



**COWLEY COLLEGE
& Area Vocational Technical School**

COURSE PROCEDURE FOR

**JAVA PROGRAMMING
CIS1868 3 Credit Hours**

Student Level:

This course is open to students on the college level in either freshman or sophomore year.

Catalog Description of the Course:

CIS1868 - JAVA PROGRAMMING (3 hrs)

An introductory course to give computer science majors an introduction to programming in Java. Applications and applets will be created using a structured programming approach. Object oriented topics will be covered. Various problems will be solved using Java.

Prerequisite:

None

Controlling Purpose:

This course is offered to teach programming skills in Java to students. The basic constructs learned in this course will apply to any programming language (with slight modifications).

Learner Outcomes: Upon completion of this course, the student should be able to write applications using Java. The student should be able to use input/output statements, selection statements, repetition structures, procedures, and arrays. The student will learn how to incorporate exception handling into their programs. The student will learn how to use Java to enhance their web pages. Lastly, they will know how to write to and retrieve data from various sources.

Units Outcomes and Criterion Based Evaluation Key for Core Content:

The following defines the minimum core content not including the final examination period. Instructors may add other content as time allows.

Evaluation Key:

- A = All major and minor goals have been achieved and the achievement level is considerably above the minimum required for doing more advanced work in the same field.
- B = All major goals have been achieved, but the student has failed to achieve some of the less important goals. However, the student has progressed to the point where the goals of work at the next level can be easily achieved.
- C = All major goals have been achieved, but many of the minor goals have not been achieved. In this grade range, the minimum level of proficiency represents a person who has achieved the major goals to the minimum amount of preparation necessary for taking more advanced work in the same field, but without any major handicap of inadequacy in his background.

- D = A few of the major goals have been achieved, but the student's achievement is so limited that he is not well prepared to work at a more advanced level in the same field.
- F = Failing, will be computed in GPA and hours attempted.
- N = No instruction or training in this area.

CHAPTER 1: Introduction to Computers, the Internet and the Web						
Outcomes: Upon completion of this unit, the student will get an introduction to programming languages and specifically the Java environment.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand basic computer science concepts.
						Become familiar with different types of programming languages.
						Introduce a typical Java development environment.
						Understand Java's role in developing distributed client/server applications for the Internet and Web.
						Introduce object-oriented design with the UML and design patterns.

CHAPTER 2: Introduction to Java Applications						
Outcomes: Upon completion of this unit, the student will be able to write simple Java applications and know how to use the input/output and decision-making statements along with some of the operators and types.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Be able to write simple Java applications.
						Be able to use input and output statements.
						Become familiar with primitive types.
						Understand basic memory concepts.
						Be able to use arithmetic operators.
						Understand arithmetic-operator precedence.
						Be able to write decision-making statements.
						Be able to use relational and equality operators.

CHAPTER 3: Introduction to Java Applets						
Outcomes: Upon completion of this unit, the student will be able to write simple Java applets and understand when to use them versus applications.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Differentiate between applets and applications.
						Observe some of Java's capabilities through the Java 2 Software Development Kit's demonstration applets.
						Be able to write simple Java applets.
						Be able to write a simple HyperText Markup Language (HTML) document to load an applet into the appletviewer or a Web browser and execute the applet
						Understand the difference between variables and references.

CHAPTER 4: Control Statements: Part 1						
Outcomes: Upon completion of this unit, the student will be able to write programs that use the selection and repetition statements.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand basic problem-solving techniques.
						Be able to develop algorithms through the process of top-down, stepwise refinement.
						Be able to use the if and if...else selection statements to choose among alternative actions.
						Be able to use the while repetition statement to execute statements in a program repeatedly.
						Understand counter-controlled repetition and sentinel-controlled repetition.
						Be able to use the assignment, increment and decrement operators.

CHAPTER 5: Control Statements: Part 2						
Outcomes: Upon completion of this unit, the student will be able to write programs that use more advanced selection and repetition statements.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Be able to use the for and do...while repetition statements to execute statements in a program repeatedly.
						Understand multiple selection using the switch selection statement.
						Be able to use the break and continue program control statements.
						Be able to use the logical operators.

CHAPTER 6: Methods						
Outcomes: Upon completion of this unit, the student will be able to write programs that include methods. They will also understand the appropriate time to use methods.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand how to construct programs modularly from small pieces called methods.
						Introduce the common math methods available in the Java API.
						Be able to create new methods.
						Understand the mechanisms for passing information between methods.
						Introduce simulation techniques that use random-number generation.
						Understand how the visibility of declarations is limited to specific regions of programs.
						Understand how to write and use methods that call themselves.

CHAPTER 7: Arrays						
Outcomes: Upon completion of this unit, the student will be able to write programs that incorporate arrays.						
A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Introduce the array data structure.
						Understand the use of arrays to store, sort and search lists and tables of values.
						Understand how to declare an array, initialize an array and refer to individual elements of an array.
						Be able to pass arrays to methods.
						Be able to declare and manipulate multidimensional arrays.

CHAPTER 8: Object-Based Programming						
Outcomes: Upon completion of this unit, the student will understand the basic concepts of object based programming.						
B	C	D	F	N		Specific Competencies Demonstrate the ability to:
						Understand encapsulation and data hiding.
						Understand the notions of data abstraction and abstract data types (ADTs).
						Create Java ADTs – namely, classes.
						Be able to create and use objects.
						Be able to control access to instance variables and methods.
						Understand the use of the this reference.
						Be able to use class variables and methods.
						Understand the value of object orientation.

