

SPRINGFIELD TECHNICAL COMMUNITY COLLEGE

ACADEMIC AFFAIRS

Course Number: CSCI 211 Department: Eng. And Science Transfer

Course Title: Intermediate Topics in Java Semester: Spring Year: 2003

Objectives/Competencies

Course Objective	Competencies
1. Learn the syntax of the Java programming language. 2. Learn object-oriented programming with the Java programming language.	1. List the key features of the Java programming language. 2. Describe the Java virtual machine. 3. Explain how garbage collection works. 4. Describe how security features work. 5. Write a simple Java application, compile and run it. 1. Describe the terms used in object-oriented programming: a. "class" b. "object" c. "attribute" d. "method" e. "constructor" 2. Write code to define a method. 3. Access the member variables of an object using the dot notation. 4. Write code to create and initialize an object. 5. Use keyword to access the "current object." 6. Use private and public access modifiers.

Course Objective	Competencies
	<ol style="list-style-type: none">7. Write code to invoke a method on a particular object.8. Write class constructors and invoke particular constructors using new with arguments.9. Understand the use of the package and import statements for library access.10. Use the Java Application Programming interface (API) online documentation.11. Use comments in a program.12. Distinguish between valid and invalid identifiers.13. Recognize keywords in the Java programming language.14. List the eight primitive types.15. Define literal values for numeric and textual types.16. Describe the coding conventions for classes, interfaces, methods, variables, constants, and control structures.17. Create a class definition for a simple class containing primitive member variables.18. Declare variables of a class type.19. Describe the significance of a reference variable and state the consequences of assignment between variables of class type.20. Distinguish between member and automatic variables.21. Recognize and correct a Possible reference before assignment compiler errors.22. Recognize, describe, and use operators.23. Distinguish between legal and illegal assignments of primitive types.24. Recognize Boolean expressions and state the requirements

Course Objective	Competencies
	<p>for these in control constructs.</p> <ol style="list-style-type: none">25. Recognize assignment compatibility and required casts in fundamental types.26. Make appropriate use of it, switch, for, while, and do constructs and the labeled forms of break and continue.27. Declare and create arrays of primitive, class, or array types.28. Explain why and show how to initialize the elements of an array.29. Determine the number of elements in an array.30. Write code to copy arrays.31. Describe encapsulation, polymorphism, and inheritance.32. Use subclassing.33. Create and use heterogeneous collections.34. Create and use methods that accept generic argument types.35. Use access control levels.36. Invoke overloaded methods and constructors.37. Write overriding methods in a subclass and describe execution flow when executing and overridden method.38. Invoke overridden constructors.39. Control invocation of parent class constructors.40. Use wrapper classes.41. Declare and use static variables and methods.42. Declare and use final classes, methods and variables.43. Use abstract methods and interfaces.44. Use inner classes.

Course Objective	Competencies
3. Learn how to create graphical user interfaces (GUI), exceptions, file input/output (I/O), threads and networking.	<ol style="list-style-type: none">1. Define exceptions.2. Describe the use of keywords try, catch and finally.3. Describe exception categories.4. Identify common exceptions.5. Write code to handle your own exceptions.6. Write code to access command-line arguments and system properties.7. Read and write text to file streams.8. Describe the collections API.9. Use iterators.10. Identify deprecated classes and describe how to handle them during migration to Java 2 SDK.11. Describe the Abstract Windowing Toolkit (AWT) package and its components.12. Explain containers, components and layout managers and how they work together in a GUI.13. Use the flow and border managers to achieve a desired dynamic layout.14. Use the frame and panel containers.15. Place panels inside other containers to build complex layouts.16. Write code to handle events that occur in a user interface.17. Create the appropriate interface and handler method for a variety of event types.18. Determine the user action that originated the event from

Course Objective	Competencies
	<p>the event object details.</p> <ol style="list-style-type: none">19. Determine how and when to use the appropriate adapter class to select a subset of event handlers for an event listener.20. Identify the key AWT components and the event types that they produce.21. Control the colors and font used by an AWT component.22. Understand the purpose of the Swing GUI library.23. Describe a thread.24. Create separate threads, controlling the code and data that are used by that thread.25. Control the execution of a thread and write platform-dependent code with threads.26. Describe some of the difficulties that arise when multiple threads share data.27. Use the keyword synchronized to protect data from corruption.28. Use wait() and notify() to communicate between threads.29. Construct and use node streams.30. Distinguish Readers and Writers from Streams and select appropriately between them.31. Construct and use processing systems.32. Understand how to create your own processing stream classes.33. Read, write and update data in random access files.34. Use the Serialization interface to encode the state of an object to a stream and to implement object persistence.

Course Objective	Competencies
	35.Create a minimal TCP/IP client.