

## Annotated Learning Outcomes for COP 2800, Java Programming, Reed, Fall 2010.

### General References – Start Here

- Java Technology Reference (This used to be Sun's and is now Oracle's starting page for downloads, docs, etc. related to Java)

<http://java.sun.com/reference/index.jsp#documentation>

- Wikipedia page on Java (while Wikipedia can be problematic on controversial issues, the technical material on programming there is first-rate).

[http://en.wikipedia.org/wiki/Java\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/Java_%28programming_language%29)

- Complete index to the Sun/Oracle Java tutorial. This is one of the oldest and best Java tutorials, from the folks who invented (or at least popularized) the language.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/reallybigindex.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/reallybigindex.html)

- Eclipse is the dominant Integrated Development Environment (IDE) for Java, and to a lesser extent, C and C++. Like any big piece of machinery, it is powerful, complex and challenging to master.

<http://eclipse.org/>

- Processing is a user-friendly, graphics-centric environment wrapped around Java. A great tool for teaching beginning programming in Java, Processing lets you write short programs that actually do something stunning, right off the bat.

<http://processing.org/>

- *InfoWorld* Magazine's Java News page. Lots of late-breaking information on Java and its components.

<http://www.javaworld.com/>

- *The Java Language Environment*. Written in 1996 by James Gosling, one of the inventors of Java, this page is still one of the best overviews for the rationale of Java and how and why it differs from other languages.

<http://java.sun.com/docs/white/langenv/index.html>

- *Java Enterprise Edition (J2EE) Tutorials*. A good starting page for lots of information about J2EE, from Sun/Oracle.

<http://java.sun.com/javaee/reference/tutorials/>

- Despite the “academic” name, this looks like a very cool Java tutorial site. It's organized by

keyword, and features lots of videos and links to Facebook, if you like that sort of thing. (I'm thinking of sampling this guy's picture for my instructor mugshot in WebCt/BB. In an on-line class, who'd know? And is that a Java tat' he's sporting?).

<http://academicjava.com/>

- Sun/Oracle site with “Feature Stories About Java Technology”. Looks like a good place to get or keep abreast of recent Java trends.

<http://java.sun.com/features/index.html>

- First in a series of videos of the lectures for an Introduction to Programming course at Stanford. For the price of a Lexus, you could be sitting in this lecture “live” for a year.

<http://www.youtube.com/watch?v=KkMDCCdjyW8&feature=related>

- Ok, I know we're not kids (anymore) but there are some great ideas in this Google video about Greenfoot, a Java-based environment for teaching children to program. And the first five minutes or so, which show how software demos are like magic tricks, is great.

<http://www.youtube.com/watch?v=Tewx-I6Arwk>

- An open book about Java for beginners, courtesy of WikiBooks.

[http://en.wikibooks.org/wiki/Java\\_Programming](http://en.wikibooks.org/wiki/Java_Programming)

- Complete on-line version of the O'Reilly book *Java in a Nutshell*.

<http://docstore.mik.ua/oreilly/java-ent/jnut/index.htm>

## **L1 – History and significance of the Java language within the context of the software industry.**

- You will be able to Describe the history of Java, and the reasoning that makes Java an important programming language in the context of other programming languages.
- Gaddis, Chapter 1, esp. pp 8-9

<http://www.java.com/en/javahistory/>

[http://en.wikipedia.org/wiki/Java\\_%28programming\\_language%29#History](http://en.wikipedia.org/wiki/Java_%28programming_language%29#History)

[http://ei.cs.vt.edu/book/chap1/java\\_hist.html](http://ei.cs.vt.edu/book/chap1/java_hist.html)

- You will be able to Recognize and be able to describe the difference between Java applications and Java applets.

Gaddis, Chapter 1, esp. pp 8-9

<http://www.javacoffeebreak.com/faq/faq0009.html>

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/deployment/applet/](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/deployment/applet/)  
<http://www.justobjects.org/cowcatcher/browse/all/slides/java-applet/core/applet/slide.0.2.html>  
[http://en.wikipedia.org/wiki/Java\\_applet](http://en.wikipedia.org/wiki/Java_applet)  
<http://java.sun.com/developer/technicalArticles/Programming/TurningAnApplet/>

- You will be able to Explain the significance of Java as it applies to programming for the internet.

<http://www.java-samples.com/showtutorial.php?tutorialid=26>  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/getStarted/intro/changemylife.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/getStarted/intro/changemylife.html)

## **L2 - Primitive data types, arithmetic and relational operators.**

- You will be able to Define variables using any of the eight primitive data types.

