & 13**
	Final Project	**3/24**	**4/28**	

"Putting off an easy thing makes it hard.
Putting off a hard thing makes it impossible."
-- George Claude Lorimer

Steps to success!

1. **READ THE TEXTBOOK**! *Before* all else fails, read the directions.

2. **PUT IN THE TIME**! You should plan to spend a lot of time on this class. In college the average amount of study time per week should be twice the amount of lecture time in each class. Since you will spend 5 hours per week in this class, you should plan on spending ***at least* 10 hours per week studying for this class**. I say at least because the 2*Lecture hours formula is an average. Since this is a programming class, plan on spending more time than the average.

3. **ASK QUESTIONS**! *Come prepared with questions*. Read the material before coming to class and develop a list of questions or topics for which you would like an alternative explanation. If the instructor doesn't answer these questions or offer the explanation during lecture, ask the question or request the explanation. Ask the instructor to clarify any points you have at the point of your confusion, don't expect to figure it out later.

4. **TAKE NOTES IN CLASS**. What did he say? That it was going to on the test or not? Writing things down makes them easier to remember.

5. **TAKE NOTES WHILE READING**! If writing notes in class helps you remember what the instructor said, taking notes while reading will help you remember what the author wrote. Highlighting does not improve memory.

6. **MAKE A LANGUAGE SUMMARY SHEET**. Every time you encounter a new Java concept, add it to a master sheet showing the correct syntax and relevant notes for each Java concept you learn. Having this handy will greatly speed programming.

7. **PLAN BEFORE CODING**. You have been speaking English for much longer than you have been speaking Java. Write out your program in English (pseudocode) first. .If you can't clearly express the program in English, you are not ready to try translating it into Java!

8. **THINK THROUGH NON-ASSIGNED EXERCISES**. Test your understanding of the material by thinking through some of the end-of-chapter exercises in the text, perhaps even going as far as writing pseudocode.

9. **REVIEW NEW MATERIAL WITHIN 24 HOURS**. A quick reading of your notes the day after you take them will almost double your memory of the material. If you read the notes out loud, it will double your memory.

10. **START ASSIGNMENTS IMMEDIATELY**! All assignments in this course will take longer than you expect. If you start early, you will be able to use the instructor's (and fellow students') class and lab time to help you develop your project. Waiting until the week it is due will not leave enough time to ask for help!

11. **HAVE FUN**! If you become bored, frustrated or sleepy this usually means that your short-term memory is "full" and it is time to take a break. Even a short break will give your short-term memory a chance to process what it has just absorbed.

The purpose of computing is insight, not numbers.

-- R. Hamming.