

Course Outline

School:	Eng. Tech. & Applied Science
Department:	Information and Communication Engineering Technology (ICET)
Course Title:	Java Programming
Course Code:	COMP 228
Course Hours/Credits:	56
Prerequisites:	COMP 123
Co-requisites:	N/A
Eligible for Prior Learning, Assessment and Recognition:	Yes
Originated by:	ILIA NIKA
Creation Date:	Fall 2004
Revised by:	ILIA NIKA
Revision Date:	Summer 2016
Current Semester:	Fall 2016
Approved by:	<i>p pesikan</i> <i>l c/o</i>

Chairperson/Dean

Students are expected to review and understand all areas of the course outline.

Retain this course outline for future transfer credit applications. A fee may be charged for additional copies.

This course outline is available in alternative formats upon request.

Course Description

Building on fundamentals of Object-Oriented programming, this course covers Component-based Application Development and Database Connectivity using the Java programming language. Coursework emphasizes how to create complex GUI applications with Java programming language, interacting with the databases using JDBC, and Java collections framework.

Program Outcomes

Successful completion of this and other courses in the program culminates in the achievement of the Vocational Learning Outcomes (program outcomes) set by the Ministry of Training, Colleges and Universities in the Program Standard. The VLOs express the learning a student must reliably demonstrate before graduation. To ensure a meaningful learning experience and to better understand how this course and program prepare graduates for success, students are encouraged to review the Program Standard by visiting <http://www.tcu.gov.on.ca/pepg/audiences/colleges/progstan/>. For apprenticeship-based programs, visit <http://www.collegeoftrades.ca/training-standards>.

Course Learning Outcomes

The student will reliably demonstrate the ability to:

1. Have a good understanding of Java SE 8 Platform
2. Have a good understanding of advanced OOP features in Java
3. Design, code and test the front end of a Java application that uses advanced GUI components
4. Design, code and test a Java application that uses exception handling techniques to produce stable and robust code.
5. Design, code and test a Java application that implements data access capabilities using JDBC and allows the user to retrieve, insert, update, and delete database records.
6. Design, code and test a Java application that uses multithreading
7. Design, code and test a Java application that uses Java Collections Framework

Essential Employability Skills (EES)

The student will reliably demonstrate the ability to*:

1. Communicate clearly, concisely and correctly in the written, spoken, and visual form that fulfills the purpose and meets the needs of the audience.
4. Apply a systematic approach to solve problems.
5. Use a variety of thinking skills to anticipate and solve problems.
10. Manage the use of time and other resources to complete projects.

**There are 11 Essential Employability Skills outcomes as per the Ministry Program Standard. Of these 11 outcomes, the following will be assessed in this course.*

Global Citizenship and Equity (GC&E) Outcomes

N/A

Text and other Instructional/Learning Materials

Text Book(s):

Required:

Deitel & Deitel, Java How to Program, 10/e (Early Objects) , ISBN-10: 0133807800, ISBN-13: 978-0133807806, Edition: 10 , 2015.

Reference:

Big Java: Early Objects, 5th Edition, ISBN : 978-1-118-54599-7, January 2013.

Evaluation Scheme

- ✦ Test 1: Hands-On and MC questions covering Week 1 - 6 materials
- ✦ Test 2: Hands-On and MC questions covering Week 8 - 13 materials
- ✦ Lab Assignment 1: Developing Simple Java Applications
- ✦ Lab Assignment 2: Creating and Using Java Methods
- ✦ Lab Assignment 3: Using Inheritance and Polymorphism in Java Applications
- ✦ Lab Assignment 4: Developing GUI applications with Java
- ✦ Lab Assignment 5: Developing database applications using JDBC.
- ✦ Lab Assignment 6: Using Multithreading and generic collections in Java applications

Evaluation Name	CLO(s)	EES Outcome(s)	GCE Outcome(s)	Weight/100
Test 1	1, 2	1, 4, 5		25
Test 2	3, 4, 5, 6, 7	1, 4, 5, 10		25
Lab Assignment 1	1, 2	4, 5, 10		5
Lab Assignment 2	1, 2	4, 5, 10		5
Lab Assignment 3	1, 2	4, 5, 10		7
Lab Assignment 4	2, 3	4, 5, 10		10
Lab Assignment 5	3, 4, 5	4, 5, 10		15
Lab Assignment 6	6, 7	4, 5, 10		8
Total				100%

If students are unable to write a test they should immediately contact their professor or program Chair for advice. In exceptional and well documented circumstances (e.g. unforeseen family problems, serious illness, or death of a close family member), students may be able to write a make-up test.

