

**CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS**

SUBJECT: IT / Java Programming
GRADE LEVEL: 9-12
COURSE TITLE: Foundations of Programming
COURSE CODE: 9007210
SUBMISSION TITLE: TeenCoder: Java Programming (Abridged)
BID ID: 2977
PUBLISHER: CompuScholar, Inc.
PUBLISHER ID: 27409378801

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include the student edition and teacher edition with the page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
<p>PUBLISHER'S NOTE AND INSTRUCTIONS: Student material (lesson text, activity instructions, etc) and Teacher material (teacher's guide, activity solutions, etc) are co-located in the same place within each online chapter. Student logins can see only the student material, while teacher logins (including review logins) can see both the teacher and student material. Therefore, a citation of "Chapter X, Lesson Y" is good for BOTH student material (lesson text, activity instructions) and teacher material (teacher's guide, activity solutions). Instructional videos are not specifically cited unless otherwise noted, though they also contain much of the same material found in the lesson text.</p> <p>The cited lessons demonstrate where concepts are first introduced, and in most cases those concepts are subsequently used throughout the course (e.g. CSS introduced in chapter 6, and used heavily thereafter).</p> <p>For a video walkthrough of the online system and review process, please visit our Florida landing page at: http://www.compuscholar.com/florida</p>		

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

Florida Standards		
01.0	Methods and strategies for using Florida Standards for grades 09-10 reading in Technical Subjects for student success in Java Development & Programming.	See below
01.01	Key Ideas and Details	See below
01.01.1 LAFS.910.RST.1.1	Cite specific textual evidence to support analysis of science and technical texts, attending to the precise details of explanations or descriptions.	Chapter 1, Lesson 4 (Read and discuss sample EULA) Chapter 3, Lesson 4 (Read and discuss Java API reference documents)
01.01.2 LAFS.910.RST.1.2	Determine the central ideas or conclusions of a text; trace the text’s explanation or depiction of a complex process, phenomenon, or concept; provide an accurate summary of the text.	Chapter 1, Lesson 4 (Read and discuss sample EULA) Chapter 3, Lesson 4 (Read and discuss Java API reference documents)
01.01.3 LAFS.910.RST.1.3	Follow precisely a complex multistep procedure when carrying out experiments, taking measurements, or performing technical tasks, attending to special cases or exceptions defined in the text.	Every chapter contains hands-on technical labs with multi-step instructions. Results are compared to activity requirements within the text.
01.02	Craft and Structure	See below
01.02.1 LAFS.910.RST.2.4	Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9–10 texts and topics.	Every chapter defines new keywords and Java symbols to be used in context of technical coding tasks.

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

01.02.2 LAFS.910.RST.2.5	Analyze the structure of the relationships among concepts in a text, including relationships among key terms (e.g., force, friction, reaction force, energy).	Chapter 1, Lesson 4 (Read and discuss sample EULA) Chapter 3, Lesson 4 (Read and discuss Java API reference documents)
01.02.3 LAFS.910.RST.2.6	Analyze the author's purpose in providing an explanation, describing a procedure, or discussing an experiment in a text, defining the question the author seeks to address.	Chapter 1, Lesson 4 (Read and discuss sample EULA) Chapter 3, Lesson 4 (Read and discuss Java API reference documents)
01.03	Integration of Knowledge and Ideas	See below
01.03.1 LAFS.910.RST.3.7	Translate quantitative or technical information expressed in words in a text into visual form (e.g., a table or chart) and translate information expressed visually or mathematically (e.g., in an equation) into words.	Chapter 17, Lesson 4 (Use flowcharts and text descriptions to describe common algorithms) Each lesson provides integrated multi-media presentations (video + text) explaining how to address a question or solve a problem. Students use videos and text together to understand technical concepts in both text and visual forms. See, for example: Chapter 19, Lesson 2 (Sorting Algorithms) with video and text explanations, plus a hands-on SortDemo program letting students verify sorting algorithms with real data.
01.03.2 LAFS.910.RST.3.8	Assess the extent to which the reasoning and evidence in a text support the author's claim or a recommendation for solving a scientific or technical problem.	Chapter 19, Lesson 2 (Students use the SortDemo program to verify the behavior of different sorting algorithms using varying input data sets)
01.03.3 LAFS.910.RST.3.9	Compare and contrast findings presented in a text to those from other sources (including their own experiments), noting when the findings support or contradict previous explanations or accounts.	Chapter 3, Lesson 4 (Understand multiple sources of help information to solve technical problems) Each lesson provides integrated multi-media presentations (video + text) explaining how to address a question or solve a problem. See, for example: Chapter 19, Lesson 2 (Sorting Algorithms) with video and text explanations, plus a hands-on SortDemo program letting students verify sorting algorithms with real data.
01.04	Range of Reading and Level of Text Complexity	See below

