NORTHWESTERN CONNECTICUT COMMUNITY COLLEGE

COURSE SYLLABUS

Course Title: Java Programming I   **Course #:** CSC* 223

Course Description: 4 Credits**.**   A course in the Java programming language that uses modular programming and emphasizes object oriented techniques. This course will address event driven programming in the Windows environment and interfaces to the internet. This course may fulfill an advanced programming language requirement.

Pre-requisite CSC* 104

Goals: Students are expected to

- discuss the history of the Java programming language
- construct executable and reusable program modules
- apply various programming paradigms
- apply algorithm design and development techniques
- apply sequence, selection and iteration control structures to program development
- apply standard debugging tools to problem solutions
- construct program code using problem-solving techniques and tools
- create specifications for testing and debugging
- construct program solutions using various data forms and basic data structures
- produce technical documentation

Outcomes: Upon successful completion of this course students will be able to:

1) Demonstrate an understanding of computing basics
   a) Describe computer basics and operating systems
   b) Explain number representation in multiple formats
   c) Explain the relationship between Java and the World Wide Web
   d) Distinguish the terms API, IDE, and JDK

2) Demonstrate the basic steps in developing a Java program
   a) Explain the basic syntax of a Java Program
   b) Write, run and save a simple Java project
   c) Demonstrate performing calculations
   d) Demonstrate the use of variables, constants, methods and classes
   e) Distinguish syntax errors, runtime errors, and logic errors
   f) Explain correct programming style and naming conventions
   g) Explain the correct usage of various data types
   h) Identify the correct usage for Boolean data types
   i) Apply relational and logic operators to write Boolean expressions
   j) Demonstrate the use of Boolean expressions in control statements
   k) Implement select control
   l) Write expressions using the conditional operator
   m) Display formatted output

3) Demonstrate the use of iterative operations
   a) Use While, and For loops to control repetition of statements
   b) Explain the control of flow in loop statements
   c) Use Boolean expressions to control loop statements
   d) Explain the differences between different types of loops

e) Write nested loops
f) Demonstrate techniques for minimizing numerical errors
g) Implement program control with Break and Continue

4) Demonstrate the creation and use of methods
   a) Define and invoke methods
   b) Develop modular code that is easy to read, debug and maintain
   c) Determine the scope of variables
   d) Solve mathematical problems using methods
   e) Apply the concept of method abstraction to software development

5) Demonstrate the use of arrays in processing data
   a) Describe why arrays are necessary in programming
   b) Describe the steps in using arrays
   c) Create arrays and declare array reference variables
   d) Initialize the variables in an array
   e) Manipulate array contents
   f) Develop and invoke methods with array arguments
   g) Implement sorting and searching algorithms
   h) Use the methods in the Array class

6) Demonstrate an understanding of using objects
   a) Describe objects and classes, and use classes to model objects
   b) Demonstrate the use of UML notation
   c) Construct objects using constructors
   d) Access objects using object reference variables
   e) Define a reference variable using s reference type
   f) Access an object's data and methods
   g) Declare a class and create an object from a class
   h) Distinguish between instance and static variables and methods
   i) Encapsulate data fields
   j) Develop methods with object arguments
   k) Use the String class to process fixed strings
   l) Use the Character class to process a single character
   m) Distinguish among the String classes
   n) Explain how to pass command line arguments
   o) Explain the file properties using the File class
   p) Read and write data using the PrintWrtiter and Scanner classes
   q) Open files using a dialog box
   r) Create immutable objects from immutable classes
   s) Determine the scope of variables ion the context of a class
   t) Use the keyword This to refer to the calling object itself
   u) Apply class abstraction to develop software
   v) Explain the differences between procedural and object-oriented paradigms
   w) Design programs using object-oriented paradigms

7) Demonstrate the manipulation of classes
   a) Develop a subclass from a superclass through inheritance
   b) Illustrate inheritance
   c) Invoke the superclasses constructors and methods
   d) Override instance methods in the subclass
   e) Distinguish between overriding and overloading
   f) Explain polymorphism, dynamic binding, and generic programming

g) Explain casting and why explicit downcasting is necessary
h) Implement a Stack class
i) Restrict access to data and methods to subclasses
j) Prevent class extending and method overriding
k) Design and use abstract classes

8) Demonstrate an understanding of interfaces
   a) Specify common behavior for objects using interfaces
   b) Define interfaces and declare classes that implement interfaces
   c) Define a natural order
   d) Enable objects to listen for action events
   e) Make objects cloneable
   f) Explain the differences between an abstract class and an interface
   g) Create objects for primitive types using wrapper classes
   h) Create a generic sort method

9) Demonstrate an understanding of software development
   a) Describe the software life cycle
   b) Determine responsibilities for a class
   c) Describe relationship types
   d) Declare classes that represent relationships
   e) Design classes that follow the class-design guidelines
   f) Explain framework-based programming using the Java API
   g) Apply design patterns for developing sound software systems

