



Credit Hours: 3

Contact Hours: This is a 3-credit course, offered in accelerated format. This means that 16 weeks of material is covered in 8 weeks. The exact number of hours per week that you can expect to spend on each course will vary based upon the weekly coursework, as well as your study style and preferences. You should plan to spend 14-20 hours per week in each course reading material, interacting on the discussion boards, writing papers, completing projects, and doing research.

Faculty Information: Faculty contact information and office hours can be found on the faculty profile page.

COURSE DESCRIPTION AND OUTCOMES

Course Description:

This course provides students with the skills needed to become a java-oriented programmer. Students will learn to program applications using discrete structures and developing programs that access and update stored information from local databases and servers. Students will also learn the underlying features and use of programming language translation and static program analysis including run-time components such as memory management in different operating system environments.

Course Overview:

The course will introduce students to fundamental programming concepts using the Java Programming language. The course will start out by providing an overview of appropriate software development processes required to analyze and solve a problem using a process-based approach. Student will then be introduced to the basics of object-oriented programming concepts using the Java programming language. The concepts that will be learned included an understanding of basic data types, programming control structures, basic data structures, and algorithmic processes. Specifically, students will gain an understanding of developing programs with a focus on design structure, decision-making, utilizing loops, arrays, and objects to create a software program.

Course Learning Outcomes:

1. Demonstrate the use of algorithms, pseudo coding, and flowcharting as it relates to the problem-solving process.
2. Distinguish between basic data types, steps, and properties of programming languages.
3. Demonstrate how to implement fundamental data structures in programming.
4. Demonstrate the use of relational comparison operators and their precedence as it relates to a given problem.
5. Create appropriate sequence, selection, and repetition structures for a given programming assignment.
6. Demonstrate the use of arrays and arraylists.

7. Demonstrate how to apply programming constructs for reading and writing to text files, including basic file operations and sequential file processing.
8. Explain and identify concepts related to object-oriented programming.
9. Implement object-oriented classes that include appropriate class attributes and methods.

PARTICIPATION & ATTENDANCE

Prompt and consistent attendance in your online courses is essential for your success at CSU-Global Campus. Failure to verify your attendance within the first 7 days of this course may result in your withdrawal. If for some reason you would like to drop a course, please contact your advisor.

Online classes have deadlines, assignments, and participation requirements just like on-campus classes. Budget your time carefully and keep an open line of communication with your instructor. If you are having technical problems, problems with your assignments, or other problems that are impeding your progress, let your instructor know as soon as possible.

COURSE MATERIALS

Required:

- Lysecky, R., & Lizarraga, A. (2017). *Programming in Java with ZyLabs*. Los Gatos, CA: Zyante, Inc.
- The Latest Eclipse IDE (or equivalent Java development tool), Eclipse IDE for Java Developers (2018). Downloadable from <http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/>

***NOTE:** All non-textbook required readings and materials necessary to complete assignments, discussions, and/or supplemental or required exercises are provided within the course itself. Please read through each course module carefully.*

COURSE SCHEDULE

Due Dates

The Academic Week at CSU-Global begins on Monday and ends the following Sunday.

- **Discussion Boards:** The original post must be completed by Thursday at 11:59 p.m. MT and peer responses posted by Sunday at 11:59 p.m. MT. Late posts may not be awarded points.
- **Opening Exercises:** Take the Opening Exercise before reading each week's content to see which areas you will need to focus on. You may take these exercises as many times as you need. The Opening Exercises will not affect your final grade.
- **Mastery Exercises:** Students may access and retake Mastery Exercises through the last day of class until they achieve the scores they desire.
- **Critical Thinking:** Assignments are due Sunday at 11:59 p.m. MT.

WEEKLY READING AND ASSIGNMENT DETAILS

Module 1

Readings

- Chapter 1 in *Programming in Java with ZyLabs*
- Oracle. (2017). *The Java tutorials*. Retrieved from <http://www.oracle.com/technetwork/java/javase/java-tutorial-downloads-2005894.html>
- Oracle. (n.d.). *What is Java?* Retrieved from https://www.java.com/en/download/faq/whatis_java.xml
- Download: Eclipse IDE for Java Developers (2018). Retrieved from <http://www.eclipse.org/downloads/packages/release/photon/r/eclipse-ide-java-developers>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (50 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: List Individual's Name and Address Information

Demonstrate understanding of basic programming concepts by:

1. Developing pseudocode that prints the following information for a fictional person:
 - a. First name
 - b. Last name
 - c. Street address
 - d. City
 - e. Zip code.

2. Creating a simple java application that will print the items listed above on individual lines. Compile and submit your pseudocode, source code, and screenshots of the application executing the application and the results in a single document.

Option #2: List Restaurant Name and Address Information

Demonstrate understanding of basic programming concepts by:

1. Developing pseudocode that prints the following information for a restaurant in your city:
 - a. Restaurant name
 - b. Business address
 - c. City, State, Zip code
 - d. Phone.

2. Creating a simple java application that will print the items listed above in the same format. Compile and submit your pseudocode, source code, and screenshots of the application executing the application and the results in a single document.

Mastery Exercise (10 points)

Module 2

Readings

- Chapter 2 in *Programming in Java with ZyLabs*
- This course requires Java SE 8 JDK. Use the link below to download the correct version. Be sure to find the latest version of SE 8 which reflects the textbook content. Java SE 8 Update: <http://java.oracle.com/javase>
- For other operating system requirements, see <http://java.com/en/download/help/>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Calculate Miles per Gallon

Develop a Java application that provides program output in a logical manner and incorporates appropriate data types. Similar to zyBooks examples in Chapter 2, your program should prompt a user to enter a car brand, model, year, starting odometer reading, ending odometer reading, and gallons used. The output should include this information along with the estimated miles per gallon consumed by the vehicle in the format MPG: *your calculation*. Print each on separate lines with the appropriate labels (example, MPG: 25) Submit your source code and screenshots of the application executing with output in a single document.

