

Syllabus

Course Overview

In this course, you will learn and practice the fundamentals of the Java programming language. You will learn and apply variables, operators, literals, and how to combine them to create expressions. You will study and develop the different types of Java statements like the assignment, selection, and looping statements. You will also apply the Java built-in array data structures in your applications. And you will practice some of Java object-oriented features, such as defining classes and instantiating them into objects.

Course Outline

Unit 1 introduces programming in general and Java programming in particular. You will learn how to set up your NetBeans Java development environment and how to write, compile, and execute your first Java program. You will write an APA-formatted paper about the software development process and your role in it as a Java programmer.

Unit 2 begins your learning about the many features of Java. You will learn about declaring Java variables, literals and how to combine them to create expressions that evaluate to a value. You will develop a simple Java application that adds and multiplies three integers.

Unit 3 focuses on Java selection statements. You will learn the various types of selection statements like if and switch statements and how and when to use each. You will debug and fix a non-working application that uses if statements.

Unit 4 addresses the two commonly used Java built-in data types of char and String. You will also learn how to apply the Java Math class. You will develop an application that validates user input using Java char and String data types.

Unit 5 examines Java looping statements. Java looping statements include the for, while, do while statements. Using Java loops, you will develop an application that finds the lowest and highest of five integers.

Unit 6 discusses Java methods. Methods makes your program modular and easy to modify. You will learn how to define methods and how to invoke them. In Unit 6, you will complete the development of a Java application that uses methods.

Unit 7 begins our learning about the Java built-in array data structures. In Unit 7, you will learn about the Java single-dimensional array, and you will develop an application that uses Java single-dimensional arrays.

Unit 8 focuses on Java multidimensional arrays and how to use them. You will debug and fix an application that deploys a 2-dimensional array, but the application does not compile nor run.

Unit 9 begins our studies of Java object-oriented features. In this unit, you will learn how to define Java classes and how to instantiate objects from these classes. You will develop a Java application that applies these object-oriented features.

Unit 10 goes deeper into Java object-oriented features like abstraction and encapsulation. You will complete the development of a Java application that uses the classes and objects features of the language.

Course Competencies

(Read Only)

To successfully complete this course, you will be expected to:

- 1 Explain the responsibilities of a programmer in software development.
- 2 Identify fundamental programming processes, logic, and constructs.
- 3 Design simple Java console programs according to specifications.
- 4 Code Java console applications using foundational Java features.
- 5 Apply tools and testing for successful program execution.
- 6 Explain Java program design decisions or strategies to stakeholders or team members.
- 7 Communicate efficiently, effectively, and in an appropriate manner for an IT professional.

Course Prerequisites

There are no prerequisites for this course.

Required

The materials listed below are required to complete the learning activities in this course.

Integrated Materials

Once the [Capella University Bookstore](#) opens for the quarter, as a registered learner you will receive an e-mail containing a direct link you can use to obtain your materials. Please follow the instructions provided to you by the bookstore to download digital materials. Some materials are available only in hard-copy format and will be shipped to you. The bookstore will indicate any materials that may involve shipping.

Book

Liang, Y. D. (2014). *Intro to Java programming, comprehensive version*. (10th ed.). New York, NY: Pearson. ISBN: 9780133761313

Software

NetBeans. (n.d.). Ada for NetBeans - plugin detail [Computer software add-in for ADA accessibility]. Available from <http://plugins.netbeans.org/plugin/13977/ada-for-netbeans>

NetBeans. (n.d.). Java standard edition (8th ed.) [Computer software]. Available from <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

NetBeans. (n.d.). NetBeans integrated development environment (8th ed.) [Computer software]. Available from <https://netbeans.org/downloads/>

Miscellaneous Item

Pearson Education. (2017). Intro to Java programming [Custom Web site]. Retrieved from http://www.pearsoncustom.com/mn/capella_java/

External Resource

Please note that URLs change frequently. While the URLs were current when this course was designed, some may no longer be valid. If you cannot access a specific link, contact your instructor for an alternative URL. Permissions for the following links have been either granted or deemed appropriate for educational use at the time of course publication.

- Brown, S. (2008). [The key difference between developer and architect roles](#). Retrieved from CodingtheArchitecture.com
- LinkedIn SlideShare. (n.d.). [6 basic steps of software development](#). Retrieved from <https://www.slideshare.net/RiantSoft123/6-basic-steps-of-software-development-process>
- Microsoft. (2017). [Team functions and responsibilities](#). Retrieved from [https://msdn.microsoft.com/en-us/library/aa266916\(v=vs.60\).aspx](https://msdn.microsoft.com/en-us/library/aa266916(v=vs.60).aspx)
- Oracle. (n.d.). [The Java tutorials: "Hello world!" for the NetBeans IDE](#). Retrieved from <https://docs.oracle.com/javase/tutorial/getStarted/cupojava/netbeans.html>
- Solve100.com. (2015). [Five key roles and responsibilities for software development projects success](#). Retrieved from <http://solve100.com/responsibilities-for-software-development-success/>
- Tutorialspoint.com. (n.d.). [SDLC: Overview](#). Retrieved from https://www.tutorialspoint.com/sdlc/sdlc_overview.htm
- Workable.com. (2017). [Java developer job description](#). Retrieved from <https://resources.workable.com/java-developer-job-description>

Suggested

The following materials are recommended to provide you with a better understanding of the topics in this course. These materials are not required to complete the course, but they are aligned to course activities and assessments and are highly recommended for your use.

