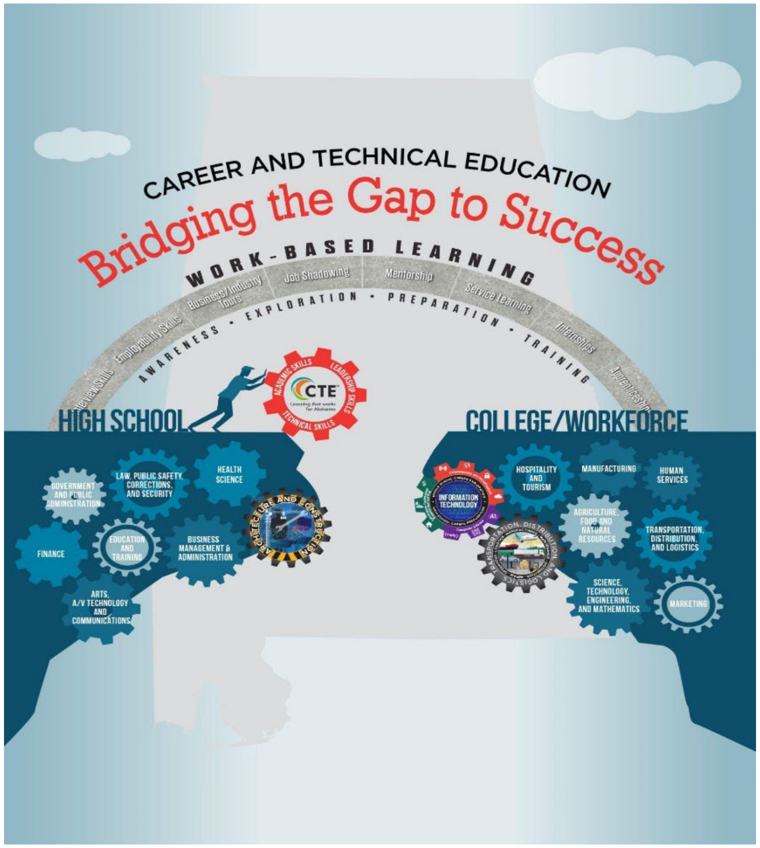
Information Technology Rubric

Publisher Name: CompuScholar, Inc.


CAREER AND TECHNICAL EDUCATION
Bridging the Gap to Success
WORK-BASED LEARNING
Interview Skills
Employability Skills
Business/Industry Tours
Job Shadowing
Mentorship
Service Learning
Internships
Apprenticeships
AWARENESS • EXPLORATION • PREPARATION • TRAINING
ACADEMIC SKILLS
LEADERSHIP SKILLS
TECHNICAL SKILLS
CTE
Learning that works for Alabama
HIGH SCHOOL
COLLEGE/WORKFORCE
GOVERNMENT AND PUBLIC ADMINISTRATION
LAW, PUBLIC SAFETY, CORRECTIONS, AND SECURITY
HEALTH SCIENCE
FINANCE
EDUCATION AND TRAINING
BUSINESS MANAGEMENT & ADMINISTRATION
ARCHITECTURE AND CONSTRUCTION
ARTS, A/V TECHNOLOGY AND COMMUNICATIONS
INFORMATION TECHNOLOGY
HOSPITALITY AND TOURISM
MANUFACTURING
HUMAN SERVICES
AGRICULTURE, FOOD AND NATURAL RESOURCES
TRANSPORTATION, DISTRIBUTION, AND LOGISTICS
SCIENCE, TECHNOLOGY, ENGINEERING, AND MATHEMATICS
MARKETING

2022

Eric G. Mackey, State Superintendent of
Education Alabama State Department of
Education

Publisher Name: CompuScholar, Inc “Java Programming” Course

CompuScholar Course Details:

Course Title:	Java Programming
Course ISBN:	978-1-946113-99-3
Course Year:	2022

Publisher Note 1: Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts and can be ignored for alignment purposes.

Publisher Note 2: Citation(s) to "Supplemental" or "Suppl." chapters refer to Supplemental Chapters found at the end of the main sequence of numbered chapters within the course.

