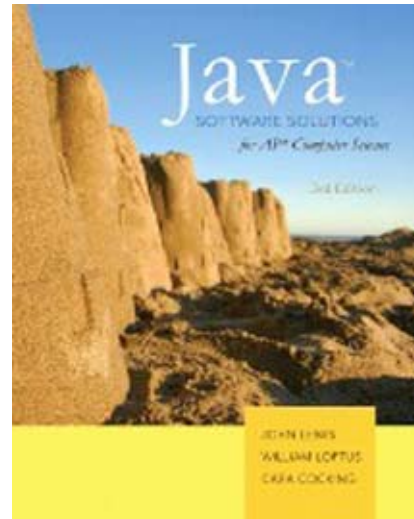


A Correlation of  
**Java Software Solutions for  
AP Computer Science**  
©2011



To the  
**Florida Department of Education  
Instructional Materials Correlation  
Java Programming Essentials  
Course 9007240**

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

**SUBJECT: Career and Technical Education  
GRADE LEVEL: 9-12  
COURSE TITLE: Java Programming Essentials  
COURSE CODE: 9007240  
SUBMISSION TITLE: Java Software Solutions for AP Computer Science  
BID ID: 2973  
PUBLISHER: Savvas Learning Company  
PUBLISHER ID: 22-1603684-03**

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
55.0	Construct statements that declare, initialize, and modify different types of variables used in Java programs. – The student will be able to:	
55.01	Describe how variables are used in programs.	<b>SE:</b> 60-65, 143
55.02	Identify the eight Java primitive data types.	<b>SE:</b> 65-68, 106
55.03	Identify the minimum and maximum ranges of primitive data types.	<b>SE:</b> 66, 67, 68
55.04	Identify which data type should be used for a given situation.	<b>SE:</b> 65-68
55.05	Identify the syntax for using variables.	<b>SE:</b> 36, 44, 60-68, 143, 187-189
55.06	Declare and initialize variables.	<b>SE:</b> 61- 65

Copyright © 2020 Savvas Learning Company LLC All Rights Reserved.  
**Savvas™** and **Savvas Learning Company™** are the exclusive trademarks of Savvas Learning Company LLC in the US and in other countries.

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
55.07	Assign new values to variables.	<b>SE:</b> 63-65, 135-136
55.08	Create and use constant variables.	<b>SE:</b> 62, 65, 112
56.0 Describe the types and characteristics of lexical units in the Java programming language. – The student will be able to:		
56.01	Describe the types of lexical units.	<b>SE:</b> 125-128
56.02	Describe identifiers and identify valid and invalid identifiers.	<b>SE:</b> 27-30
56.03	Describe and identify reserved words, delimiters, literals, and comments.	<b>SE:</b> 23-25, 27-30, 57, 67-68, 89-92
57.0 Describe the data types employed in Java programs. – The student will be able to:		
57.01	Describe the data type categories.	<b>SE:</b> 65-68
57.02	Give examples of primitives, reference data types.	<b>SE:</b> 65-68, 74-77, 106, 242-249
57.03	Identify and use enumerations.	<b>SE:</b> 73-74, 106, 111
57.04	Understand the use of Wrapper Classes in programs.	<b>SE:</b> 79-81
57.05	Describe the difference between real and integer data types.	<b>SE:</b> 66-68, 72-73

BENCHMARK CODE	BENCHMARK	<b>LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL</b> (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
58.0	Construct Java statements that employ the use of various operators. – The student will be able to:	
58.01	Construct statements using arithmetic operators.	<b>SE:</b> 68-71, 107-111, 112, 113, 175
58.02	Construct statements using relational operators.	<b>SE:</b> 123-125, 175, 177
58.03	Construct and use statements using logical operators.	<b>SE:</b> 130-132, 175, 177, 321
58.04	Construct and use statements using assignment operators.	<b>SE:</b> 63-65, 69-71, 72-73, 112, 135-136, 176
58.05	Construct and execute statements using operator precedence.	<b>SE:</b> 69-71, 112, 113
59.0	Write executable statements using Java. – The student will be able to:	
59.01	Construct variable assignment statements.	<b>SE:</b> 63-65, 69-71, 72-73, 112, 113, 117
59.02	Construct statements using built-in Math functions.	<b>SE:</b> 86-89, 112-113, 177, 186, 232, 253-254
59.03	Differentiate between implicit and explicit data type conversions.	<b>SE:</b> 72-73
59.04	Describe when implicit data type conversions take place.	<b>SE:</b> 72-73
59.05	List the drawbacks of implicit data type conversions.	<b>SE:</b> 72-73