Optional

The following optional materials are offered to provide you with a better understanding of the topics in this course. These materials are not required to complete the course.

Library

The following suggested Skillsoft resources are available via the Capella University Library.

- Campbell, J. (2015). [Java SE 8 fundamentals: Compiling and running a program \[Video\]](#). Skillsoft Ireland.
- Campbell, J. (2015). [Java SE 8 fundamentals: Working with data types \[Video\]](#). Skillsoft Ireland.
- Skillsoft, (n.d.). [Getting started with software programming. \[Tutorial\]](#).

External Resource

Please note that URLs change frequently. While the URLs were current when this course was designed, some may no longer be valid. If you cannot access a specific link, contact your instructor for an alternative URL. Permissions for the following links have been either granted or deemed appropriate for educational use at the time of course publication.

- Astah. (n.d.). [Astah community](#). Retrieved from <http://astah.net/editions/community>
- [Gliffy](#). (n.d.). Retrieved from <http://www.gliffy.com>
- Gnome.org. (n.d.). [Dia](#). Retrieved from <https://wiki.gnome.org/Apps/Dia>
- [io](#). (n.d.). Retrieved from <https://www.draw.io>
- [Lucidchart](#). (2019). Retrieved from <https://www.lucidchart.com/>
- Programming Knowledge. (2016). [How to download and install NetBeans IDE on Windows](#). Retrieved from <https://youtu.be/pgr-Bzkc4zg>
- Vigil, A. (2014). [NetBeans setup on a Mac](#). Retrieved from <https://youtu.be/2t4j5RFjO9E>
- Visual Paradigm. (n.d.). [Visual Paradigm](#). Retrieved from <http://www.visual-paradigm.com/features/database-design/>

Unit 1 >> Introduction to Java Programming

Introduction

Welcome to this very first unit!

You begin this Introduction to Programming with Java course by learning about computers, software development, and the Java programming language.

You will learn about computers and how they work. You will learn about the software development process and the role a programmer plays in this process. And you will also learn about the Java programming language and get an overview of its many

programming features.

In this unit, and all other units in this course, you will use the NetBeans IDE to develop your Java applications. In this unit, you will setup your NetBeans development environment so that you can use it for completing your weekly assignments.

Learning Activities

u01s1 - Software Preparation and Technology Access

In this course, you will be using software and technology that is needed to complete designated activities and assignments. There is no additional cost for this software and technology. Some software packages will be made available to you at no additional cost through Capella's subscription with Microsoft, while other software packages are available for free download through open-source licensing.

Capella University requires learners to meet certain minimum [computer requirements](#). Please note that some software required for a course may exceed these minimum requirements. Check the requirements for the software you may need to download and install to make sure it will work on your device. Most software will require a Windows PC. If you use a Mac, refer to [Installing a Virtual Environment and Windows on a Mac](#).

The software and technologies below are strongly recommended to support you in completing the course objectives. If you have access to other tools that you believe may still meet the requirements of this course, please discuss your selected alternatives with your instructor.

If you use assistive technology or any alternative communication methods to access course content, please contact [Disability Services](#) with any access-related questions or to request accommodations.

For this course, follow the instructions provided through the links below to download and install software or register for an account, as required.

Open-Source Software

- NetBeans, Java SE 8: Go to [Downloading and Installing NetBeans and Java SE 8/JDK](#)

If you encounter any difficulties in the download and installation process, post a detailed question in the Ask Your Instructor section of the course. Your instructor should be able to help you or point you in the right direction for the answers you need.

u01s2 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about computers, programming them, and the Java programming language:

- Chapter 1, "Introduction to Computers, Programs, and Java," pages 1–32.

Use the Internet to review the following:

- [6 Basic Steps of Software Development.](#)
- [The Key Difference Between Developer and Architect Roles.](#)
- [Team Functions and Responsibilities.](#)
- [Five Key Roles and Responsibilities for Software Development Projects Success.](#)
- [SDLC: Overview.](#)
- [Java Developer Job Description.](#)

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site.](#)

- **Note:** After you transact for course materials in your learner cart at the Capella Bookstore, you will receive an access code (electronic ancillary) and an e-mail with instructions from the Capella Bookstore for navigating this Pearson custom Web site.

Review these resources to learn how to code, compile, and run your first Java application using the NetBeans IDE. You will also learn about the software development process and its various stages.

- Your First Java Program.
 - On the textbook custom Web site > VideoNotes > Chapter 1.
- Compile and Run a Java Program.
 - On the textbook custom Web site > VideoNotes > Chapter 1.
- NetBeans Brief Tutorial.
 - On the textbook custom Web site > VideoNotes > Chapter 1.
- Software Development Process.
 - On the textbook custom Web site > VideoNotes > Chapter 2.