Gaddis, Chapter 2, esp. pp. 44-52

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/datatypes.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/datatypes.html)  
[http://java.sun.com/docs/books/jls/second\\_edition/html/typesValues.doc.html](http://java.sun.com/docs/books/jls/second_edition/html/typesValues.doc.html)

- You will be able to Use variables and arithmetic operators to write statements containing arithmetic expressions.

Gaddis, Chapter 2, esp. pp 52-64

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/op1.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/op1.html)  
<http://www.javabeginner.com/learn-java/java-operators-tutorial>

- You will be able to Use variables and relational operators to write statements containing relational expressions.

Gaddis, Chapter 3, esp. pp. 111-116

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/op2.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/op2.html)  
<http://www.javabeginner.com/learn-java/java-operators-tutorial>  
[http://www.freejavaguide.com/relational\\_operators.htm](http://www.freejavaguide.com/relational_operators.htm)

- You will be able to Write statements that combine data types in a single expression, and promote and demote (cast) values from one primitive data type to another.

Gaddis, Chapter 2, esp. pp. 64-68, 95-98, Chapter 10, esp. pp 624-625

[http://en.wikibooks.org/wiki/Java\\_Programming/Types#Data\\_Conversion\\_.28Casting.29](http://en.wikibooks.org/wiki/Java_Programming/Types#Data_Conversion_.28Casting.29)  
[http://java.sun.com/docs/books/jls/third\\_edition/html/conversions.html](http://java.sun.com/docs/books/jls/third_edition/html/conversions.html)

## **L3 - Selection and repetition constructs in Java.**

- You will be able to Design and implement selections constructs using 'if' and 'if-else' statements.

Gaddis, Chapter 3.

- [http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/if.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/if.html)  
[http://java.sun.com/docs/books/jls/third\\_edition/html/statements.html](http://java.sun.com/docs/books/jls/third_edition/html/statements.html)  
<http://www.youtube.com/watch?v=FEkceTEI5DA>  
[http://en.wikipedia.org/wiki/Conditional\\_%28programming%29](http://en.wikipedia.org/wiki/Conditional_%28programming%29)  
[http://javaboutique.internet.com/tutorials/Java\\_by\\_Example/section1\\_4.html](http://javaboutique.internet.com/tutorials/Java_by_Example/section1_4.html)
- You will be able to Design and implement selection constructs using the 'switch' construct.

Gaddis, Chapter 3, esp. pp. 152-159

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/switch.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/switch.html)  
<http://www.youtube.com/watch?v=4leW1nU6goY>  
[http://javaboutique.internet.com/tutorials/Java\\_by\\_Example/section1\\_6.html](http://javaboutique.internet.com/tutorials/Java_by_Example/section1_6.html)

- You will be able to Design and implement repetition using 'while', 'do - while', and 'for' loops.

Gaddis, Chapter 4.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/while.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/while.html)  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/for.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/for.html)  
<http://www.youtube.com/watch?v=rjkYAs6gAkk>  
<http://www.youtube.com/watch?v=8ZuWD2CBjgs>

- You will be able to Design and implement nested selection and repetition.

Gaddis, Chapter 4, esp. pp. 211-212

[http://www.java2s.com/Tutorial/Java/0080\\_\\_Statement-Control/NestedforLoop.htm](http://www.java2s.com/Tutorial/Java/0080__Statement-Control/NestedforLoop.htm)  
[http://www.javamex.com/tutorials/java/for\\_loop\\_nested.shtml](http://www.javamex.com/tutorials/java/for_loop_nested.shtml)

## **L4 – Collections (e.g. List, Queue) and Arrays**

Gaddis, Chapter 8 (Arrays).

- You will be able to Declare and initialize Collections and Arrays  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/nutsandbolts/arrays.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/nutsandbolts/arrays.html)  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/collections/interfaces/](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/collections/interfaces/)  
<http://forum.codecall.net/java-tutorials/7776-tutorial-java-collections.html>  
<http://www.learn-java-tutorial.com/Java-Arrays.cfm>  
<http://www.youtube.com/watch?v=JdsYh4pNTtI>  
[http://www.youtube.com/watch?v=0MIyqDtDs\\_M](http://www.youtube.com/watch?v=0MIyqDtDs_M)
- You will be able to explain the difference between a collections interface, its implementation, and the algorithms it provides.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/collections/interfaces/](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/collections/interfaces/)

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/IandI/createinterface.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/IandI/createinterface.html)  
<http://www.codeguru.com/java/tij/tij0059.shtml>  
[http://en.wikipedia.org/wiki/Interface\\_%28Java%29](http://en.wikipedia.org/wiki/Interface_%28Java%29)  
<http://mindprod.com/jgloss/interfacevsabstract.html>  
<http://java.sun.com/developer/onlineTraining/collections/Collection.html>  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/collections/](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/collections/)

- You will be able to name and briefly explain the core collection interfaces present in the Java API.