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

<p>01.04.1 01.04.2 LAFS.910.RST.4.10</p>	<p>By the end of grade 9, read and comprehend literature [informational texts, history/social studies texts, science/technical texts] in the grades 9–10 text complexity band proficiently, with scaffolding as needed at the high end of the range. By the end of grade 10, read and comprehend literature [informational texts, history/social studies texts, science/technical texts] at the high end of the grades 9–10 text complexity band independently and proficiently.</p>	<p>Chapter 1, Lesson 4 (Read and discuss sample EULA) Chapter 3, Lesson 4 (Read and discuss Java API reference documents)</p>
<p>02.0</p>	<p>Methods and strategies for using Florida Standards for grades 09-10 writing in Technical Subjects for student success in Java Development & Programming.</p>	<p>See below</p>
<p>02.01</p>	<p>Text Types and Purposes</p>	<p>See below</p>
<p>02.01.1 LAFS.910.WHST.1.1</p>	<p>Write arguments focused on discipline-specific content.</p>	<p>Most chapter activities involve writing or formatting of code within the formal Java syntax and best coding practices. Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)</p>

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

02.01.2 LAFS.910.WHST.1.2	Write informative/explanatory texts, including the narration of historical events, scientific procedures/experiments, or technical processes.	Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)
02.02	Production and Distribution of Writing	See below
02.02.1 LAFS.910.WHST.2.4	Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.	Most chapter activities involve writing or formatting of code within the formal Java syntax and best coding practices. Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)
02.02.2 LAFS.910.WHST.2.5	Develop and strengthen writing as needed by planning, revising, editing, rewriting, or trying a new approach, focusing on addressing what is most significant for a specific purpose and audience.	Most chapter activities involve writing or formatting of code within the formal Java syntax and best coding practices. Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)
02.02.3 LAFS.910.WHST.2.6	Use technology, including the Internet, to produce, publish, and update individual or shared writing products, taking advantage of technology's capacity to link to other information and to display information flexibly and dynamically.	Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)
02.03	Research to Build and Present Knowledge	See below

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

02.03.1 LAFS.910.WHST.3.7	Conduct short as well as more sustained research projects to answer a question (including a self-generated question) or solve a problem; narrow or broaden the inquiry when appropriate; synthesize multiple sources on the subject, demonstrating understanding of the subject under investigation.	Chapter 21 (Team project involves successive class presentations regarding project requirements, design, implementation, and test results using any desired documentation or media) Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise) Supplemental Lesson 4 / Activity 4 (Software Design Process and SDLC Docs)
02.03.2 LAFS.910.WHST.3.8	Gather relevant information from multiple authoritative print and digital sources, using advanced searches effectively; assess the usefulness of each source in answering the research question; integrate information into the text selectively to maintain the flow of ideas, avoiding plagiarism and following a standard format for citation.	Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise)
02.03.3 LAFS.910.WHST.3.9	Draw evidence from informational texts to support analysis, reflection, and research.	Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise)
02.04	Range of Writing	See below
02.04.1 LAFS.910.WHST.4.10	Write routinely over extended time frames (time for reflection and revision) and shorter time frames (a single sitting or a day or two) for a range of discipline-specific tasks, purposes, and audiences.	Students engage in short, hands-on labs in every chapter to write code. They have the opportunity to write longer code with iterative improvements in Chapter 16 (Jail Break Project) and larger projects with supporting technical documentation in Chapter 21 (Team Project and report).