Multimedia

You may view the following walkthrough to help you prepare for the unit assignment:

- [Compile, Debug Errors in Your First Project Walkthrough.](#)

Optional Skillsoft Resource

- Skillsoft, (n.d.). [Getting started with software programming.\[Tutorial\]](#).

u01s3 - Discussion Preparation

It is now customary that the first program you develop when learning a new programming language is the Hello, World application. The Hello, World application is a simple application yet it provides an opportunity for you to setup the development environment and test it out for successful operations. We will use the NetBeans Integrated Development Environment (IDE) to develop your Java programs in this course. Following the tradition, your first Java console (command line application) will be the Hello, World application.

In this week's discussion, you will setup your Java development environment using the NetBeans IDE and use it to develop your first Java console application that displays the words "Hello, World!" to the screen. Review the resources section to help you with this activity.

Download, install, and configure the NetBeans IDE to your computing device. Then, using NetBeans, create a new Java console application that displays "Hello, World!" on the screen.

You will be expected to provide a screenshot of your result of your work. Your screenshot should look like the "Hello World Result" course file.

Course Resources

[Opening an Existing Project in NetBeans](#)

[Hello World Result](#)

[The Java Tutorials:"Hello World!" for the NetBeans IDE](#)

[How to Download and Install NetBeans IDE on Windows](#)

[NetBeans Setup on a Mac](#)

u01d1 - Your First Java Console Application

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

To prepare for this discussion, set up your Java development environment using the NetBeans IDE and develop your first Java console application that displays the words "Hello, World!" to the screen. Review the resources in the Unit 1 Discussion Prep Study.

Download, install, and configure the NetBeans IDE to your computing device. Then, using NetBeans, create a new Java console application that displays "Hello, World!" on the screen.

For this discussion:

- Provide a screenshot of the result of your work. Your screenshot should look like the "Hello World Result."
- Explain, briefly, how you completed this exercise, the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

[Undergraduate Discussion Participation Scoring Guide](#)

Hello World Result

u01a1 - Software Development and Your Role in It

In this assignment, you will articulate the roles played by team members in a software development project. You will explain how the Java programming language and development environment enable you to perform your role as a software programmer/developer in a development project.

Please refer to the studies in this unit to learn about the software development process and the roles played in a typical software development project. You may wish to reference these resources in your assignment.

Follow these steps to complete this assignment:

1. Research the software development process and the roles played by the development team members in software development projects.
 - o Create a document named "U1A1_SWRoles.docx" of 2–3 pages in which you:
 - a. Describe the software development process and its phases.
 - b. Explain at least four of the roles played by team members in a software development project and examine their corresponding responsibilities. The four roles must include the role of the software developer/programmer.
 - c. As an aspiring Java programmer, articulate how the Java programming language and the Java development environment enable you to perform the role of a Java developer in a software development project.

Your document should follow APA style and guidelines, including proper spelling and grammar. See the APA Module linked in the Resources for more information.

Deliverables

Submit your paper as Word document saved as: "U1A1_SWRoles.doc."

Your assignment will be scored on the following criteria:

1. Describe the software development process and its phases.
2. Explain roles and responsibilities of the developer/programmer teams.
3. Articulate how the programming language and development environment enable the role of a developer/programmer.
4. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Resources

Use the following, linked in the Resources:

- 6 Basic Steps of Software Development.
- The Key Difference Between Developer and Architect Roles
- Team Functions and Responsibilities.
- Five Key Roles and Responsibilities for Software Development Projects Success.
- SDLC: Overview.

- Java Developer Job Description.

Course Resources

[APA Module](#)

[6 Basic Steps of Software Development](#)

[SDLC: Overview](#)

[The Key Difference Between Developer and Architect Roles](#)

[Java Developer Job Description](#)

[Team Functions and Responsibilities](#)

[Five Key Roles and Responsibilities for Software Development Projects Success](#)

Unit 2 >> Working with Variables, Operators, and Expressions

Introduction

In Unit 2, you will learn about the fundamental concepts in the Java programming language. You will learn about variables and how they are used to store values. You will also learn about operators and how they are used to combine variables and values into Java expressions.

You will develop a simple Java application that adds and multiplies three integer values.

Learning Activities

u02s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about the various Java operators and how to combine them to create Java expressions and statements:

- Chapter 2, "Elementary Programming," pages 33–74.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

- **Reminder:** You received an access code and an e-mail with instructions from the Capella Bookstore for this Pearson custom Web site.

Review these resources to learn about Java operators and how to use them in your Java programs:

- Obtain Input.
 - On the textbook custom Web site > VideoNotes > Chapter 2.
- Use Operators / and %.
 - On the textbook custom Web site > VideoNotes > Chapter 2.
- Compute Loan Payments.
 - On the textbook custom Web site > VideoNotes > Chapter 2.
- Compute BMI.
 - On the textbook custom Web site > VideoNotes > Chapter 2.

Multimedia

You may view the following walkthrough to help you prepare for the unit assignment:

- [Input and Output of Various Data Types Walkthrough](#).

Optional Skillsoft Resource

- Campbell, J. (2015). [Java SE 8 fundamentals: Working with data types \[Video\]](#). Skillsoft Ireland.

u02s2 - Discussion Preparation

Reading from the console (keyboard) and writing to the screen are common tasks in Java programming.

To prepare for this week's discussion, you will practice these tasks by echoing the value read from the console to the screen.

Create a Java program that reads in any value entered at the console and then prints it out to the screen.

