

Java Programming I

Course Design

2005-2006

Course Information

Organization	Eastern Arizona College
Division	Business
Course Number	CMP 126
Title	Java Programming I
Credits	3
Developed by	Alice Irwin
Lecture/Lab Ratio	2 Lecture/2 Lab
Transfer Status	ASU: CSE110, NAU: CS 126, UA: C SC127A
Activity Course	No
CIP Code	11.0100
Assessment Mode	Pre/Post Test (60 Questions/100 Points)
Semester Taught	Upon Request
GE Category	None
Separate Lab	No
Awareness Course	No
Intensive Writing Course	No

Prerequisites

None

Educational Value

Java Programming I is intended for computer majors in both vocational and transfer degree programs. It also benefits community students who are interested in learning object-oriented programming or the Java programming language.

Description

An introductory programming course for vocational computer majors and students transferring to universities in a business program. Course concentration includes object-oriented programming including data types, classes, objects, methods, decision and repetition structures, string and array manipulation, and Java applets. Computer majors are encouraged to take CMP 126 as their final programming language in their program of study. Students who have no previous computer programming experience should consider taking CMP 120 before taking CMP 126.

Supplies

USB flash drives and CD-RWs preferred. 3 1/2" Floppy Disks are acceptable but can be inconvenient due to capacity limitations.

Competencies and Performance Standards

1. Explain the concepts of object oriented programming..

Learning objectives

What you will learn as you master the competency:

- a. Define Java objects, methods, states, and classes.
- b. Apply concept of reusability.
- c. Apply concept of encapsulation.
- d. Employ Java access modifiers

Performance Standards

Competence will be demonstrated:

- in the completion of an executable sample Java program.
- in the written description of the concepts of object oriented programming.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner defines Java objects, methods, states, and classes.
- Learner applies concept of reusability.
- Learner applies concept of encapsulation.
- Learner employs Java access modifiers.

2. Demonstrate the ability to use Java data types within a Java program.

Learning objectives

What you will learn as you master the competency:

- a. Define various Java primitive data types including int, float, char, String, Boolean, and double.
- b. Define Java constants.
- c. Declare, assign, and initialize variables.
- d. Write simple arithmetic statements.
- e. Understand operator precedence.
- f. Understand arithmetic and logical operators.

Performance Standards

Competence will be demonstrated:

- in the successful execution of simple Java programs

- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner defines various Java primitive data types including int, float, char, String, Boolean, and double.
- Learner defines Java constants.
- Learner declares, assigns, and initializes variables.
- Learner writes simple arithmetic statements.
- Learner applies operator precedence.
- Learner applies arithmetic and logical operators.

3. Apply Java methods, classes, and objects.

Learning objectives

What you will learn as you master the competency:

- a. Code Java methods with and without arguments.
- b. Code Java methods that return values.
- c. Create and use Java classes, objects, and constructors.

Performance Standards

Competence will be demonstrated:

- in the completion of simple Java program exercises.
- in the coding and debugging of simple Java programs.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner codes Java methods with and without arguments.
- Learner codes Java methods that return values.
- Learner creates and uses Java classes, objects, and constructors.

4. Apply Java principles to accomplish data input, output, selection, and repetition structures.

Learning objectives

What you will learn as you master the competency:

- a. Write Java programs that will accept keyboard input.
- b. Use Java decision structures including IF, IF...ELSE, nested IF, and Switch statements.
- c. Use Java logical operators including AND, OR, and the conditional NOT

Performance Standards

Competence will be demonstrated:

- in coding and debugging simple code blocks and program files.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner writes Java programs that will accept keyboard input.
- Learner uses Java decision structures including IF, IF...ELSE, nested IF, and Switch statements.
- Learner uses Java logical operators including AND, OR, and the conditional NOT

5. Construct arrays within a Java program

Learning objectives

What you will learn as you master the competency:

- a. Declare and initialize a Java array.
- b. Correctly use subscripts within a Java array.
- c. Search and sort Java arrays.
- d. Pass arrays to methods
- e. Differentiate one-dimensional, two-dimensional, and multi-dimensional arrays

Performance Standards

Competence will be demonstrated:

- in the completion of coded programs using Java arrays.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner declares and initializes a Java array.
- Learner correctly uses subscripts within a Java array.
- Learner searches and sorts Java arrays.
- Learner passes arrays to methods
- Learner differentiates one-dimensional, two-dimensional, and multi-dimensional arrays

6. Debug simple Java programs.

Learning objectives

What you will learn as you master the competency:

- a. Apply correct order and problem solving techniques in debugging compilation errors.
- b. Recognize typical compilation error messages.
- c. Evaluate results of program execution to detect logic errors.

Performance Standards

Competence will be demonstrated:

- in the successful debugging, compilation, and execution of simple Java program assignments.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- a. Learner applies correct order and problem solving techniques in debugging compilation

errors.

- b. Learner recognizes typical compilation error messages.
- c. Learner evaluates results of program execution to detect logic errors.

7. Design and display simple Java applets.

Learning objectives

What you will learn as you master the competency:

- a. Write the code for a simple Java applet.
- b. Write the HTML file that hosts a simple Java applet.
- c. Use basic Java applet components including labels, textfields, and buttons.

Performance Standards

Competence will be demonstrated:

- when the student codes programs with alternative selection.
- when the programs with alternative selection function successfully.
- in multiple choice and practical quizzes and exams.

Criteria - Performance will be satisfactory when:

- Learner writes the code for a simple Java applet.
- Learner writes the HTML file that hosts a simple Java applet.
- Learner uses basic Java applet components including labels, textfields, and buttons.

Types of Instruction

Classroom Presentation

On Campus Laboratory and Clinical

Grading Information

Grading Rationale

Grades will be weighted in the following way:

Lab Assignments: 90%

Post Test: 10%

Grading Scale

A 90-100%

B 80 - 89%

C 70 - 79%

D 60 - 69%

F Below 60%