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

03.0	Methods and strategies for using Florida Standards for grades 09-10 Mathematical Practices in Technical Subjects for student success in Java Development & Programming].	See below
03.01 MAFS.K12.MP.1.1	Make sense of problems and persevere in solving them.	Chapter 14 Activity (Define and solve management of numbers to calculate baseball statistics) Chapter 17, Lesson 4 / Activity 2 (Use flowcharts and text descriptions to describe common mathematical problems and solve those algorithms in code.) Chapter 19, Lesson 2 (Understand numeric sorting problems and how multiple algorithms can solve those requirements.)
03.02 MAFS.K12.MP.2.1	Reason abstractly and quantitatively.	Chapter 14 Activity (Define and solve management of numbers to calculate baseball statistics) Chapter 17, Lesson 4 / Activity 2 (Use flowcharts and text descriptions to describe common mathematical problems and solve those algorithms in code.) Chapter 19, Lesson 2 (Understand numeric sorting problems and how multiple algorithms can solve those requirements.)
03.03 MAFS.K12.MP.3.1	Construct viable arguments and critique the reasoning of others.	Supplemental Lesson 2 / Activity 2 (Technical Writing Exercise)
03.04 MAFS.K12.MP.4.1	Model with mathematics.	Chapter 7, Lesson 1 (Boolean Algebra, logical expressions, mathematical expressions) Chapter 14 Activity (Define and solve management of numbers to calculate baseball statistics) Chapter 17, Lesson 2 (Understand decimal, binary, octal, and hexadecimal numbering systems, how to count in each system, and how to convert values between each representation) Chapter 17, Lesson 4 / Activity 2 (Use flowcharts and text descriptions to describe common mathematical problems and solve those algorithms in code.)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

03.05 MAFS.K12.MP.5.1	Use appropriate tools strategically.	<p>Chapter 2, All Lessons and Activity (Use the JDK and command-line tools to build and execute a Java program)</p> <p>Chapter 3, All Lessons and Activity (Use the Eclipse IDE to write code, build, and run programs)</p> <p>Chapter 9, Lesson 4 / Activity (Use the Eclipse debugger to identify and solve coding errors)</p> <p>Chapter 17, Lesson 4 / Activity 2 (Use flowcharts and text descriptions to describe common mathematical problems and solve those algorithms in code.)</p> <p>Supplemental Lesson 5 / Activity 5 (Use UML for program design)</p> <p>Supplemental Lesson 6 / Activity 6 (Productivity Tools)</p>
03.06 MAFS.K12.MP.6.1	Attend to precision.	<p>Chapter 4, Lesson 1 (Numeric ranges and limitations of data types)</p> <p>Chapter 14 Activity (Accurately calculate statistics from source data)</p> <p>Chapter 17, Lesson 2 (Understand finite integer bounds, limitations of floating point precision, and representation of fractions in binary)</p> <p>Chapter 17, Lesson 3 / Activity 1 (Use Java Math library functions in successful calculations)</p>
03.07 MAFS.K12.MP.7.1	Look for and make use of structure.	<p>Chapter 7, All Lessons and Activity (Identify and apply logical expressions for flow control structures)</p> <p>Chapters 10 and 11, All Lessons and Activities (Use object-oriented concepts to model real-world structures in code and make use of objects in a program)</p> <p>Chapter 14, All Lessons and Activity (Identify and use array and linked list data structures appropriately)</p> <p>Chapter 15 and 16, All Lessons and Activities (Identify and use object inheritance and polymorphism to model real-world structures appropriately)</p>

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

03.08 MAFS.K12.MP.8.1	Look for and express regularity in repeated reasoning.	<p>Chapter 7, All Lessons and Activitiy (Identify and apply logical expressions to flow control structures, including loops)</p> <p>Chapter 17, Lesson 2 (Identify and use binary, octal, and hexadecimal numbering systems to systematically represent, count and convert values)</p> <p>Chapter 17, Lesson 4 and Activity 2 (Identify algorithm requirements and implement re-usable functions to solve specific problems)</p> <p>Chapter 19, Lessons 2 and 3 (Understand and use multiple algorithms to solve sorting and searching problems, including iterative and recursive approaches)</p>
CTE Standards and Benchmarks		
31.0	Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:	See below
31.01 MAFS.912.A-REI.1.1	Describe the evolution of programming and programming careers.	<p>Chapter 1, Lesson 3 (Common Programming Languages)</p> <p>Chapter 2, Lesson 1 (The Java Platform)</p> <p>Supplemental Lesson 1 (Software Development Careers)</p>
31.02 MAFS.912.N-Q.1.1	Identify tasks performed by programmers.	<p>Chapter 2, Lesson 1 (The Java Platform) - code/compile/run sequence</p> <p>Chapter 2, Lesson 2 (Writing Your First Program) - write code</p> <p>Chapter 2, Lesson 3 (Building and Running from the Command Line) - build and run code</p> <p>Chapter 9, Lesson 3 (Finding Runtime Errors) - testing and debugging</p> <p>Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - requirements, design, implementation, testing</p> <p>Supplemental Lesson 1 (Software Development Careers) - description of job roles</p> <p>Supplemental Lesson 4 (Software Development Process) - steps in the software lifecycle</p>
31.03 MAFS.912.A-REI.1.1	Describe how businesses use computer programming to solve business problems.	Chapter 1, Lesson 3 (Common Programming Languages)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

