



Clemson University - Center for Corporate Learning  
1 North Main Street, 7th Floor,  
Greenville, SC 29601

<http://www.clemson.edu/online/>

Contact: Juanita Durham | 864.656.3984 | [jdrhm@clemson.edu](mailto:jdrhm@clemson.edu)

### Education & Training Plan

## Java Programming Specialist Certificate Program with Externship

Student Full Name: \_\_\_\_\_

Start Date: \_\_\_\_\_ End Date: \_\_\_\_\_

**Program includes National Certification & an Externship Opportunity**  
**Mentor Supported**

### Java Programming Specialist Certificate Program with Externship

Course Code: CLEM-IT-JAVA  
Program Duration: 6 Months  
Course Contact Hours: 375  
Student Tuition: \$3,999

#### The Java Programming Specialist

Professional web developers work to build and maintain high-end websites for a multitude of clients. Specialists who work in this field aren't just webmasters, but rather have the technical capability to write custom scripts that will accommodate clients' needs. Web developers require professional training, certification and experience working in different programming codes, and scripting languages. The Oracle Certified Associate, Java SE 5 / SE 6 certification provides an ideal entry into an application development or a software project management career using Java technologies. This worldwide credential validates basic knowledge of Object-Oriented (OO) Concepts, UML representation of OO concepts, the Java programming language, and general knowledge of Java Platforms and Technologies. Candidates for this exam include entry level Java programmers, students studying to become Java programmers, as well as project or program managers working with Java technology in the software development industry.

#### The Java Programming Specialist Program

Recognizing the need for skilled developers, the Java Programming Specialist program provides a platform to produce skilled developers with the ability to take into account the client's needs and vision along with the target market to create a website and accompanying mobile applications. Students who complete this program will work through hands-on exercises and labs using real Java code. Additionally, the certification exams validate students' capabilities using real-world, scenario-based questions that assess and challenge students' ability to perform as experienced web developers.

## Education and National Certifications

- Students should have or be pursuing a high school diploma or GED.
- There are no state approval and/or state requirements associated with this program.
- There are several National Certification exams that are available to students who successfully complete this program:
  - **Oracle Certified Associate JAVA SE 5 / SE 6 Exam**
  - **Microsoft Office Specialist (MOS) Certification Exam.**

## Program Objectives

At the conclusion of this program, students will be able to:

- Solve simple problems using the fundamental syntax and semantics in Java
- Examine elementary techniques in Java programming
- Write Java programs that use selection, loops, methods for transfer of control and arrays
- Examine object-oriented programming using objects and classes, including immutable
- Write Java programs that use inheritance and polymorphism
- Extend error handling techniques by adding exception handlers to Java programs
- Design generic classes and interfaces to improve code reusability
- Use the Java Collections Framework in a Java program
- Distinguish between the Java platforms and Java integration technologies
- Describe Java's server-side technologies and application tiers
- Understand the history of Java in the Open Source movement
- Explain Java's role in the Open Source movement and cutting edge technologies
- Identify common characteristics shared by Open Source communities and guidelines for successful participation in them
- Use Microsoft Office

## National Certification

Upon successful completion of this Clemson University program, students would be eligible to sit for the Oracle Certified Associate JAVA SE 5 / SE 6 exam and the Microsoft Office Specialist (MOS) exam. Although there are no state approval, state registration or other state requirements for this program, students who complete this program at Clemson University will be prepared and are eligible to sit for this national certification exam. Students who complete this program are encouraged to complete the externship option with their program. Students who complete this program can and do sit for the MOS national certification exams and are qualified, eligible and prepared to do so. Clemson University works with each student to complete the exam application and register the student to take their national certification exam.

## Externship / Hands on Training / Practicum

Although not a requirement, once students complete the program, they have the ability to participate in an externship and/or hands on practicum so as to practice the skills necessary to perform the job requirements of a professional in this field. Students will be assisted with completing a resume and/or other requirements necessary to work in this field. All students who complete this program are eligible to participate in an externship and will be placed with a participating organization near their location. Clemson University works with national organizations and has the ability to place students in externship opportunities nationwide.

**Clemson University contact:** If students have any questions regarding this program including national certification and externships, **they should call Juanita Durham of Clemson University at 864.656.3984 or via email at [jdrhm@clemsun.edu](mailto:jdrhm@clemsun.edu).**

Note: No refunds can be issued after the start date published in your Financial Award document.