You will be expected to provide a screenshot of the result of your work. Your screenshot should look like the example in the "Echo Input Result" resource.

Course Resources

Echo Input Result

u02d1 - Echo Input from the Console to the Screen

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

To prepare for this discussion, you practiced echoing the value read from the console to the screen. Create a Java program that reads in any value entered at the console and then prints it out to the screen.

- Provide a screenshot of your result of your work. Your screenshot should look like the example in the "Echo Input Results" resource.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

Echo Input Result

u02a1 - Add and Multiply Three Integers

In this assignment, you will design and code a simple Java application that reads in three integer values and prints out both the sum and the product of these three values.

Your program output should look like the sample output provided in the "Add and Multiply Three Integers" course file resource.

Complete instructions and required templates are included in the Resources section (Week X Solution Submission Template and Add and Multiply Three Integers Instructions).

Your assignment will be scored on the following criteria:

1. Design a program that meets Java arithmetic operators requirements.
2. Code an application that exercises fundamental constructs.
3. Test the application and document testing.
4. Explain the approach taken to develop the application and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

Opening an Existing Project in NetBeans

Week X Solution Submission Template

Add and Multiply Three Integers Instructions

Unit 3 >> Making Decisions Using Selection Statements

Introduction

Unit 3 introduces Java selection statements. Selection statements like if and switch statement allow us to make a decision in our program and to select a path to follow in our code based on the outcome of the decision.

In this unit, you will debug a non-working Java application that uses selection statement and fix it so that it meets its stated requirements.

Learning Activities

u03s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about the different selection statements provided by Java:

- Chapter 3, "Selections," pages 75–118.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about Java selection statements and how to program them:

- Program Addition Quiz.
 - On the textbook custom Web site > VideoNotes > Chapter 3.
- Use Multi-way if-else Statements.
 - On the textbook custom Web site > VideoNotes > Chapter 3.
- Sort Three Integers.
 - On the textbook custom Web site > VideoNotes > Chapter 3.

Multimedia

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Debug and Fix If, Else/If, and Else Statements Walkthrough](#).
- [Converting If Statements to Switch Statements Walkthrough](#).

Optional Skillsoft Resource

- Campbell, J. (2015). [Java SE 8 fundamentals: Compiling and running a program \[Video\]](#). Skillsoft Ireland.

u03s2 - Discussion Preparation

It is easy to make compile time errors when you start writing programs in Java. We must follow the language rules to get our code compiled without errors.

To prepare for this discussion, you will practice debugging a Java program that does not compile and fixing it so that it compiles without errors.

Consider the Java program in the "Four Compile Errors" resource.

The program is to read in an integer value and then determines if the value is a multiple of 5 number, an even number, or neither.

But the program has four compile errors. Identify these compile errors and propose remedies for each compile error. Note that these errors are compile errors and are not runtime or program logic errors.

You will be expected to provide a screenshot of your result of your work.

Course Resources

[Four Compile Errors Result](#)

u03d1 - Debug and Fix Four Compile Time Errors

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, you practiced debugging a Java program that does not compile and fixing it so that it compiles without errors. The program is to read in an integer value and then determines if the value is a multiple of 5 number, an even number, or neither. But the program has four compile errors.

- Identify these compile errors and propose remedies for each compile error. Note that these errors are compile errors and are not runtime or program logic errors.
- Provide a screenshot of your result of your work. (See the example in the "Four Compile Errors Results" resource.)
- Explain, briefly, how you completed this exercise, the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Four Compile Errors Result

u03a1 - Debug and Fix if Statements

In this assignment, you will debug and fix a given Java console application that uses if statements, but the application does not compile nor execute.

Your program output should look like the sample output provided in the "Debug and Fix if Statements" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template provided in the resources, WeekXSolutionSubmissionTemplate.docx.

Your assignment will be scored on the following criteria:

1. Identify Java decision statement bugs in a program using development tools.
2. Code an application to fix bugs.
3. Test the application and document testing.
4. Explain the approach taken to complete the fix and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

Week X Solution Submission Template

Debug and Fix if Statements Instructions

DebugFixIFStmts.zip

Unit 4 >> Working with Math Functions, Char Types, and String Types

Introduction

Unit 4 examines the methods of Java Math class and how to invoke them in your application. In Unit 4, you will also learn about the char and the String data types of Java, and many of their usage patterns.

You will develop a simple Java application that validates user input using the char and String data types.

Learning Activities

Readings

Use your *Intro to Java Programming* textbook to read the following about how mathematical functions are implemented using Java as well as text strings:

- Chapter 4, "Mathematical Functions, Characters, and Strings," pages 119–156.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn how to develop Java applications using mathematical functions and strings:

- Introduce Math Functions.
 - On the textbook custom Web site > VideoNotes > Chapter 4.
- Introduce Strings and Objects.
 - On the textbook custom Web site > VideoNotes > Chapter 4.
- Compute Great Circle Distance.
 - On the textbook custom Web site > VideoNotes > Chapter 4.

Multimedia

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Demonstrate the Uses of the Java Math Class in a Program Walkthrough](#).
- [Design a Program That Validates Char and String Data Types and Patterns Walkthrough](#).

u04s2 - Discussion Preparation

In many Java programming situations, we need to handle with each character of a string data type and manipulate it separately.