All submitted work may be reviewed for authenticity and originality utilizing Turnitin®. Students who do not wish to have their work submitted to Turnitin® must, by the end of the second week of class, communicate this in writing to the instructor and make mutually agreeable alternate arrangements.

When writing tests, students must be able to produce official College photo identification or they may be refused the right to take the test or test results will be void.

Student Accommodation

Students with permanent or temporary accommodations who require academic accommodations are encouraged to register with the Centre for Students with Disabilities (CSD) located at Ashtonbee (L1-04), Progress (C1-03), Morningside (Rm 190), and Story Arts Campus (Rm 284). Documentation outlining the functional limitations of a disability is required; however, interim accommodations pending receipt of documentation may be possible. This service is free and confidential. For more information, please email

csd@centennialcollege.ca.

Use of Dictionaries

- Any dictionary (hard copy or electronic) may be used in regular class work.
- Dictionaries may be used in tests and examinations, or in portions of tests and examinations, as long as they are non-electronic (not capable of storing information) and hard copy (reviewed by the invigilator to ensure notes are not incorporated that would affect test or examination integrity).

Program or School Policies

N/A

Course Policies

N/A

College Policies

Students should familiarize themselves with all College Policies that cover academic matters and student conduct.

All students and employees have the right to study and work in an environment that is free from discrimination and harassment and promotes respect and equity. Centennial policies ensure all incidents of harassment, discrimination, bullying and violence will be addressed and responded to accordingly.

Academic honesty is integral to the learning process and a necessary ingredient of academic integrity. Academic dishonesty includes cheating, plagiarism, and impersonation. All of these occur when the work of others is presented by a student as their own and/or without citing sources of information. Breaches of academic honesty may result in a failing grade on the assignment/course, suspension or expulsion from the college.

For more information on these and other policies, please visit www.centennialcollege.ca/about-centennial/college-overview/college-policies.

Students enrolled in a joint or collaborative program are subject to the partner institution's academic policies.

PLAR Process

This course is eligible for Prior Learning Assessment and Recognition (PLAR). PLAR is a process by which course credit may be granted for past learning acquired through work or other life experiences. The PLAR process involves completing an assessment (portfolio, test, assignment, etc.) that reliably demonstrates achievement of the course learning outcomes. Contact the academic school to obtain information on the PLAR process and the required assessment.

This course outline and its associated weekly topical(s) may not be reproduced, in whole or in part, without the prior permission of Centennial College.

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
1	Introduction to Java Programming language. Anatomy of a Java Application. Eclipse IDE.	Chapter 1,2	Understand Java 8 Platform. Explain the concept of JVM. Write simple Java applications with I/O capabilities using Eclipse IDE.	Lecture Demonstration Lab Session		
2	Classes and Objects in Java. Instance variables and instance methods. Primitives and Reference Types	Chapter 3	Declare classes and use them to create objects. Implement class's attributes and behaviors. Differentiate between local and instance variables. Use constructors to initialize data. Call object's methods to perform their tasks.	Lecture Demonstration Lab Session	Lab Assignment 1: Creating and using simple Java classes.	Week 2
3	Methods in Java Anatomy of a Java method. Static methods. Java API packages. Method overloading enum type.	Chapter 6, 7	Create and use programmer defined methods in Java classes. Declare static variables. Create and use static methods in Java classes. Implement method overloading. Use arrays and pass them as arguments to Java methods.	Lecture Demonstration Lab Session		
4	Advanced Object-oriented programming in Java. A deeper look at Java classes.	Chapter 8	Create and use Java classes as a means of developing component-based applications. Use 'this' keyword. Use composition in Java applications. Define and use enum type in Java classes. Use package access.	Lecture Demonstration Lab Session	Lab Assignment 2: Creating and using Java methods	Week 4