31.04	Investigate job opportunities in the programming field.	Supplemental Lesson 1 (Software Development Careers) Supplemental Activity 1 (Exploring Software Careers)
31.05 MAFS.912.A-REI.1.1 MAFS.912.G- SRT.1.2	Explain different specializations and the related training in the computer programming field.	Supplemental Lesson 1 (Software Development Careers) Supplemental Activity 1 (Exploring Software Careers)
31.06 MAFS.912.A-REI.1.1	Explain the need for continuing education and training of computer programmers.	Supplemental Lesson 1 (Software Development Careers) Supplemental Activity 1 (Exploring Software Careers)
31.07 MAFS.912.A-REI.1.1	Explain enterprise software systems and how they impact business.	Supplemental Lesson 6 / Activity 6 (Productivity Tools)
31.08 MAFS.912.A-REI.1.1	Describe ethical responsibilities of computer programmers.	Chapter 1, Lesson 4 (Computer Ethics and Security)
31.09 MAFS.912.A-REI.1.1	Describe the role of customer support to software program quality.	Supplemental Lesson 1 (Software Development Careers)
31.10 MAFS.912.N-Q.1.1	Identify credentials and certifications that may improve employability for a computer programmer.	Supplemental Lesson 1 (Software Development Careers) Supplemental Activity 1 (Exploring Software Careers)
31.11 MAFS.912.G- CO.4.12; MAFS.912.N-Q.1.1	Identify devices, tools, and other environments for which programmers may develop software.	Chapter 1, Lesson 1 (A Survey of Computer Hardware) Chapter 1, Lesson 3 (Common Programming Languages) Chapter 2, Lesson 1 (The Java Platform) Chapter 3, Lesson 1 (Introducing Eclipse) Chapter 9, Lesson 4 (The Eclipse Debugger)
32.0	Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:	See below

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

32.01 MAFS.912.N-Q.1.2	Identify the characteristics (e.g., size, limits) and uses of different numerical and non- numerical data types.	Chapter 4, Lesson 1 (Primitive Data Types) Chapter 4, Lesson 2 (Variables) Chapter 5, Lesson 1 (Reference Data Types) Chapter 17, Lesson 2 (The Binary Number System)
32.02 MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1	Explain the types and uses of variables in programs.	Chapter 4, Lesson 1 (Primitive Data Types) Chapter 4, Lesson 2 (Variables) Chapter 5, Lesson 1 (Reference Data Types) (Plus re-enforced in all hands-on programming projects)
32.03 MAFS.912.A-REI.1.1	Determine the best data type to use for given programming problems.	Chapter 4, Lesson 1 (Primitive Data Types) Chapter 4, Lesson 2 (Variables) Chapter 5, Lesson 1 (Reference Data Types) Chapter 10, Lesson 2 (Defining a Class) (Plus re-enforced in all hands-on programming projects)
32.04 MAFS.912.N-Q.1.1	Identify the types of operations that can be performed on different data types.	Chapter 4, Lesson 2 (Variables) Chapter 4, Lesson 3 (Printing Data) Chapter 5, Lesson 2 (Comparing Strings) Chapter 5, Lesson 3 (Common String Operations) Chapter 5, Lesson 5 (Converting Between Strings and Numbers)
32.05 MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1	Evaluate arithmetic and logical expressions using appropriate operator precedence.	Chapter 4, Lesson 2 (Variables) Chapter 7, Lesson 1 (Logical Expressions and Relational Operators)
32.06 MAFS.912.A-REI.1.1	Explain how computers store different data types in memory.	Chapter 4, Lesson 1 (Primitive Data Types) Chapter 5, Lesson 1 (Reference Data Types) Chapter 17, Lesson 2 (The Binary Number System)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