In this discussion, you will explore some of the available methods of the Character class to manipulate individual characters.

Create a Java program that reads in a single character entered at the console and then prints it out if the character is a letter, a digit, or neither.

You will be expected to provide screenshots of your result of your work. Your screenshots should look like those in the "IsLetterIsDigitElse Result" resource.

Course Resources

IsLetterIsDigitElse Result

u04d1 - Determine if a Character is a Letter, a Digit, or Neither

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, you explored some of the available methods of the Character class to manipulate individual characters.

- Create a Java program that reads in a single character entered at the console and then prints it out if the character is a letter, a digit, or neither.
- Provide screenshots of the result of your work. Your screenshots should look like those in the "IsLetterIsDigitElse Result" Resource.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

IsLetterIsDigitElse Result

u04a1 - Validate User Input Using Java Chars and Strings

In this assignment, you will design and code a Java console application that validates the data entry of a course code (like IT4782) and report back if the course code is valid or not valid. The application uses the Java char and String data types to implement the validation.

Your program output should look like the sample output provided in the "Validate User Input Using Java Chars and Strings Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template, WeekXSolutionSubmissionTemplate.docx.

Your assignment will be scored on the following criteria:

1. Design a program that meets char and String processing requirements.
2. Code an application that validates data entry.
3. Test the application and document testing.
4. Explain the approach taken to complete data validation and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Validate User Input Using Java Chars and Strings Instructions

Week X Solution Submission Template

Unit 5 >> Iterating over Statements Using Looping Statements

Introduction

In Unit 5, you will examine the different Java statement for looping or iterating over statements. Looping statements include the for, while, and do while statements. You will learn how to program these looping statements and when to use each.

You will develop a Java console application that finds the highest and lowest integers among five given integers.

Learning Activities

u05s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about the different Java statements that supports looping and iterating over a number of other statements:

- Chapter 5, "Loops," pages 157–202.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java looping statements in your application:

- Guess a Number.
 - On the textbook custom Web site > VideoNotes > Chapter 5.
- Multiple Subtraction Quiz.
 - On the textbook custom Web site > VideoNotes > Chapter 5.
- Display Loan Schedule.
 - On the textbook custom Web site > VideoNotes > Chapter 5.

Multimedia

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Design a Program That Meets Java Looping Statements Requirements Walkthrough.](#)
- [Find the Highest and Lowest of Five Integers Using Java Loops Walkthrough.](#)

u05d1 - Add Five Integers Using Java Loops

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, practiced looping constructs by repeatedly reading the values of integers from the console and tallies their sum.

- Create a Java program that reads in five integer values entered at the console and tallies their sum. The program then prints the sum out to the screen.
- Provide a screenshot of the result of your work. Your screenshot should look like the "Add5IntsWithLoops Results" resource.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

Add5IntsWithLoops Results

u05s2 - Discussion Preparation

In almost every Java program, looping constructs are used to repeat the execution of a number of Java statements.

In this discussion, you will practice these looping constructs by repeatedly reading the values of integers from the console and tallies their sum.

Create a Java program that reads in five integer values entered at the console and tallies their sum. The program then prints the sum out to the screen.

You will be expected to provide a screenshot of your result of your work. Your screenshot should look like the "Add5IntsWithLoops Results" resource.

Course Resources

u05a1 - Find Highest and Lowest of Five Integers Using Java Loops

In this assignment, you will design and code a Java console application that takes as input five integer values and produces as output the lowest and highest values of these five integer values. The application uses Java looping constructs to implement its functionality.

Your program output should look like the sample output provided in the "Find Highest and Lowest of Five Integers Using Java Loops Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the provided submission template, WeekXSolutionSubmissionTemplate.docx.

Your assignment will be scored according to the following criteria:

1. Design a program that meets Java looping statements requirements.
2. Code an application that exercises looping constructs.
3. Test the application and documents testing.
4. Explain the approach taken to develop the application and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

[Week X Solution Submission Template](#)

[Find Highest and Lowest of Five Integers Using Java Loops Instructions](#)

Unit 6 >> Defining Methods and Invoking Them

Introduction

In Unit 6, you will examine Java methods. Using Java methods results in modular programs that are easy to understand and to modify. You will learn how to define a Java method and how to invoke or call it when needed.

You will complete the development of a Java console application that registers students for classes using Java methods.

Learning Activities

u06s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about how to define Java methods and how to invoke them:

- Chapter 6, "Methods," pages 203–244.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java methods in your application:

- Define/Invoke Max Method.
 - On the textbook custom Web site > VideoNotes > Chapter 6.
- Use Void Method.
 - On the textbook custom Web site > VideoNotes > Chapter 6.
- Modularize Code.
 - On the textbook custom Web site > VideoNotes > Chapter 6.

Multimedia

Click on **Console Register for Course Result** to launch video.

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Java Methods With Return Value Walkthrough](#).
- [Java Methods With Parameters Walkthrough](#).
- [Java Method Without Return Value and Parameters Walkthrough](#).

Course Resources

[Console Register for Course Result](#)

u06s2 - Discussion Preparation

Java methods are used to modularize our applications so that the applications are easy to understand and to modify.

For this discussion, you will practice using Java methods to echo input from the console to the screen.

