Software Development Tools	Scope and Sequence
Unit Lesson	**Objectives**
INTRODUCTION TO SOFTWARE DEVELOPMENT TOOLS	
Coding Standards and Conventions	
	Examine coding standards and conventions.
	Assess design processes and methodology.
	Distinguish between software types and function.
Software Processes and Methodology	
	Analyze the role of the IPO in other structural processes and methods.
	Evaluate the elements of the SDLC model systems development life cycle.
	Apply standard software processes and methodology in creation of a functional IPO chart.
Project: Grades Projection IPO	
Software Types and Elements	
	Relate the logic of the IPO chart to the programming code action statements.
	Apply the principles of programming flow control statements to action code statements in programming language code.
	Distinguish between software types and best-use purposes.
	Research and compare programming languages.
Project: Software Types and Elements	
Multimedia and Graphics Software Applications	
	Understand how the basic picture elements present images.
	Understand the mechanics of determining x, y coordinates.
	Evaluate various multimedia tools.
	Understand the mechanics of automation.

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Web-Based Software Applications	
		Evaluate the pros and cons of various software development tools.
		Apply multimedia using software tools for applications or for web pages.
		Understand the mechanics of determining x, y coordinates for presentation on the web.
		Understand the importance of integrating on-screen and web page coding to achieve results in the Cloud.
	Project: Multimedia and Web Design Careers	
	Software Design Principles and Tools	
		Apply flowcharting relationships using software tools.
		Apply IPO logic to flowcharting, and examine the similarities in Unified Modeling Language (UML).
		Evaluate the use of software development tools specifically designed to produce language-specific code from pseudocode.
	Project: Software Design Principles Table	
	Unit 1 Test	
SOFTWARE DEVELOPMENT		
	Personal Information Management (PIM) Tools	
		Evaluate the pros and cons of using a free personal information management (PIM) tool.
		Experiment with a mind-mapping tool to hone critical thinking skills.
		Employ a prominent personal information management (PIM) tool.
		Compare the benefits versus the risks of providing personal information via readily available PIM tools.
	Project: My Mind-Mapping	
	Computer Security Application Tools	

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Differentiate between types of security violations.
		Examine email options and preferences to filter spam.
		Evaluate available software security tools.
	Individual Programming Development Tools	
		Evaluate the pros and cons of cloud-based applications.
		Differentiate between streamed documents and Web-based documents.
		Evaluate the merits of streamed, installed-streamed, and installed applications.
	Project: Assessment of Competitive Office Suites	
	Database Software Development Tools	
		Interpret a provided Excel structure in terms of database terminology.
		Evaluate the pros and cons of various popular database software applications.
		Experiment with the use of various databases to create a table.
		Interpret programming code that creates a database object.
	Web Design Software Development Tools	
		Create a free web domain.
		Design a website page.
		Examine Hypertext Markup Language tag coding.
		Demonstrate understanding of how to upload a file to student's hosted website.
	Project: My Personal Website	
	Integrated Development Environments (IDEs)	
		Explain the difference between individual applications and integrated applications.
		Experiment and demonstrate the results of working with at least one enhanced text editor.

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Evaluate and debate the pros and cons of plain text editors versus enhanced text editors.
		Experiment with the use of at least one IDE and demonstrate results.
	Project: My Text Editor IDE Evaluation	
	Unit 2 Test	
DEBUGGING		
	Download, Install, Explore IntelliJ IDEA	
		Examine the interface and features of intelliJ IDEA.
		Distinguish between manual coding and automatic, intuitive functions of the intelliJ IDEA IDE.
		Evaluate the convenience of the built-in IDE functions versus manual coding.
	Download, Install, Explore NetBeans	
		Examine the interface and features of the NetBeans IDE.
		Distinguish between manual coding and automatic code generation when using the graphical user interface (GUI) interface option.
		Evaluate the convenience of the built-in IDE functions versus manual coding.
		Create a simple program using the IDE and assess its function.
	Project: MY IntelliJ NetBeans IDE Evaluation	
	Download, Install, Explore Eclipse	
		Examine the interface and features of Eclipse.
		Distinguish and explain the differences between manual coding and automatic, intuitive functions of the Eclipse IDE.
		Evaluate the convenience of the built-in IDE functions versus manual coding and describe the strengths and weaknesses of each.
		Create a simple program using the IDE and assess its function.