## About Clemson University!

Clemson Online, a unit reporting directly to the Provost, works closely with leadership teams across the University to develop, market, and deliver top-quality courses and programs in blended and online formats. The office provides vision, leadership, coordination, and expertise in support of faculty design, delivery, and evaluation of technology-enhanced, blended, and fully online courses and instructional materials. Dynamic, transformative, and unique eLearning opportunities characterize Clemson's approach to online teaching and learning.

**Our Mission:** Clemson Online provides strategic leadership for online education, emphasizing innovative teaching and superior learning outcomes to maximize student success in 21st-century academic and professional contexts.

**Our Vision:** Clemson Online will define the public web-grant university through measurable achievements in online education, research, and service.

The office is committed to pursuing strategic opportunities, providing supportive resources, promoting superior educational quality, and ensuring faculty involvement and responsibility in shaping Clemson's online future.



## Clemson University and Pearson Education

The Clemson University's Office of Professional and Continuing Education eLearning programs were developed in partnership with Pearson Education to produce the highest quality, best-in-class content and delivery necessary to enhance the overall student learning experience, boost understanding and ensure retention. Pearson Education is the premier content and learning company in North America offering solutions to the higher education and career training divisions of colleges and universities across the country aimed at driving quality education programs to ensure student success. Please visit us at [www.pearson.com](http://www.pearson.com).

## About Pearson Education

Welcome to Pearson. We have a simple mission: to help people make more of their lives through learning. We are the world's leading learning company, with 40,000 employees in more than 80 countries helping people of all ages to make measurable progress in their lives. We provide a range of education products and services to institutions, governments and direct to individual learners, that help people everywhere aim higher and fulfil their true potential. Our commitment to them requires a holistic approach to education. It begins by using research to understand what sort of learning works best, it continues by bringing together people and organizations to develop ideas, and it comes back round by measuring the outcomes of our products.

## Java Programming Specialist Program Detailed Student Objectives:

### ELEMENTARY COMPUTING CONCEPTS

- Explain computer basics, programs, and operating systems
- Represent numbers in binary, decimal, and hexadecimal
- Explain the relationship between the Java programming language and the World Wide Web
- Distinguish the terms application programming interface (API), integrated development environment (IDE), and Java Development Kit (JDK)
- Revise a simple Java program
- Use built-in Java library calls to display output on the console
- Create, compile, and execute Java programs
- Explain the basic syntax of Java necessary to write a simple program

### Java BASICS

- Write Java programs to perform simple calculations
- Use identifiers to name variables, constants, methods, and classes
- Use variables to store data
- Write a program with assignment statements and assignment expressions
- Use constants to store permanent data
- Write expressions that declare Java primitive data types: byte, short, int, long, float, double, boolean, and char
- Use Java operators to write numeric expressions
- Represent characters using the "char" type
- Represent a string using the "String" type
- Use the Scanner class to obtain input from the console
- Describe Java documentation, programming style, and naming conventions
- Distinguish syntax errors, runtime errors, and logic errors
- Debug logic errors

### STATEMENTS

- Write expressions that declare "boolean" type, and use boolean values true and false
- Apply relational operators (<, <=, ==, !=, >, >+) and logic operators (!, &&, ||, ^) to write boolean expressions
- Describe the use of boolean expressions to control selection statements
- Use if and nested if statements to implement selection control
- Use switch statements to implement selection control
- Write expressions using the conditional operator
- Format output and strings using the "System.out.printf" method and the "String.format" method, respectively
- Explain the rules governing operator precedence and associativity

### CONTROL FLOW

- Describe the use of boolean expressions to control loop statements
- Use "while", "do-while", and "for" loop statements to control the repetition of statements
- Differentiate among the three types of loops
- Write nested loops
- Describe the flow of control in loop statements
- Use break and continue to implement program control

### TRANSFER OF CODING USING METHODS

- Write expressions that define, invoke, and pass arguments to methods
- Write reusable code

- Create new methods from existing methods using "overloading"
- Design and implement overloaded methods
- Determine the scope of variables
- Explain how to use the methods in the *Math* class
- Employ the concept of method abstraction
- Design and implement methods using stepwise refinement