Create a Java program that reads in any value entered at the console and then prints it out to the screen. The program must have at least three methods including the main() method.

Provide a screenshot of your result of your work. Your screenshot should look like the "Echo Using Methods Results" resource.

Course Resources

u06d1 - Echo Input from the Console to the Screen Using Methods

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, you practiced using Java methods to echo input from the console to the screen. Create a Java program that reads in any value entered at the console and then prints it out to the screen. The program must have at least three methods including the main() method.

- Provide a screenshot of the result of your work.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

[Undergraduate Discussion Participation Scoring Guide](#)

[Echo Using Methods Results](#)

u06a1 - Complete the Programming of a Java Console Application Using Methods

In this assignment, you will complete the programming of a Java method in a console application that registers students for courses in a term of study. The application does compile and does run, but it does not produce the expected result as stated in its requirements.

Your program output should look like the sample output provided in the "Complete the Programming of a Java Console Application Using Methods Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template, `WeekXSolutionSubmissionTemplate.docx`.

Your assignment will be scored on the following criteria:

1. Design a program that meets Java method invocation requirements.
2. Code an application that uses methods.
3. Test the application and document testing.
4. Explain the approach taken to complete this assignment and the major decisions made.

5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

[Console Register for Course Result](#) | [Transcript](#)

U6A1_ConsoleRegisterForCourse.zip

Week X Solution Submission Template

Complete the Programming of a Java Console Application Using Methods Instructions

Unit 7 >> Working with Single-Dimensional Arrays

Introduction

In Unit 7, you will learn about Java arrays. In particular, you will learn about single-dimensional arrays and how to create, initialize, and access them.

You will develop a Java console application that counts occurrences in seven integers using single-dimensional arrays.

Learning Activities

u07s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about creating, initializing, and using Java single-dimensional array structures:

- Chapter 7, "Single-Dimensional Arrays," pages 245–286.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java single-dimensional array structures:

- Random Shuffling.
 - On the textbook custom Web site > VideoNotes > Chapter 7.
- Deck of Cards.
 - On the textbook custom Web site > VideoNotes > Chapter 7.

- Selection Sort.
 - On the textbook custom Web site > VideoNotes > Chapter 7.
- Command-line Arguments.
 - On the textbook custom Web site > VideoNotes > Chapter 7.

Multimedia

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Single-Dimensional Array in Java Walkthrough](#).
- [Parallel Arrays in Java Walkthrough](#).

u07s2 - Discussion Preparation

Java arrays data structures allow us to store multiple values in a single easy to manipulate data structure.

For this discussion, you will practice using Java arrays to store values entered at the console and then to print them out to the screen. That is, you will practice echoing the entered values using arrays.

Create a Java program that reads in five integer values entered at the console and stores them into an array. The program then prints these integer values out to the screen.

Provide a screenshot of your result of your work. Your screenshot should look like the "Echo5Ints Using Arrays Result" resource.

Course Resources

Echo5Ints Using Arrays Result

u07d1 - Echo Five Integers Using Java Arrays

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

In this discussion, you will practice using Java arrays to store values entered at the console and then to print them out to the screen. That is, you will practice echoing the entered values using arrays.

- Create a Java program that reads in five integer values entered at the console and stores them into an array. The program then prints these integer values out to the screen.
- Provide a screenshot of the result of your work.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

Echo5Ints Using Arrays Result

u07a1 - Count Occurrences in Seven Integers Using Java Single Dimension Arrays

In this assignment, you will design and code a Java console application that reads in seven integer values and prints out the number of occurrences of each value. The application uses the Java single dimension array construct to implement its functionality.

Your program output should look like the sample output provided in the "Count Occurrences in Seven Integers Using Java Single Dimension Arrays Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template available in the resources, WeekXSolutionSubmissionTemplate.docx.

Your assignment will be scored on the following criteria:

1. Design a program that meets Java single dimensional arrays requirements.
2. Code an application that exercises single array constructs.
3. Test the application and document testing.
4. Explain the approach taken to develop the application and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

Count Occurrences in Seven Integers Using Java Single Dimension Arrays Instructions

Week X Solution Submission Template

Unit 8 >> Working with Multidimensional Arrays

Introduction

In Unit 8, you will continue your study of Java arrays. In this unit, you will learn about Java multidimensional arrays and applications that use them.

You will debug and fix a Java console application that does not compile nor run.

Learning Activities

u08s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about Java multidimensional arrays and their applications:

- Chapter 8, "Multidimensional Arrays," pages 287–320.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java multidimensional arrays in your application:

- Find the Row with the Largest Sum.
 - On the textbook custom Web site > VideoNotes > Chapter 8.
- Grade Multiple-Choice Test.
 - On the textbook custom Web site > VideoNotes > Chapter 8.
- Even Number of 1s.
 - On the textbook cusotm Web site > VideoNotes > Chapter 8.

Multimedia

Click on **2D Console Register for Course** to launch video (3 minutes).

You may view the following walkthroughs to help you prepare for the unit assignment:

- [Two-Dimensional Arrays in Java Walkthrough](#).
- [Calculate Average in a Two-Dimensional Array Walkthrough](#).