32.07 MAFS.912.N-Q.1.1	Use different number systems to represent data.	Chapter 17, Lesson 2 (The Binary Number System)
32.08 MAFS.912.A-REI.1.1	Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	Chapter 5, Lesson 2 (Comparing Strings)
32.09	Use Boolean logic to perform logical operations.	Chapter 7, Lesson 1 (Logical Expressions and Relational Operators)
33.0	Distinguish between iterative and non-iterative program control structures. – The student will be able to:	See below
33.01 MAFS.912.A-REI.1.1	Explain non-iterative programming structures (e.g., if, if/else) and their uses.	Chapter 7, Lesson 2 (Using the "if" Statement) Chapter 7, Lesson 3 (The "switch" Statement)
33.02 MAFS.912.A-REI.1.1	Explain iterative programming structures (e.g., while, do/while) and their uses.	Chapter 7, Lesson 4 (For Loops) Chapter 7, Lesson 5 (While Loops) Chapter 14, Lesson 3 (Iterators)
34.0	Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:	See below
34.01 MAFS.912.N-Q.1.1	Identify the characteristics, uses, and limits of low-level programming languages.	Chapter 1, Lesson 3 (Common Programming Languages)
34.02 MAFS.912.N-Q.1.1	Identify the characteristics, uses, and limits of high-level programming languages.	Chapter 1, Lesson 3 (Common Programming Languages)
34.03 MAFS.912.N-Q.1.1	Identify the characteristics, uses, and limits of rapid development programming languages.	Chapter 1, Lesson 3 (Common Programming Languages)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

34.04 MAFS.912.A-REI.1.1	Describe object-oriented concepts.	Chapter 10, Lesson 1 (Object-Oriented Concepts) Chapter 10, Lesson 2 (Defining a Class) Chapter 10, Lesson 3 (Public, Private, and Protected Classes) (Plus elaboration of specific concepts throughout Chapters 11 and 15)
34.05 MAFS.912.A-REI.1.1	Explain the characteristics of procedural and object-oriented programming languages.	Chapter 10, Lesson 1 (Object-Oriented Concepts)
34.06 MAFS.912.G- SRT.1.2	Compare and contrast programming languages that are compiled (e.g., C++), interpreted (e.g., Basic, HTML), and translated (e.g., Java).	Chapter 1, Lesson 3 (Common Programming Languages) Chapter 2, Lesson 1 (The Java Platform)
35.0	Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:	See below
35.01 MAFS.912.A-REI.1.1; MAFS.912.G- CO.4.12	Describe and explain tools used in software development.	Chapter 2, Lesson 1 (The Java Platform) Chapter 3, Lesson 1 (Introducing Eclipse) Chapter 9, Lesson 4 (The Eclipse Debugger) Supplemental Lesson 4 (Software Development Process)
35.02 MAFS.912.A-REI.1.1	Describe the stages of the program life cycle.	Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - requirements, design, implementation, testing Supplemental Lesson 4 (Software Development Process)
35.03 MAFS.912.G- SRT.1.2	Compare and contrast alternative methods of program development (e.g., rapid prototyping, waterfall).	Supplemental Lesson 4 (Software Development Process)
35.04 MAFS.912.A-REI.1.1	List and explain the steps in the program development cycle.	Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - requirements, design, implementation, testing Supplemental Lesson 4 (Software Development Process)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

35.05 MAFS.912.N-Q.1.1	Describe different types of documentation used in the program development cycle (e.g., requirements document, program design documents, test plans).	Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - requirements, design, implementation, testing Supplemental Lesson 4 (Software Development Process)
35.06 MAFS.912.A-REI.1.1	Describe the on-going need for program maintenance.	Supplemental Lesson 4 (Software Development Process)
35.07 MAFS.912.A-REI.1.1; MAFS.912.G- SRT.1.2	Describe different methods companies use to facilitate program updates for enhancements and defects (e.g., how customers receive patches, updates, new versions, upgrades).	Supplemental Lesson 4 (Software Development Process)
35.08 MAFS.912.A-REI.1.1; MAFS.912.G- SRT.1.2	Describe different methods used to facilitate version control and change management.	Supplemental Lesson 4 (Software Development Process)
36.0	Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:	See below
36.01 MAFS.912.A-REI.1.1; SC.912.N.1.1	Explain the uses and limits of testing in ensuring program quality.	Chapter 9, Lesson 3 (Finding Runtime Errors) Supplemental Lesson 4 (Software Development Process)
36.02 MAFS.912.A-REI.1.1; MAFS.912.A- CED.1.1	Explain testing performed at different stages of the program development cycle (e.g. unit testing, system testing, user acceptance testing).	Supplemental Lesson 4 (Software Development Process)
36.03 MAFS.912.A-REI.1.1; SC.912.N.1.1	Describe data and the use of test plans/scripts to be used in program testing.	Chapter 9, Lesson 3 (Finding Runtime Errors) Supplemental Lesson 4 (Software Development Process)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