<http://java.sun.com/developer/Books/javaprogramming/corejava/>  
<http://javaboutique.internet.com/tutorials/testing/>  
<http://www.oracle.com/technology/pub/articles/maps1.html>  
<http://www.youtube.com/watch?v=xD9J3V5O1kk>  
[http://www.youtube.com/watch?v=ZeO\\_J2OcHYM](http://www.youtube.com/watch?v=ZeO_J2OcHYM)

- You will be able to Design and code Java applications that use Collections.

[http://www.idevelopment.info/data/Programming/java/PROGRAMMING\\_Java\\_Programming.shtml](http://www.idevelopment.info/data/Programming/java/PROGRAMMING_Java_Programming.shtml)  
<http://www.cs.aau.dk/~torp/Teaching/E01/Oop/handouts/collections.pdf>  
<http://www.java-examples.com/java-collections-and-data-structures-%28-java.util.package-%29>  
<http://www.youtube.com/watch?v=jU5ACV5MucM>

## **L5 - Object-oriented principles using Java.**

Gaddis, Chapter 6, and Chapter 11

- You will be able to Describe the concepts of class, object, method, interface, inheritance and polymorphism.

[http://en.wikipedia.org/wiki/Object-oriented\\_programming](http://en.wikipedia.org/wiki/Object-oriented_programming)  
<http://java.sun.com/developer/onlineTraining/Programming/BasicJava2/oo.html>  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/IandI/subclasses.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/IandI/subclasses.html)  
<http://archive.eiffel.com/doc/manuals/technology/oosc/finding/page.html>  
[http://en.wikipedia.org/wiki/List\\_of\\_object-oriented\\_programming\\_terms](http://en.wikipedia.org/wiki/List_of_object-oriented_programming_terms)  
<http://www.youtube.com/watch?v=iYtri45lhtc>

- You will be able to Invoke methods from existing classes including 'String' and 'StringBuffer.'

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/data/strings.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/data/strings.html)  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/i18n/text/stream.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/i18n/text/stream.html)  
<http://www.javaworld.com/javaworld/jw-03-2000/jw-0324-javaperf.html>  
<http://java.sun.com/developer/technicalArticles/releases/1.4regex/>  
[http://en.wikipedia.org/wiki/Java\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/Java_%28programming_language%29)

- You will be able to Create objects from existing classes.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/objectcreation.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/objectcreation.html)

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/objects.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/objects.html)  
[http://en.wikibooks.org/wiki/Java\\_Programming/Tutorials/Notepad](http://en.wikibooks.org/wiki/Java_Programming/Tutorials/Notepad)  
<http://www.javacoffeebreak.com/java102/java102.html>

- You will be able to Design and implement classes that use static variables and methods, and instance variables and methods.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/classvars.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/classvars.html)  
<http://www.javaworld.com/javaworld/javaqa/2001-11/03-qa-1121-mrhappy.html?page=1>  
<http://today.java.net/article/2003/12/13/not-so-stupid-questions-1-static-methods>  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/methods.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/methods.html)

- You will be able to Design and implement class hierarchies that exercise method overloading, inheritance, and polymorphism.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/methods.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/methods.html)  
<http://www.java-samples.com/showtutorial.php?tutorialid=284>  
[http://java.sun.com/docs/books/jls/second\\_edition/html/classes.doc.html#228745](http://java.sun.com/docs/books/jls/second_edition/html/classes.doc.html#228745)  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/concepts/inheritance.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/concepts/inheritance.html)

- You will be able to Write code that demonstrates the concepts of abstract classes, and abstract methods.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/AndI/abstract.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/AndI/abstract.html)  
<http://www.java-samples.com/showtutorial.php?tutorialid=288>  
<http://www.codeguru.com/java/tij/tij0079.shtml>  
[http://docstore.mik.ua/oreilly/java-ent/jnut/ch03\\_06.htm](http://docstore.mik.ua/oreilly/java-ent/jnut/ch03_06.htm)

## **L6 – Organization of packages, and the most significant packages that are part of the Java API.**

Gaddis, Chapter 6, esp. pp 343-345, Chapter 11, esp. pp. 694-695, Appendix 11 (CD)

- You will be able to Recognize the common packages in the Java API and be able to navigate through the documentation that describes the packages.