Course Resources

[2D Console Register for Course](#)

u08s2 - Discussion Preparation

Java multidimensional arrays can be thought of as a number of single-dimensional arrays stored in one single structure.

For this discussion, you will practice using a Java two-dimensional array to count the number of occurrences of seven integer values entered by the user from the keyboard.

Create a Java program that reads in seven integer values entered at the console, counts the number of occurrences of each of the seven values, and prints these number of occurrences to the screen.

You will be expected to provide a screenshot of your result of your work. Your screenshot should look like the "2DNumOfOccurin7Ints Result" resource.

Course Resources

2DNumOfOccurin7Ints Result

u08d1 - Count Occurrences in Seven Integers Using Java Two Dimension Arrays

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

In this discussion, you will practice using a Java 2-dimensional array to count the number of occurrences of seven integer values entered by the user from the keyboard.

- Create a Java program that reads in seven integer values entered at the console, counts the number of occurrences of each of the seven values, and prints the number of occurrences to the screen.
- Provide a screenshot of the result of your work.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

2DNumOfOccurin7Ints Result

u08a1 - Debug and Fix a 2-Dimensional Array Java Console Application

In this assignment, you will debug and fix a given Java console application that uses 2-dimensional arrays, but the application does not compile nor execute.

Your program output should look like the sample output provided in the "Debug and Fix a 2-Dimensional Array Java Console Application Instructions" course file resource. **Full instructions for successfully completing this assignment are included in**

this resource. Use the submission template from the resources, [WeekXSolutionSubmissionTemplate.docx](#).

Your assignment will be scored on the following criteria:

1. Identify Java 2-dimensional array bugs in a program using development tools.
2. Code an application to fix bugs.
3. Test the application and document testing.
4. Explain the approach taken to complete this assignment and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

[2D Console Register for Course](#) | [Transcript](#)

U8A1_2DConsoleRegisterForCourse.zip

Debug and Fix a 2-Dimensional Array Java Console Application Instructions

Week X Solution Submission Template

Unit 9 >> Defining Classes and Instantiating Them into Objects

Introduction

In Unit 9, we begin our studies of Java object-oriented features. In this unit, you will learn about Java classes and how to define them. You will also learn about how to instantiate these classes into objects and how to use these objects.

You will develop a Java console application that defines classes and instantiates them into objects.

Learning Activities

u09s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about how to define Java classes and instantiate their objects:

- Chapter 9, "Objects and Classes," pages 321–364.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following URL for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java classes and objects in your application:

- Define Classes and Objects.
 - On the textbook custom Web site > VideoNotes > Chapter 9.
- Use Classes.
 - On the textbook custom Web site > VideoNotes > Chapter 9.
- Static vs. Instance.
 - On the textbook custom Web site > VideoNotes > Chapter 9.
- Data Field Encapsulation.
 - On the textbook custom Web site > VideoNotes > Chapter 9.
- The Fan Class.
 - On the textbook custom Web site > VideoNotes > Chapter 9.

Multimedia

You may view the following walkthroughs to help you prepare for the unit assignment:

- [UML Class Diagram Using Lucidchart Walkthrough.](#)
- [Adding Additional Classes to an Existing Java Project Walkthrough.](#)
- [Create a Java Program Using a UML Diagram Walkthrough.](#)

u09s2 - Discussion Preparation

Java is an object-oriented programming language that enables us to define classes and to instantiate them into objects. These objects then call each other's methods to implement the behavior of the application. The Unified Modeling Language (UML) is an object-oriented visual notation to document the design of object-oriented classes.

For this discussion, you will practice designing a Java class called Course, drawing a UML class diagram for the Course class, and then implementing the Course class in Java code. Review the Resources section to help you with this discussion.

Design a Java class called Course that has the following attributes:

1. code – a string field to store the course code (for example, IT1006).
2. creditHours – an int field to store the credit hours of the course (for example, 6).

Using one of the UML drawing tools available in the Resources, draw a UML class diagram for the Course class. Then implement your design of the Course class in Java.

You will be expected to provide a screenshot of the result of your work.

Course Resources

[Astah.](#)

[Dia](#)

[Giffly.](#)

[Lucidchart](#)

[lo.](#)

[Visual Paradigm.](#)

u09d1 - Design, Draw UML Class Diagram, and Implement a Java Course Class

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, you practiced designing a Java class called Course, drawing a UML class diagram for the Course class, and then implementing the Course class in Java code. Review the Resources section to help you with this discussion.

- Design a Java class called Course that has the following attributes:
 - code – a string field to store the course code (e.g., IT1006).
 - creditHours – an int field to store the credit hours of the course (e.g., 6).
- Draw a UML class diagram for the Course class. Then implement your design of the Course class in Java.
- Provide a screenshot of the result of your work.
- Explain, briefly, how you completed this exercise, the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

[lo.](#)

[Giffly.](#)

[Dia](#)

[Astah.](#)

[Lucidchart](#)

u09a1 - Define Java Classes and Instantiate Their Objects

In this assignment, you will design and code a simple Java application that defines a class, instantiates the class into a number of objects, and prints out the attributes of these objects in a specific way.

Your program output should look like the sample output provided in the "Define Java Classes and Instantiate their Objects Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template available in the resources, [WeekXSolutionSubmissionTemplate.docx](#).