36.04 MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1	Describe and identify types of programming errors (e.g., syntactical, logic, usability, requirements mismatch).	Chapter 9, Lesson 1 (Logic Errors, Runtime Errors and Exceptions)
36.05 MAFS.912.N-Q.1.1	Identify the data to be used for boundary conditions.	Chapter 9, Lesson 3 (Finding Runtime Errors)
36.06 MAFS.912.A-REI.1.1; MAFS.912.G- CO.4.12; SC.912.N.1.1	Explain different types of testing (e.g., usability, automated, regression) and testing tools.	Supplemental Lesson 4 (Software Development Process)
37.0	Create a program design document using Unified Modeling Language (UML) or other common design tool. – The student will be able to:	See below
37.01 MAFS.912.A-REI.1.1; SC.912.N.1.1, SC.912.N.3.5	Describe different design methodologies and their uses (e.g., object-oriented design, structured design, and rapid application development).	Supplemental Lesson 5 (Exploring UML) Supplemental Activity 5 (UML Design)
37.02 MAFS.912.A-REI.1.1; SC.912.N.1.1	Describe tools for developing a program design (e.g., UML, flowcharts, design documents, pseudocode).	Chapter 17, Lesson 4 (Use flowcharts to describe and implement common algorithms) Supplemental Lesson 5 (Exploring UML) Supplemental Activity 5 (UML Design)
37.03 MAFS.912.A-REI.1.1	Explain the role of existing libraries and packages in facilitating programmer productivity.	Chapter 2, Lesson 4 (Java Classes and Packages) Chapter 12, Lesson 1 (Java Swing) Chapter 17, Lesson 1 (Java Math Functions)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

37.04 MAFS.912.A- CED.1.1; SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	Participate and contribute to a design review of a program design developed using a common program design tool (e.g., UML, flowcharts, design documents, pseudocode).	Supplemental Lesson 5 (Exploring UML) Supplemental Activity 5 (UML Design)
37.05 MAFS.912.A- CED.1.1	Write a program design document using UML or other standard design methodology.	Supplemental Lesson 5 (Exploring UML) Supplemental Activity 5 (UML Design)
37.06 MAFS.912.F-IF.1.1	Define input and output for a program module using UML or other standard design methodology.	Supplemental Lesson 5 (Exploring UML) Supplemental Activity 5 (UML Design)
38.0	Describe information security risks, threats, and strategies associated with software development. – The student will be able to:	See below
38.01 MAFS.912.S-IC.2.6	Explain the security risks to personal and business computer users.	Chapter 1, Lesson 4 (Computer Ethics and Security)
38.02 MAFS.912.N-Q.1.1	Identify different types of threats to computer systems.	Chapter 1, Lesson 4 (Computer Ethics and Security)
38.03 MAFS.912.N-Q.1.1	Identify methods to protect against different threats to computer systems.	Chapter 1, Lesson 4 (Computer Ethics and Security)
38.04	Understand the importance of a disaster / emergency response plan.	Chapter 1, Lesson 4 (Computer Ethics and Security)
38.05 MAFS.912.N-Q.1.1	Identify alternative methods for data storage and backup (e.g., mirroring, fail-over, high availability, types of backups).	Chapter 1, Lesson 4 (Computer Ethics and Security)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