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
			<p>Use final instance variables.</p> <p>Design Java applications composed of multiple classes.</p>			
5	Inheritance In Java	Chapter 9	<p>Understand the concept of inheritance and how to use it to create new classes based on existing classes.</p> <p>Understand the concept of superclasses and subclasses.</p> <p>Use protected keyword in superclass to give subclass methods access to superclass members.</p> <p>Understand how constructors are used in inheritance hierarchies.</p> <p>Develop a Java application that implements an inheritance hierarchy.</p>	Lecture Demonstration Lab Session		
6	Polymorphism in Java Abstract classes Interfaces	Chapter 10	<p>Understand the polymorphism concept. Use overridden methods to implement polymorphism.</p> <p>Define and use abstract classes.</p> <p>Define and implement interfaces.</p> <p>Develop Java applications that utilize the abstract and interface polymorphism concepts.</p>	Lecture Demonstration Lab Session	Lab Assignment 3: Using Inheritance and Polymorphism in Java applications	Week 6
7	Introduction to GUI Programming and Swing API: • Simple Swing components • Event handling	Chapter 12	<p>Understand the Graphical User Interfaces (GUI) concepts and event-handling mechanism in Java. Define common GUI event types, and listener interfaces.</p> <p>Write simple GUI applications that utilize the following Swing components: - JTextField - JLabel</p>	Lecture Demonstration	Test 1 Hands-On, covering week 1-6 topics.	

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
			- JTextBox - JButton			
8	Standard Swing components Layout Managers	Chapter 12	Design GUI applications that use JList, JRadioButton, JCheckBox, JComboBox, JTextArea, and JPanel components. Use layout managers FlowLayout, BorderLayout and GridLayout to arrange GUI components.	Lecture Demonstration Lab Session		
9	JavaFX Basics	Lecture Handout Chapter 25	Understand JavaFX. Develop JavaFX application using layout panes, buttons, labels, text fields, colors, fonts, images, image views, and shapes.	Lecture Demonstration Lab Session Demonstration	Lab Assignment 4: Developing GUI applications with Java	Week 9
10	Exception Handling Introduction to Data Access with JDBC Connection, Statement and ResultSet interfaces	Chapter 11, 24	Understand Exception Handling mechanism in Java. Use try and catch blocks to detect and handle exceptions. Use throw statement to indicate a problem. Use finally block to release the resources. Understand JDBC API to access databases. Create Java applications that establish a connection to a database and retrieve data from its tables.	Lecture Demonstration Lab Session		
11	Inserting, Updating and Deleting rows Prepared Statements RowSet interface	Chapter 24	Write Java applications that insert, update or delete database records. Use PreparedStatement. Use the RowSet interface to create a disconnected set of records and process multiple tables.	Lecture Demonstration Lab Session	Lab Assignment 5: Developing database applications using JDBC. Students can work in pairs for this assignment.	Week 11
12	Multithreading in Java	Chapter 23	Understand multithreading and its implementation in Java Applications.	Lecture Demonstration		

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
			Understand the life-cycle of a thread. Use Thread class and Runnable interface. Write multithreaded applications in Java.	Lab Session		
13	Java Collections Framework	Chapter 16	Understand Java Collections Framework. Use generic collection classes: ArrayList, LinkedList, Queue, Stack, HashSet, and HashMap in Java Applications.	Lecture Demonstration Lab Session	Lab Assignment 6: Using Multithreading and generic collections in Java applications.	Week 13
14	Test 2	Chapter 12,16,23,24, 25	Develop Complex Java applications	Review materials for Test 2	Test 2 Hands-On, covering week 7-13 topics.	Week 14