### **ARRAYS/DATA ABSTRACTION IN A PROGRAM**

- Explain why arrays are necessary in programming
- Describe the steps involved in using arrays including declaring array reference variables and creating arrays
- Use indexed variables to initialize the values in an array and access array elements
- Use the "for-each" loops to simplify programming
- Develop and invoke methods with array arguments and return value
- Write expressions to declare a method with variable-length argument lists
- Use the linear or binary search algorithm to search elements
- Use the selection sort to sort an array
- Use the insertion sort algorithm to sort an array
- Use the methods in the *Arrays* class
- Write expressions to declare and create two-dimensional arrays
- Write expressions to declare and create multidimensional arrays
- Use an "Array List" to store, retrieve, and manipulate objects
- Describe objects and classes, and use classes to model objects
- Use Unified Modeling Language (UML) graphical notations to describe classes & objects
- Write expressions that declare a class and create an object from a class
- Create objects using constructors
- Distinguish between object reference variables and primitive data type variables
- Distinguish between instance and static variables and methods
- Write expressions that declare private data fields with appropriate get and set methods
- Write expressions that encapsulate data fields to make classes easy to maintain
- Differentiate between primitive-type arguments and object-type arguments
- Store and process objects in arrays

### **OBJECTS AND CLASSES (PUTTING IT ALL TOGETHER)**

- Create immutable objects from immutable classes to protect the contents of objects
- Determine the scope of variables in the context of a class
- Use the keyword "this" to refer to the object itself
- Apply class abstraction to develop software
- Explain the differences between the procedural paradigm and object-oriented paradigm
- Design programs that adhere to the object-oriented paradigm

### **INHERITANCE AND POLYMORPHISM**

- Use inheritance to develop a subclass from a superclass
- Describe a superclass and how it uses constructors and methods
- Illustrate overriding instance methods in the subclass
- Explain the role of polymorphism, dynamic binding, and generic programming in object-oriented programming
- Use the protected visibility modifier to restrict access to data and methods
- Use the "final" modifier to prevent class extending and method overriding

**STRING MANIPULATION**

- Explain immutability and scope of variables
- Use the concatenation ( + ) operator with strings
- Develop code using methods from class *String*
- Define encapsulation and information hiding
- Use Java command-line tools (*Javap*)
- Use arrays to manage a series of variables
- Explain the difference between the *String* == operator and the *String* method *equals*
- Use regular expressions with *String*
- Distinguish immutable classes from nonimmutable classes
- Explain thread safety and the difference between *StringBuffer* and *StringBuilder*
- Explain regular expressions and use string methods that require them
- Recognize extensible markup language (XML) and how regular expressions are used by XML applications such as XPath

**OBJECT-ORIENTED DESIGN**

- Explore software design using Java
- Express software designs visually using UML
- Distinguish types of class compositions and associations, including one-to-one, one-to-many, many-to-many, and association navigation
- Develop code that uses interfaces and inheritance, and abstract classes and subclasses
- Use UML to illustrate classes, inheritance, attributes and operations, class associations and compositions, and association multiplicity indicators and association navigation indicators
- Develop code that uses polymorphism

**OBJECT-ORIENTED DESIGN WITH UML – PART 1**

- Explain serialization and deserialization in Java
- Recognize JAXB and JAXB annotations in Java code
- Develop code that follows the JavaBeans coding guidelines
- Explain the difference between JavaBeans and POJOs, and explain the purpose of the JavaBeans coding conventions—setters and getters
- Understand good versus bad programming habits, and when to use public, private, and protected scope modifiers
- Describe at a high level the benefits and basic characteristics of Remote Method Invocation (RMI)

**OBJECT-ORIENTED DESIGN WITH UML – PART 2**

- Develop code that uses polymorphism for both classes and interfaces
- Recognize code that uses the program-to-an-interface principle
- Explain the value of polymorphism as it relates to code reuse and application extensibility
- Develop code that uses polymorphism
- Describe the purpose of packages in the Java language
- Recognize the proper use of import and package statements
- Demonstrate the proper use of the *Javac* command, including the command-line options *-d* and *-classpath*
- Demonstrate the proper use of the *Java* command, including the command-line options *-classpath*, *-d*, and *-version*

**OBJECT-ORIENTED DESIGN WITH UML – PART 3**

- Explain the purpose of design patterns and how design patterns apply the program-to-an-interface principle

- Recognize common design patterns and write Java code that applies design patterns
- Distinguish the basic characteristics of the three Java platforms: J2SE, J2ME, and J2EE
- Select the appropriate Java platform or platforms
- Recognize the current trends in computer programming and how Oracle is positioning Java to compete in an always-connected world
- Explain the current state and future evolution of the Java platforms

### **CLIENT TECHNOLOGIES**

- Describe the basic characteristics of servlets, JavaServer Pages, and servlet and JavaServer Pages support for hypertext markup language thin clients
- Describe the basic characteristics, benefits, drawbacks, and deployment issues related to creating thin clients using the hypertext markup language and JavaScript
- Describe the basic characteristics, benefits, drawbacks, and deployment issues related to creating fat clients using applets
- Describe the basic characteristics, benefits, drawbacks, and deployment issues related to creating fat clients using Swing
- Explain the basic concepts of relational database management systems
- Compare and contrast the most common pay-for, shareware, and free relational database management system solutions available today