CHAPTER 9: Object-Oriented Programming: Inheritance
 Outcomes: Upon completion of this unit, the student will understand how inheritance works in object oriented programming.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand how inheritance promotes software reusability.
						Understand the notions of superclasses and subclasses.
						Understand access modified protected.
						Be able to access superclass members with super.
						Understand the use of constructors and finalizers in inheritance hierarchies.

CHAPTER 10: Object-Oriented Programming: Polymorphism
 Outcomes: Upon completion of this unit, the student will understand how polymorphism relates to object oriented programming.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand the concept of polymorphism.
						Understand how to use overridden methods to effect polymorphism.
						Distinguish between abstract and concrete classes.
						Learn how to declare abstract methods to create abstract classes.
						Appreciate how polymorphism makes systems extensible and maintainable.
						Be able to determine an object's type at execution time.

CHAPTER 11: Strings and Characters

Outcomes: Upon completion of this unit, the student will be able to programs that manipulate strings and charters.

A	B	C	D	F	N	Specific Competencies
						Demonstrate the ability to:
						Be able to create and manipulate nonmodifiable character string objects of class String.
						Be able to create and manipulate modifiable character string objects of class StringBuffer.
						Be able to create and manipulate objects of class Character.
						Be able to use a StringTokenizer object to break a String object into tokens.

CHAPTER 13: Graphical User Interface Components: Part 1

Outcomes: Upon completion of this unit, the student will be able to programs that use buttons, labels, lists, text fields, and panels. They will also be able to use layout managers and understand how to respond to mouse and keyboard events.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand the design principles of graphical user interfaces (GUI).
						Be able to build graphical user interfaces.
						Understand the packages containing GUI-related components, event-handling classes and interfaces.
						Be able to create and manipulate buttons, labels, lists, text fields and panels.
						Understand mouse events and keyboard events.
						Understand and be able to use layout managers.

CHAPTER 14: Graphical User Interface Components: Part 2

Outcomes: Upon completion of this unit, the student will be able to write programs that use text areas, sliders, menus, popup menus, windows, JPanel objects, and additional layout managers. Also the student will get an introduction to Swing.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Create and manipulate text areas, sliders, menus, popup menus and windows.
						Be able to create customized JPanel objects.
						Be able to change the look-and-feel of a GUI, using Swing's pluggable look-and-feel (PLAF).
						Be able to create a multiple-document interface with JDesktopPane and JInternalFrame.
						Be able to use additional layout managers.

CHAPTER 15: Exception Handling

Outcomes: Upon completion of this unit, the student will be able write programs that have error checking .

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand exception and error handling.
						Use try, throw and catch to detect, indicate and handle exceptions, respectively.
						Use the finally clause to release resources.
						Understand the Java exception hierarchy.
						Declare new exception classes.
						Create chained exceptions.

CHAPTER 17: Files and Streams

Outcomes: Upon completion of this unit, the student will be able to write programs that manipulate files in various ways. Also they will gain an understanding of Streams.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Be able to create, read, write and update files.
						Be able to use class File.
						Understand the Java streams class hierarchy.
						Be able to use the FileInputStream and FileOutputStream classes.
						Be able to use a JFileChooser dialog to access files and directories.
						Be able to use the ObjectInputStream and ObjectOutputStream classes.
						Be able to use class RandomAccessFile.
						Become familiar with sequential-access and random-access file processing.

CHAPTER 23: Java Database Connectivity with JDBC

Outcomes: Upon completion of this unit, the student will be able to write programs that access a relational database.

A	B	C	D	F	N	Specific Competencies Demonstrate the ability to:
						Understand relational databases.
						Understand basic database queries using SQL.
						Use the classes and interfaces of package java.sql to manipulate databases.

Projects Required:

Projects will vary according to the instructor.

Text Book:

Contact bookstore for current textbook information.

Materials/Equipment Required:

Traditional Classroom Delivery: None.

Attendance Policy:

Students should adhere to the attendance policy outlined by the instructor in the course syllabus.

Grading Policy:

The grading policy will be outlined by the instructor in the course syllabus.

Maximum Class Size:

Based on classroom occupancy.

Course Time Frame:

The U.S. Department of Education, Higher Learning Commission and the Kansas Board of Regents define credit hour and have specific regulations that the college must follow when developing, teaching and assessing the educational aspects of the college. A credit hour is an amount of work represented in intended learning outcomes and verified by evidence of student achievement that is an institutionally-established equivalency that reasonably approximates not less than one hour of classroom or direct faculty instruction and a minimum of two hours of out-of-class student work for approximately fifteen weeks for one semester hour of credit or an equivalent amount of work over a different amount of time. The number of semester hours of credit allowed for each distance education or blended hybrid courses shall be assigned by the college based on the amount of time needed to achieve the same course outcomes in a purely face-to-face format.

Refer to the following policies:

[402.00 Academic Code of Conduct](#)

[263.00 Student Appeal of Course Grades](#)

[403.00 Student Code of Conduct](#)

Disability Services Program:

Cowley College, in recognition of state and federal laws, will accommodate a student with a documented disability. If a student has a disability which may impact work in this class which requires accommodations, contact the Disability Services Coordinator.