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
59.06	Describe the process of autoboxing and promotion.	<b>SE:</b> 81-82, 105
59.07	Construct statements using functions to explicitly convert data types.	<b>SE:</b> 73, 112, 113, 176, 177
60.0 Describe variable scope and its implications in Java programming. – The student will be able to:		
60.01	Understand the scope and visibility of variables.	<b>SE:</b> 187, 191, 202
60.02	Write programs using local variables.	<b>SE:</b> 61-64, 177, 432
60.03	Describe the scope of a variable.	<b>SE:</b> 187-189, 226
60.04	Describe the default value of local, instance, and static scope of variables.	<b>SE:</b> 253-256
60.05	Describe how compiler uses scope to identify variables with the same name.	<b>SE:</b> 202
61.0 Apply common Java programming style guidelines and conventions. – The student will be able to:		
61.01	List examples of good programming practices.	<b>SE:</b> 30-31, 127, 191, 227
61.02	Insert comments into code.	<b>SE:</b> 23-27
61.03	Follow formatting guidelines when writing code.	<b>SE:</b> 30-31

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
61.04	Understand the different types of errors produced by programs.	<b>SE:</b> 37, 118-119, 256-259, 271-273
62.0 Demonstrate use of the compiler and interpreter through command line interface. – The student will be able to:		
62.01	Describe the use of the Java compiler (javac) and Java interpreter (Java VM).	<b>SE:</b> 34-36
62.02	Demonstrate the use of the -classpath flag and -d flag to the compiler.	Pages 34-36 may be used in preparation to meet this benchmark.
62.03	Identify the environmental variables of PATH and CLASSPATH.	5.3 Interfaces, 259-267, may be used in preparation to meet this benchmark.
62.04	Describe the process of command line arguments to the program.	<b>SE:</b> 307-309
62.05	Create programs that take in multiple command line arguments.	<b>SE:</b> 307-309
63.0 Construct conditional control statements in Java. – The student will be able to:		
63.01	Construct and use an if statement.	<b>SE:</b> 121-122, 124-125, 175
63.02	Construct and use a switch statement.	Students could use a switch statement in the following projects; 113, 175
63.03	Construct and use a while, do while, and for loop.	<b>SE:</b> 136-140, 148-149, 150-157, 175, 176, 177, 232
63.04	Construct and use a conditional operator.	<b>SE:</b> 121-122, 124-125, 175

BENCHMARK CODE	BENCHMARK	<b>LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL</b> (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
64.0 Construct iterative control statements in Java. – The student will be able to:		
64.01	Describe the types of loop statements and their uses.	<b>SE:</b> 120-121, 156-157
64.02	Construct and use the while and do while loop.	<b>SE:</b> 136-140, 141, 175, 176, 177, 321
64.03	Construct and use the loop.	<b>SE:</b> 150-155, 175, 175, 177, 307
64.04	Construct and use the enhanced for loop.	<b>SE:</b> 155-156, 176, 315, 320, 355
64.05	Describe when a while loop is used.	<b>SE:</b> 156-157, 169
64.06	Describe when a loop is used.	<b>SE:</b> 150-155, 156-157, 169, 319
65.0 Use nested loop iterative control statements in Java. – The student will be able to:		
65.01	Construct and execute a program using nested loops.	<b>SE:</b> 143-146, 154, 175, 176, 319-320, 330
65.02	Construct and execute a loop using break and continue.	Pages 155-157 may be used in preparation to meet this benchmark.
65.03	Evaluate a nested loop construct and sentinel value.	<b>SE:</b> 136-139, 143-146, 156-159, 169
66.0 Produce input and output for Java programs. – The student will be able to:		
66.01	Describe and use classes (e.g., Scanner, System) to input data into programs.	<b>SE:</b> 89-92, 129, 147-150, 153, 207-208

BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
66.02	Demonstrate the use of different ways to input data into programs using Scanner or System class.	<b>SE:</b> 89-92, 491
66.03	Describe and demonstrate the use of the System class to produce output to the console.	This objective is addressed throughout. See, for example: <b>SE:</b> 26, 55, 95-96, 129, 143, 153, 160, 194, 263, 308, 380, 526
66.04	Explain the difference between print and println functions in the System class.	<b>SE:</b> 26, 55-56, 95
66.05	Create and use escape sequences.	<b>SE:</b> 60
67.0 Use packages and import statements in a Java program. – The student will be able to:		
67.01	Describe the use of import statements.	<b>SE:</b> 83-85
67.02	Describe the use of packages.	<b>SE:</b> 82-85
67.03	Create code that uses package statements to avoid class conflict.	<b>SE:</b> 485
67.04	Create packages that abide by standard Java naming convention.	Java Programming Language, 23-31, may be used in preparation to meet this benchmark.
67.05	Demonstrate the use of Java-API to search for classes and packages.	<b>SE:</b> 82, 484
68.0 Create a Java program that uses methods. – The student will be able to:		
68.01	Differentiate between anonymous blocks and methods.	<b>SE:</b> 125-128, 192-197
68.02	Identify the benefits of using methods.	<b>SE:</b> 192-197



BENCHMARK CODE	BENCHMARK	LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
68.03	Describe a method signature.	<b>SE:</b> 192, 197, 201, 203
68.04	Create a method.	<b>SE:</b> 192-197, 201, 229-233
68.05	Describe how a method is invoked.	<b>SE:</b> 26, 54, 192, 201
68.06	Describe the purpose of overloading methods.	<b>SE:</b> 203-204, 227, 234
68.07	Create overloaded methods in programs.	<b>SE:</b> 229-233
69.0 Create a Java program that uses parameters in methods. – The student will be able to:		
69.01	Describe how parameters are passed into functions.	<b>SE:</b> 56, 199-200
69.02	Define a parameter.	<b>SE:</b> 199-200
69.03	Create a method using a parameter.	<b>SE:</b> 229-232
69.04	Invoke a method that has parameters.	<b>SE:</b> 56
69.05	Distinguish between formal and actual parameters.	<b>SE:</b> 199-200, 227
69.06	Demonstrate the use of reference parameters in methods.	<b>SE:</b> 200, 217, 249-252, 309-310
70.0 Describe and use recursion in a Java program. – The student will be able to:		
70.01	Describe the use of recursion in solving problems.	<b>SE:</b> 23, 430, 431-432, 436-440, 440-444, 444-451

BENCHMARK CODE	BENCHMARK	<b>LESSONS WHERE BENCHMARK IS DIRECTLY ADDRESSED IN-DEPTH IN MAJOR TOOL</b> (Include page numbers of lesson, a link to lesson, or other identifier for easy lookup by reviewers.)
70.02	Describe the difference of iterative and recursive methods.	<b>SE:</b> 434-435, 461
70.03	Demonstrate the use of direct recursion.	<b>SE:</b> 432-433, 434, 436-440, 446-448, 462-465, 465-467
70.04	Demonstrate the use of indirect recursion.	<b>SE:</b> 435, 465-467
<b>71.0</b> Construct Java statements that use the String class to manipulate String data. – The student will be able to:		
71.01	Explain the use of the String class.	<b>SE:</b> 77-79
71.02	Create code to concatenate strings using the concatenation operator.	<b>SE:</b> 57, 59, 107, 112, 176, 177, 211, 287
71.03	Demonstrate how to search a string using index of method of the String class.	<b>SE:</b> 79, 207
71.04	Explain the effect of immutability of Strings.	<b>SE:</b> 77
71.05	Create Strings using string literals, and through new keyword.	<b>SE:</b> 76, 78, 112, 145, 176, 177
71.06	Demonstrate the use of the following string manipulation methods of the String class: charAt, length, trim, substring, replace, starts With and ends With.	<b>SE:</b> 78-79, 145, 209-210