### **ENTERPRISE Java**

- Describe the basic characteristics of Enterprise JavaBeans
- Describe the basic characteristics of the Java Naming and Directory Interface
- Describe the basic characteristics of message queues
- Describe the basic characteristics of Simple Mail Transfer Protocol and JavaMail
- Describe the benefits and basic characteristics of Java database connectivity, the structured query language, and relational database management system technologies

### **ORIGINS OF THE Java PROGRAMMING LANGUAGE**

- Summarize the features incorporated into the first Java release and the corresponding challenges they were designed to resolve

### **Java'S IMPACT**

- Summarize the features and capabilities of the Java Standard Edition, Java Micro Edition, and Java Enterprise Edition

### **CONTRIBUTORS TO Java'S SUCCESS**

- Describe Dr. James Gosling's role in the establishment and evolution of Java
- Describe the ongoing role of Java evangelists and Java User Groups in the continued success of Java

### **Java'S ROLE IN THE OPEN SOURCE MOVEMENT**

- Summarize the significance of Sun's decision to release Java under the General Public License

### **Java'S ROLE IN CUTTING EDGE TECHNOLOGIES**

- Summarize Java's role in three current technologies or devices

### **THE HISTORY AND EVOLUTION OF OPEN SOURCE**

- Briefly explain the social positioning aspect of open source software
- Summarize the three waves of the open source movement and the key individuals and organizations who contributed to each wave

**OPEN SOURCE BUSINESS AND COMMUNITIES**

- Summarize characteristics of open source software that make it suitable for use in business enterprises
- Identify a few ways individuals and companies are able to earn money by working with open source software
- Distinguish the three types of open source software licenses
- Describe the mission of the Apache Software Foundation
- Summarize the purpose served by the Linux Online community
- Describe the mission and impact of the OpenSolaris Communities
- Summarize the purpose of the OpenOffice community
- Describe the purpose of the Project Wonderland community

**PARTICIPATING IN OPEN SOURCE COMMUNITIES**

- Recognize characteristics shared among open source communities
- Outline practical steps to take to become an active participant in any open source community

**Note:** This program can be completed in 6 months. However, students will have online access to this program for a 24-month period.

**MICROSOFT OFFICE Module**

- Use an integrated software package, specifically the applications included in the Microsoft Office suite
- Demonstrate marketable skills for enhanced employment opportunities
- Describe proper computer techniques for designing and producing various types of documents
- Demonstrate the common commands & techniques used in Windows desktop
- List the meaning of basic PC acronyms like MHz, MB, KB, HD and RAM
- Use WordPad and MSWord to create various types of documents
- Create headings and titles with Word Art
- Create and format spreadsheets, including the use of mathematical formulas
- Demonstrate a working knowledge of computer database functions, including putting, processing, querying and outputting data
- Define computer terminology in definition matching quizzes
- Use the Windows Paint program to alter graphics
- Use a presentation application to create a presentation with both text and graphics
- Copy data from one MS Office application to another application in the suite
- Use e-mail and the Internet to send Word and Excel file attachments
- Demonstrate how to use the Windows Taskbar and Windows Tooltips
- Explain how copyright laws pertain to data and graphics posted on the Internet
- Take the college computer competency test after course completion
- Follow oral and written directions and complete assignments when working under time limitations

**Note:** Although the Microsoft Office Module is not required to successfully complete this program, students interested in pursuing free Microsoft MOS certification may want to consider completing this Microsoft Office Module at no additional cost.



**System Requirements:**

**Windows Users:**

- Windows 8, 7, XP or Vista
- 56K modem or higher
- Soundcard & Speakers
- Firefox, Chrome or Microsoft Internet Explorer

**Mac OS User:**

- Mac OS X or higher (in classic mode)
- 56K modem or higher
- Soundcard & Speakers
- Apple Safari

**iPad Users:**

- Due to Flash limitations, eLearning programs are NOT compatible with iPads

**Screen Resolution:**

- We recommend setting your screen resolution to 1024 x 768 pixels.

**Browser Requirements:**

- System will support the two latest releases of each browser. When using older versions of a browser, users risk running into problems with the course software.
- Windows Users: Mozilla Firefox, Google Chrome, Microsoft Internet Explorer
- Mac OS Users: Safari, Google Chrome, Mozilla Firefox

**Suggested Plug-ins:**

- Flash Player
- Real Player
- Adobe Reader
- Java