Your assignment will be scored on the following criteria:

1. Design a program that meets basic object-oriented constructs requirements.
2. Code an application that exercises classes and objects.
3. Test the application and document testing.
4. Explain the approach taken to develop the application and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

[Define Java Classes and Instantiate Their Objects Instructions](#)

[Week X Solution Submission Template](#)

Unit 10 >> More Classes and Objects

Introduction

In Unit 10, you will continue to learn about Java object-oriented features. In particular, you will learn about class abstraction and method encapsulation, and how they are used to define classes that are easier to use and modify.

You will complete the development of a Java console application that uses classes and objects.

Learning Activities

u10s1 - Studies

Readings

Use your *Intro to Java Programming* textbook to read the following about Java object orientation and how to benefit from it in your own code:

- Chapter 10, "Object-Oriented Thinking," pages 365–408.

Internet Resources

In this unit you will be asked to access ancillary content from Pearson VideoNotes. Visit the following for instructions: [Intro to Java Programming Custom Web site](#).

Review these resources to learn about how to apply Java object orientation features in your application:

- The Loan Class.
 - On the textbook custom Web site > VideoNotes > Chapter 10.
- The StackOfIntegers Class.
 - On the textbook custom Web site > VideoNotes > Chapter 10.
- The String Class.
 - On the textbook custom Web site > VideoNotes > Chapter 10.
- The MyPoint Class.
 - On the textbook custom Web site > VideoNotes > Chapter 10.

Click on **OO Console Register for Course** to launch video (3 minutes).

Multimedia

You may view the following walkthrough to help you prepare for the unit assignment:

- [Array of Objects Walkthrough](#).

Course Resources

OO Console Register for Course

u10s2 - Discussion Preparation

Defining Java classes is one of two steps we practice when using objects. The second step is to instantiate Java classes into objects and to use these objects in our application.

In this discussion, you will practice instantiating an already defined Java class into objects and invoking the attributes of these objects.

Unzip the attached NetBeans project zip file (U10D1_InstantiateUseArrayOfObjects.zip) and load it into your NetBeans IDE.

The project defines two classes:

1. A Java class called Course (in Course.java) with certain attributes, and
2. Another class called U10D1_InstantiateUseArrayOfObjects (in U10D1_InstantiateUseArrayOfObjects.java). This class creates an array of Course objects and initializes their attributes to specific values and then calls the WriteCurrentRegistration() to print the current list of registered classes and their total credit hours. The WriteCurrentRegistration() still needs to be coded.

Code the WriteCurrentRegistration() method to meet the stated requirements specified in the project.

Provide a screenshot of the result of your work. Your screenshot should look like the "Instantiate Use Array of Objects Result" resource.

Course Resources

InstantiateUseArrayOfObjects.zip

Instantiate Use Array of Objects Result

u10d1 - Complete the Programming of a Simple Object-Oriented Application

Review the resources and instructions in the Discussion Prep Study before completing this discussion.

For this discussion, you practiced instantiating an already defined Java class into objects and invoking the attributes of these objects. Unzip the attached NetBeans project zip file (U10D1_InstantiateUseArrayOfObjects.zip) and load it into your NetBeans IDE.

The project defines two classes:

1. A Java class called Course (in Course.java) with certain attributes, and
2. Another class called U10D1_InstantiateUseArrayOfObjects (in U10D1_InstantiateUseArrayOfObjects.java). This class creates an array of Course objects and initializes their attributes to specific values and then calls the WriteCurrentRegistration() to print the current list of registered classes and their total credit hours. The WriteCurrentRegistration() still needs to be coded.

Code the WriteCurrentRegistration() method to meet the stated requirements specified in the project.

For this discussion:

- Provide a screenshot of the result of your work.
- Explain, briefly, how you completed this exercise, the algorithm you used (via pseudo code or other description tools), the major issues you faced, and how you solved them.
- Reflect on your experience and what you learned.

Response Guidelines

Comment on the responses of at least one other learner.

Course Resources

Undergraduate Discussion Participation Scoring Guide

InstantiateUseArrayOfObjects.zip

Instantiate Use Array of Objects Result

u10a1 - Complete the Programming of an Object-Oriented Console Application

In this assignment, you will complete the programming of two Java class methods in a console application that registers students for courses in a term of study. The application is written using the object-oriented features of the Java programming language. The application does compile and does run, but it does not produce the expected result as stated in its requirements.

Your program output should look like the sample output provided in the "Complete the Programming of an Object-Oriented Console Application Instructions" course file resource. **Full instructions for successfully completing this assignment are included in this resource.** Use the submission template available in the resources, `WeekXSolutionSubmissionTemplate.docx`.

Your assignment will be scored on the following criteria:

1. Design a program that meets object-oriented design requirements.
2. Code an application that exercises object-orientation.
3. Test the application and document testing.
4. Explain the approach taken to develop the application and the major decisions made.
5. Identify relevant fundamental constructs in the submitted program.
6. Communicates efficiently, effectively, and in an appropriate manner as an IT professional.

Course Resources

[OO Console Register for Course](#) | [Transcript](#)

Opening an Existing Project in NetBeans

OOConsoleRegisterForCourse.zip

Complete the Programming of an Object-Oriented Console Application Instructions

Week X Solution Submission Template