

Course Prefix/Number: CPT 236
Course Title: Introduction to Java Programming
Lec Hours/Week: 3.0
Lab Hours/Week: 0.0
Credit Hours/Semester: 3.0

[Distance Learning Attendance/VA Statement](#)
[Textbook Information](#)

COURSE DESCRIPTION

This course is an introduction to Java programming. Topics will cover Java syntax and classes for use in the development of Java applications and applets.

COURSE COMPETENCIES

Upon successful completion of this course, the student should be competent to complete the following tasks:

Module 1 – Object Oriented Programming, Operators, Primitive Types, and Strings

- Explain the concept of classes.
- Explain the relationship between classes and objects.
- Describe the difference between Java applications and applets.
- Create a Java application to produce output at the command prompt.
- Create a Java applet to produce output in a label.
- Obtain user input and produce output.
- Describe the differences between the primitive data types used in Java.
- Declare and initialize variables.
- Evaluate mathematical expressions using arithmetic and assignment operators to and store the resulting value in a variable.
- Invoke methods from the String class to compare and manipulate String variables. String methods used should include: length, equals, equalsIgnoreCase, toLowerCase, toUpperCase, substring, indexOf, and compareTo.
- Document code with well-placed, descriptive comments.
- Find and fix syntax and logic errors in programs that produce input and output and use arithmetic operations.

Module 2 - Control Structures

- Create logical expressions with relational and logical operators.
- Solve problems requiring the use of nested if/else statements.
- Solve problems requiring the use of while or do/while loops.
- Solve problems using for loops.
- Solve problems using switch statements.
- Find and fix syntax and logic errors in programs containing decisions and loops.

Module 3 - Methods and Classes

- Write a method call given the structure of a pre-defined method.
- Explain when parameters are sent by value and when parameters are sent by reference
- Relate the effect of methods on program design and testing.
- Write methods to perform stated tasks.
- Overload methods and describe the benefits of overloading.
- Find and fix syntax and logic errors in programs which use pre-defined and user-defined methods.
- Describe the design benefits of user-defined classes.
- Relate the differences between static and non-static class members.
- Define and create constructors.
- Define and create accessor and mutator functions.
- Define the this reference.
- Create packages containing related classes.

Module 4 – Inheritance and Polymorphism

- Explain the concepts of inheritance and composition.
- Explain the concept of polymorphism.
- Describe the difference between public, protected, and private data members.
- Write classes illustrating the concepts of inheritance and polymorphism

Module 5 – Arrays

- Declare and initialize single-subscripted and multi-subscripted arrays
- Input, output and manipulate the values in an array
- Create methods that take arrays as parameters
- Find and fix syntax and logic errors in programs containing arrays

MINIMAL STANDARDS

Minimal standards of performance on all course competencies for receiving credit for the course and indicated by 60% overall accuracy on evaluation instruments that address the course competencies listed above. Required standards of performance on all course competencies for enrollment in subsequent higher-level computer technology courses are indicated by 70% overall accuracy on evaluation instruments that address the course competencies listed above.

COURSE REQUIREMENTS

Students are responsible for attending all schedule class meetings until they have completed all course requirements. Students are responsible for all material covered and for all assignments made in all classes. Any student caught cheating or involved in other academic dishonesty will be given a grade of zero and will be subject to further disciplinary action.

ATTENDANCE POLICY

The attendance policy as stated in the York Technical College Handbook will be enforced. Make-up tests will not be given for theory tests. If a student must miss a theory test, he/she will get a zero for that test. However, students have the option of taking the comprehensive final. The student's grade on the comprehensive final will replace his/her lowest theory test grade. It is the student's responsibility to schedule a time for a make-up hands-on test with his/her instructor.

EVALUATION STRATEGIES/GRADING

Module 1 (20% total)	Module 2 (20% total)	Grading Scale	
		90-100	A
Test – 10% Program(s) /Homework – 10%	Test – 10% Program(s) /Homework – 10%	89-89	B
Module 3 (20% total)	Module 4 (20% total)	70-79	C
Test – 10% Program(s) /Homework – 10%	Test – 10% Program(s) /Homework – 10%	60-69	D
Module 5 (20% total)		Below 60	F
Test – 10% Program(s) /Homework – 10%			

*Completion of Module 5 Program(s) is required to receive a grade for the course

ENTRY LEVEL SKILLS

A student entering this course should be familiar with structured programming concepts, have adequate flowcharting skills and be familiar with the Windows environment.

PREREQUISITES: CPT 114 and CPT 168

CO-REQUISITES: None

Disabilities Statement: Any student who feels s/he may need an accommodation based on the impact of a disability should contact the Special Resources Office (SRO) at 803-327-8007 in the 300 area of Student Services. The SRO coordinates reasonable accommodations for students with documented disabilities.