[http://www.jarticles.com/package/package\\_eng.html](http://www.jarticles.com/package/package_eng.html)  
[http://en.wikipedia.org/wiki/Java\\_package](http://en.wikipedia.org/wiki/Java_package)  
[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/package/packages.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/package/packages.html)  
[http://java.sun.com/docs/books/jls/third\\_edition/html/packages.html](http://java.sun.com/docs/books/jls/third_edition/html/packages.html)  
<http://www.apl.jhu.edu/~hall/java/Documentation.html>

- You will be able to Implement a package of classes.

[http://tinf2.vub.ac.be/~dvermeir/java/other\\_doc/JavaPackages.html](http://tinf2.vub.ac.be/~dvermeir/java/other_doc/JavaPackages.html)  
<http://pages.cs.wisc.edu/~hasti/cs368/JavaTutorial/NOTES/Packages.html>  
<http://www.particle.kth.se/~lindsey/JavaCourse/Book/Part1/Java/Chapter05/packagesImport.html>  
<http://java.sys-con.com/node/49108>  
[http://www.youtube.com/watch?v=IqDIYt\\_j5nM](http://www.youtube.com/watch?v=IqDIYt_j5nM)

- You will be able to Implement classes that use the access modifiers public, private, protected and the default modifier.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/java/javaOO/accesscontrol.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/java/javaOO/accesscontrol.html)  
<http://www.javabeginner.com/learn-java/introduction-to-java-access-modifiers>  
<http://www.javacoffeebreak.com/faq/faq0002.html>

## **L7 – Exception handling methods - try-catch-finally construct and its use according to accepted conventions.**

Gaddis, Chapter 12.

- You will be able to create classes that inherit from the Exception class.

[http://download.oracle.com/docs/cd/E17409\\_01/javase/tutorial/essential/exceptions/handling.html](http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/essential/exceptions/handling.html)  
<http://today.java.net/article/2006/04/04/exception-handling-antipatterns>  
<http://onjava.com/pub/a/onjava/2003/11/19/exceptions.html>

- You will be able to Be able to throw and catch exceptions in a method.

[http://www.hostitwise.com/java/java\\_exceptions.html](http://www.hostitwise.com/java/java_exceptions.html)  
[http://www.youtube.com/watch?v=K\\_-3OLkXkzY](http://www.youtube.com/watch?v=K_-3OLkXkzY)

- You will be able to Design and implement methods that throw exceptions that will be caught and handled in other methods.

<http://www.artima.com/designtechniques/exceptions.html>  
<http://www.developer.com/java/other/article.php/1455891/The-Essence-of-OOP-using-Java-Exception-Handling.htm>  
<http://www.javabeginner.com/learn-java/understanding-java-exceptions/all/1>

- You will be able to Write code that appropriately uses the assert mechanism to throw AssertionError objects.

[http://download.oracle.com/docs/cd/E17476\\_01/javase/1.4.2/docs/guide/lang/assert.html](http://download.oracle.com/docs/cd/E17476_01/javase/1.4.2/docs/guide/lang/assert.html)  
<http://java.sun.com/developer/technicalArticles/JavaLP/assertions/>  
<http://www.particle.kth.se/~lindsey/JavaCourse/Book/Part1/Supplements/Chapter05/debugging.html>  
[http://www.deitel.com/articles/java\\_tutorials/20060106/Assertions.html](http://www.deitel.com/articles/java_tutorials/20060106/Assertions.html)

## **L8 – Advanced Topics**

- Compilation, interpretation, byte code and the JVM

<http://www.javaworld.com/javaworld/jw-07-2010/100713-jvm-languages.html>  
[http://www.ibm.com/developerworks/ibm/library/it-haggar\\_bytecode/](http://www.ibm.com/developerworks/ibm/library/it-haggar_bytecode/)

- Java Frameworks and derivative languages

<http://groovy.codehaus.org/>

<http://processing.org>

<http://www.javaworld.com/javaworld/jw-07-2010/100713-jvm-languages.html>

- Embedded Java

<http://java.sun.com/javame/index.jsp>

<http://www.vogella.de/articles/Android/article.html>

- Native code

<http://java.sun.com/docs/books/jni/html/objtypes.html>

- Multimedia applications

[http://www.deitel.com/articles/java\\_tutorials/20060422/PlayingVideowithJMF/](http://www.deitel.com/articles/java_tutorials/20060422/PlayingVideowithJMF/)