39.0	Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:	See below
39.01 SC.912.N.1.9, SC.912.N.1.10	Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.	Chapter 21, Lessons & Activities 1, 2, 3 (Team Project) - team presentations at the end of each phase use a variety of documents and techniques Chapter 21, Lesson 4 / Activity 4 (Project Testing) - use of a test plan to facilitate testing Supplemental Lesson 2 / Activity 2 (Technical Writing) Supplemental Lesson 4 / Activity 4 (Software Development Process) - create common SDLC documents to complement the planning, requirements, design, and testing phases
39.02 SC.912.N.1.1.6	Locate, organize and reference written information from various sources.	Chapter 3, Lesson 4 (Help and Reference Documentation) Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - team projects reference self-generated requirements, design, and testing documents Supplemental Lesson 2 / Activity 2 (Technical Writing)
39.03 MAFS.912.A- CED.1.1; SC.912.N.1.1.9, SC.912.N.1.1.10	Design, develop and deliver formal and informal presentations using appropriate media to engage and inform diverse audiences.	Chapter 21, Lessons & Activities 1, 2, 3, 4 (Team Project) - team presentations at the end of each phase use a variety of documents and techniques
39.04 MAFS.912.G- SRT.1.2; SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	Interpret verbal and nonverbal cues/behaviors that enhance communication.	Chapter 21, Lesson 1 (Project Requirements) - working as a team Supplemental Lesson 1 (Software Development Careers) - identifying talents Plus, all chapter lessons contain instructional videos that allow students to interpret verbal lectures while watching nonverbal (visual) information to explain and re-enforce lesson concepts.
39.05	Apply active listening skills to obtain and clarify information.	All chapter lessons contain instructional videos that allow students to listen to lectures to understand and clarify key lesson concepts.

**CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS**

39.06 MAFS.912.A-REI.1.1; MAFS.912.A- CED.1.1 MAFS.912.F-IF.3.9; SC.912.N.1.1.6-11	Develop and interpret tables and charts to support written and oral communications.	Supplemental Lesson 2 / Activity 2 (Technical Writing)
39.07	Exhibit public relations skills that aid in achieving customer satisfaction.	Chapter 21, All Activities (Students present progress at each team project stage to the class and solicit feedback for improvement) Supplemental Lesson 1 (Software Development Careers - customer support role)
40.0	Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:	See below
40.01 MAFS.912.G-CO.3.9; SC.912.N.1.1	Employ critical thinking skills independently and in teams to solve problems and make decisions.	All chapters contain hands-on programming activities allowing students to independently make decisions and solve coding problems. Chapter 21 also contains a team-driven project allowing students to creatively use their skills in planning, developing, and testing a unique project.
40.02 MAFS.912.G-CO.3.9; SC.912.N.1.3, SC.912.N.4.1	Employ critical thinking and interpersonal skills to resolve conflicts.	Chapter 21 (Team Project) - students must work together as a team, resolving conflicts as needed using interpersonal and thinking skills.
40.03 MAFS.912.N-Q.1.1	Identify and document workplace performance goals and monitor progress toward those goals.	Chapter 21 (Team Project) - Students will identify, document, and present team goals, designs, and output through progressive stages of the software development lifecycle. Supplemental Lesson 1 / Activity 1 (Software Development Careers, including the talents, traits, education, and certification necessary to be successful) Supplemental Lesson 4 / Activity 4 (Software Development Process, including planning, documentation, and management approaches)

CORRELATION
FLORIDA DEPARTMENT OF EDUCATION
INSTRUCTIONAL MATERIALS CORRELATION
COURSE STANDARDS/BENCHMARKS

40.04 MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1; SC.912.N.1.3, SC.912.N.1.1.5	Conduct technical research to gather information necessary for decision-making.	Chapter 3, Lesson 4 (Research and explore online technical documentation to aid in coding efforts) Chapter 19, Lesson 2 (Use the SortDemo program to explore sorting algorithms and verify behavior) Supplemental Lesson 1 / Activity 1 (Research available job listings as part of career exploration) Supplemental Lesson 2 / Activity 2 (Technical Writing)
41.0	Use information technology tools. – The student will be able to:	See below
41.01	Use personal information management (PIM) applications to increase workplace efficiency.	Supplemental Lesson 6 / Activity 6 (Productivity Tools)
41.02 MAFS.912.G- CO.4.12	Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	Supplemental Lesson 6 / Activity 6 (Productivity Tools)
41.03 MAFS.912.Z- CED.1.1	Employ computer operations applications to access, create, manage, integrate, and store information.	Supplemental Lesson 6 / Activity 6 (Productivity Tools)
41.04	Employ collaborative/groupware applications to facilitate group work.	Supplemental Lesson 6 / Activity 6 (Productivity Tools)