OBJECT-ORIENTED PROGRAMMING II
CONTENT STANDARDS

Each content standard completes the stem *"**Students will**…"*

Topic	Content Standard	Location of Standard in Resource (chapters, sections, pages, etc.)
Ethical Practices	1. Identify potential abuses and unethical uses of computers and/or networks. 2. Research and discuss legal issues and the terms of use related to copyright laws, fair use laws, and ethics pertaining to the downloading and personal use of elements including images, photographs, documents, video, sounds, music, and trademarks. 3. Describe ethical and legal practices for securing the confidentiality of business-related information.	1. Chapter 1, Lessons 4, 5 Text 2. Chapter 1, Lessons 4, 5 Text 3. Chapter 1, Lessons 4, 5 Text
Object-Oriented Program Design	4. Create an effective graphical representation of logic using Unified Modeling Language (UML) diagrams. 5. Explain the advanced functions of the integrated development environment (IDE). *Examples: interpreting IDE feedback and errors, debugging the code for all type errors* 6. Explain the higher-order functions of object-oriented programming, including encapsulation, abstraction, inheritance, polymorphism, composition, and instantiation.	4. Suppl. Chapter 1, Lesson 6 Text 5. Chapter 29, Lessons 2, 3, 4 Text 6. Chapter 15, Lesson 3 Text (Encapsulation) Chapters 22, 23 (abstraction, inheritance, polymorphism) Chapter 14, Lesson 1 Text (composition) Chapter 26, Lesson 2 Text (composition) Chapter 5, Lessons 1, 2 Text (instantiation)

	7. Describe a program's general use and purpose in the program documentation so the end user can execute the product successfully. 8. Identify all objects that need to be called and their functions in a computer program. 9. Describe the functions of different objects and their purposes in a program.	7. N/A (focus on internal documentation, not end-user docs, see below) 8. Chapter 27, Lesson 2 Text Suppl. Chapter 1, Lesson 2 Text Suppl. Chapter 2, Lesson 2 Text 9. Chapter 27, Lesson 2 Text Suppl. Chapter 1, Lesson 2 Text Suppl. Chapter 2, Lesson 2 Text
Object-Oriented Data Manipulation	10. Describe the purpose and function of various types of arrays. *Examples: single, multidimensional* 11. Construct and evaluate arithmetic expressions in a program or class. *Examples: arithmetic operators, assignment, compound assignment operators, operator precedence* 12. Explain the purpose and function of a data structure as it relates to object-oriented programming. *Examples: array, linked list, stack, queue, binary tree, binary search tree, heap, hashing, graph* 13. Construct and evaluate code that creates, iterates, and manipulates arrays and array lists. *Examples: one- and two-dimensional arrays, including initialization, null, size, iterating elements, accessing elements; adding and removing array list elements, traversing the list* 14. Construct and evaluate code that performs parsing, casting, and conversion.	10. Chapters 18, 21 11. Chapter 4, Lessons 1, 2 Text 12. Chapters 18, 19, 21 Suppl. Chapter 1, Lesson 5 Text 13. Chapters 18, 19, 21 14. Chapter 4, Lesson 3 Text (casting) Chapter 7, Lesson 1 Text (parsing & conversion)

	Examples: cast between primitive data types, convert primitive types to equivalent object types, parse strings to numbers, convert primitive data types to strings 15. Create an external file for data storage and manipulation through a program. 16. Differentiate among types of sorting algorithms. *Examples: linear, bubble, selection, insert, binary*	15. Chapter 25 16. Chapter 20, Lessons 1, 2, 3 Text Chapter 24, Lesson 3 Text
Object-Oriented Practical Programming	17. Construct a program that uses appropriate sorting algorithms. *Examples: binary sort, bubble sort, merge sort, selection sort* 18. Write a program using advanced programming features. *Examples: multiple windows, splash screens, menus, dialogs* 19. Write a program that integrates multiple external applications, including spreadsheets, databases, and word processing documents. 20. Create an advanced macro for applications software. 21. Describe the purpose and function of web controls. 22. Create a web application that includes input validation. 23. Create an interactive program which gathers input from the user and provides appropriate output and feedback based on the user's input.	17. Chapter 20, Lessons 1, 2, 3 Text Chapter 24, Lesson 3 Text 18. Chapter 24, Lesson 1 Text (recursion) Chapters 30, 31 (GUI programs) 19. N/A – not (see Digital Savvy course) 20. N/A 21. N/A (see Web Design course) 22. N/A (see Web Design course) 23. Most programs and examples from Chapter 5 onwards (e.g. Chapter 5 Activity) include interactive user input and output.

	24. Describe decision structures and how they control the flow of a program.	24. Chapter 8, 9
	25. Describe the conversion to and from ASCII and Unicode using hexadecimal and binary number systems.	25. Chapter 7, Lesson 2 Text Suppl. Chapter 1, Lesson 1 Text
	26. Declare, implement, and access data members in classes. *Examples: private, public, protected; instance data members; static data members; use static final to create constants; describe encapsulation*	26. Chapter 14, Lesson 3 Text Chapter 15, Lesson 3 Text Chapter 16, Lesson 1 Text
	27. Instantiate and use class objects in programs. *Examples: initialization, null, access and modify data members, access methods, access and modify static members, import packages and classes*	27. Chapter 5, Lessons 1, 2, 3 Text Chapter 7, Lesson 3 Text Chapter 14, Lesson 2 Text Chapters 15, 16, 22, 23
	28. Debug a program for errors. *Examples: run-time, exception, logic, semantic, try/catch/finally, exception class, exception class types, display exception information*	28. Chapter 10, Lessons 1, 2 Text Chapter 11, Lessons 1, 2 Text