10) Demonstrate creating GUI applications
    a) Distinguish between Swing and AWT
    b) Describe the Java GUI API hierarchy
    c) Create GUI interfaces
    d) Explain the role of layout managers
    e) Use layout manages to layout components in a container
    f) Apply common features on Swing components
    g) Use borders to visually group user interface components
    h) Create image icons
    i) Describe Java coordinate systems in a GUI component
    j) Draw things using methods
    k) Explain how and when a Graphics object is created
    l) Use a panel as a canvas to draw things
    m) Obtain font properties and center messages
    n) Display an image in a GUI component
    o) Develop reusable GUI components

11) Demonstrate the management of events
    a) Describe events, event sources, and event classes
    b) Declare listener classes and register listener objects
    c) Write programs to deal with ActionEvent and WindowEvent
    d) Simplify coding for listener classes using interface adaptors
    e) Write programs to deal with MouseEvent and KeyEvent
    f) Use classes to control animation
    g) Create graphical user interfaces using various components
    h) Create listeners for various types of events
    i) Explain the use of various interface objects
    j) Display multiple windows in an application

12) Demonstrate the use of applets
   a) Explain how the Web browser controls and executes applets
   b) Describe the methods of the Applet class
   c) Develop Swing applets
   d) Embed applets in web pages
   e) Write Java programs that can run as both applications and applets
   f) Run applets from the appletviewer and from a browser
   g) Pass values to applets from HTML
   h) Locate resources using the URL class
   i) Play audio in a Java program

13) Demonstrate the skills in managing exceptions
   14) Explain exceptions and exception handling
   15) Explain the advantages of exception handling
   16) Distinguish between exception types
   17) Declare exceptions in a method header
   18) Throw exceptions in a method
   19) Write a try-catch block to handle exceptions
   20) Create chained exceptions
   21) Declare custom exception clauses

College Policies

Plagiarism: Plagiarism and Academic Dishonesty are not tolerated at Northwestern Connecticut Community College. Violators of this policy will be subject to sanctions ranging from failure of the assignment (receiving a zero), failing the course, being removed/expelled from the program and/or the College. Please refer to your "Student Handbook" under "Policy on Student Rights," the Section entitled "Student Discipline," or the College catalog for additional information.

Americans with Disabilities Act (ADA): The College will make reasonable accommodations for persons with documented learning, physical, or psychiatric disabilities. Students should notify Dr. Christine Woodcock, the Counselor for Students with Disabilities. She is located at Green Woods Hall, in the Center for Student Development. Her phone number is 860-738-6318 and her email is cwoodcock@nwcc.edu.

School Cancellations: If snowy or icy driving conditions cause the postponement or cancellation of classes, announcements will be made on local radio and television stations and posted on the College's website at www.nwcc.edu. Students may also call the College directly at **(860) 738-6464** to hear a recorded message concerning any inclement weather closings. Students are urged to exercise their own judgment if road conditions in their localities are hazardous.

Use of Electronic Devices: Some course content as presented in Blackboard Learn is not fully supported on mobile devices at this time. While mobile devices provide convenient access to check in and read information about your courses, they should not be used to perform work such as taking tests, quizzes, completing assignments, or submitting substantive discussion posts.

Sexual Assault and Intimate Partner Violence Resource Team: NCCC is committed to creating a community that is safe and supportive of people of all gender and sexual identities. This pertains to the entire campus community, whether on ground or virtual, students, faculty, or staff.

Sexual assault and intimate partner violence is an affront to our national conscience, and one we cannot ignore. It is our hope that no one within our campus community will become a victim of these crimes. However, if it occurs, NCCC has created the SART Team - Sexual Assault and Intimate Partner Violence Resource Team - to meet the victim's needs.
SART is a campus and community based team that is fully trained to provide trauma-informed compassionate service and referrals for comprehensive care. The team works in partnership with The Susan B. Anthony Project to extend services 24 hours a day, 7 days a week throughout the year.

The NCCC team members are:
Ruth Gonzalez, Ph.D.	860-738-6315	Green Woods Hall Room 207
Susan Berg	860-738-6342	Green Woods Hall Room 223
Kathleen Chapman	860-738-6344	Green Woods Hall Room 110
Michael Emanuel	860-738-6389	Founders Hall Annex Room 308
Seth Kershner	860-738-6481	Library
Jane O'Grady	860-738-6393	Founders Hall Annex Room 212
Robin Orlomoski	860-738-6416	Business Office Room 201
Patricia Bouffard, Ex-Officio	860-738-6319	Founders Hall Room 103
Savannah Schmitt		Student Representative

At NCCC we care about our students, staff and faculty and their well-being. It is our intention to facilitate the resources needed to help achieve both physical and emotional health.