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Project: MY IntelliJ NetBeans Eclipse IDE Evaluation	
	Exceptions	
		Recognize what an exception is and procedures for writing code that is used to handle these exceptions.
	Project: Best Practices in Exception Handling in Java Programming	
	STDIN and STDOUT	
		Create applications reading data from files and the command line.
		Read and write bytes and characters from STDIO, files, and sockets.
	File Input, Output, and Network Input, Output	
		Learn how data input and output works with files.
		Learn how data input and output works with networks.
	Project: Concepts of File I/O and Network I/O	
	Unit 3 Test	
SOFTWARE CONFIGURATION MANAGEMENT		
	Code Blocks	
		Divide design specifications into logical process blocks.
		Understand variable scoping.
	Project: Concepts of Programming Code Structure in Java	
	Iterative Loops	
		Describe repetition (looping) control structures in Java.
		Examine how to construct counter, sentinel, flag, and EOF controlled repetition structures.
		Understand and explain break and continue statements.

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Discover how to minimize errors in looping.
		Describe how to code and use nested loops.
	For-Each Loops	
		Examine how to construct For and For Each loops.
		Describe control structures in Java.
		Discover how to minimize errors in looping.
		Evaluate logical expressions in Java using operators.
	Project: Computing Class Grades	
	Java Logic	
		Plan a programming project.
		Use logic to organize and outline a project.
		Describe the steps used to document the steps in application project planning.
		Explain how to use Java logic.
	If, Else If, Else	
		Evaluate logical expressions in Java using operators.
		Differentiate between assignment “=” and comparison “==”.
		Understand logic and primitives comparison use of “==”.
		Understand logic and object comparison.
		Understand use of the equals() method.
		Understand what == compares when used on objects*.
		Understand creating equals() methods objects.
		Understand strings are not primitive, must use equals.

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Understand arrays are not primitive, must compare each element.
		Understand lists are not primitive, must compare each element.
		Understand If, If…Else statements.
	Project: Write an IF...ELSE Program that Computes the New Salary for the CIO	
	Switch Statements	
		Understand how to structure and code SWITCH statements.
		Recognize when a SWITCH statement is the best choice in selection structure.
		Understand searching and sorting in Java.
	Project: Write a Program Using a SWITCH Statement	
	Unit 4 Test	
OBJECT MODELING UML AND SOFTWARE TESTING		
	Swing and AWT	
		Translate data structure and program design into code in an appropriate language.
		Demonstrate knowledge of key constructs and commands specific to a language.
		Develop an application that responds to user input.
	Creating Frames and Dialog Boxes, Components, Form Fields, Panels, Buttons	
		Understand frames and dialog boxes.
		Understand developing an application that responds to user input.
		Understand components, form fields, panels, buttons.
	Project: Building Better Java using GUI Applications, Frames, Containers, and Dialogs	
	HTML and Web Pages	

Software Development Tools		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Understand HTML markup language.
		Understand basic HTML tags.
		Understand beginning website concepts.
	Project: Creating a Web Page	
	Business Information System Trends, Applications, and eCommerce	
		Understand the need and use of business information systems.
		Understand systems applications development.
		Understand eCommerce and emerging technology trends in business.
	Project: Social Media on Campus	
	Application Servers and JavaServer Pages (JSP)	
		Understand the virtual environment and the need for online content.
		Recognize dynamic content for websites.
	JavaServer Faces and Future Trends in Programming	
		Understand the growth of information sharing on the web through technology.
		Recognize JavaServer Faces as an alternative method of web application development.
		Understand the overall business climate with respect to information technology to share information and data.
	Project: Create a Simple Java Server Page	
	Unit 5 Test	
COURSE PROJECT, REVIEW AND EXAM		
	Review	
	Exam	