Option #2: Calculate Data Usage

Develop a Java application that provides program output in a logical manner and incorporates appropriate data types. Similar to zyBooks examples in Chapter 2, your program should prompt a user to enter a phone provider (AT&T, Provider, or Verizon, for example), phone model monthly plan minutes, monthly data limit, actual current minute usage, and actual current data usage. Then calculate the remaining balances for minutes and data usage. Print each on separate lines with appropriate labels (example, Model: iPhone 8). Submit your source code and screenshots of the application executing with output in a single document.

Mastery Exercise (10 points)

Module 3

Readings

- Chapter 3 in *Programming in Java with ZyLabs*
- Oracle. (2015). *Variables*. Retrieved from <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>
- Oracle. (2015). *Operators*. Retrieved from <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/operators.html>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Calculate Average Withholding

Create a program that will calculate the weekly average tax withholding for a customer given the following weekly income guidelines:

- Income less than \$500: tax rate 10%
- Incomes greater than/equal to \$500 and less than \$1500: tax rate 15%
- Incomes greater than/equal to \$1500 and less than \$2500: tax rate 20%
- Incomes greater than/equal to \$2500: tax rate 30%.

Submit your source code and screenshots of the application executing the application and the results in a single document.

Option #2: Calculate Grocery Bill

Create a program that can be used to calculate the average weekly and monthly grocery bill for a family of four:

- Prompt the user for the coupon amount as a decimal (example, .10).
- Ensure the value is set to 10% if the value exceeds 100% or is less than or equal to zero.
- Prompt the user for weeks 1-4 grocery bills.
- Calculate the monthly and weekly average for groceries.
- Display monthly total and weekly average without the coupon.
- Display monthly total and weekly average with the coupon.

Submit your source code and screenshots of the application executing the application and the results in a single document.

Mastery Exercise (10 points)

Module 4

Readings

- Chapter 4 in *Programming in Java with ZyLabs*
- Oracle. (2015). The for statement. Retrieved from <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>
- Oracle. (2015). The while and do-while statements. Retrieved from <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Looping Construct with Floating Point Numbers

Write a program that utilizes a while-loop to read a set of five floating-point values from user input. Include code to prevent an endless loop. Ask the user to enter the values, then print the following data:

- Total
- Average
- Maximum
- Minimum
- Interest on total at 20%.
-

Submit your source code and screenshots of the application executing the application and the results in a single document.

Option #2: Capture Grade Statistics Using For-loop

Write a program that will provide important statistics for the grades in a class. The program will utilize a for-loop to read ten floating-point grades from user input. Include code to prevent an endless loop. Ask the user to enter the values, then print the following data:

- Average

- Maximum
- Minimum.

Submit your source code and screenshots of the application executing the application and the results in a single document.

Mastery Exercise (10 points)

Module 5

Readings

- Chapters 6 & 13 in Programming in Java with ZyLabs
- Oracle. (2015). Arrays. Retrieved from <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>
- Oracle. (2015). The list interface. Retrieved from <https://docs.oracle.com/javase/tutorial/collections/interfaces/list.html>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Get Weekly Temperatures

Develop a Java program that will store data in the form of daily average temperatures for one week. Store the day and average temperature in two different arraylists. Your program should prompt the user for the day of the week (Monday through Sunday) and display both the day and temperature for each day. If “week” is entered, the output for your program should provide the temperature for each day and the weekly average. *Use the looping and decision constructs in combination with the arrays to complete this assignment.*

Submit your source code and screenshots of the application executing the application and the results in a single document.

Option #2: Get Monthly Temperatures

Develop a Java program that will store data in the form of monthly temperatures for a year. Store the month and temperature in two different arrays. Your program should prompt the user for the month to be viewed and display both the month and average temperature. If “year” is entered, the output for your program should provide the temperature for each month along with the yearly average as well as the highest and lowest monthly averages. *Use the looping and decision constructs in combination with the arrays to complete this assignment.*

Submit your source code and screenshots of the application executing the application and the results in a single document.

Mastery Exercise (10 points)

Module 6

Readings

- *Chapter 5 in Programming in Java with ZyLabs*
- *Chapter 7 in Programming in Java with ZyLabs - Sections 7.13 to 7.15*
- Oracle. (2015). Defining methods. Retrieved from <http://docs.oracle.com/javase/tutorial/java/javaOO/methods.html>
- Oracle. (2015). Passing information to a method or constructor. Retrieved from <http://docs.oracle.com/javase/tutorial/java/javaOO/arguments.html>
- Oracle. (2015). Exceptions. <https://docs.oracle.com/javase/tutorial/essential/exceptions/index.html>

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Create a Method with a Scanner Input

Write a program with a method that will accept as input a scanner object. Prompt the user for three individual string input values (e.g., Red, Blue, and Green). Use Try..Catch error handling. The method should print the contents of the user input and return a string value to the calling method indicating “Success” or “Failure” based on the program flow through the try..catch error handling (see Chapter 13 in the text). Print the return message. Call the method from the main program to demonstrate its functionality.

Submit your source code and screenshots of the application executing the application and the results in a single document.

Option #2: Create a Method to Return a String Array

Write program with a method that returns an array. The method should accept as input a comma-delimited string with three values from a user. The array should store each value in a different element. Use Try..Catch error handling and print any failure messages, or print success from within method if execution is successful (see Chapter 13 in the text). Call the method from the main method of the program to demonstrate its functionality by looping through the array and printing the individual values.

Submit your source code and screenshots of the application executing the application and the results in a single document.

Portfolio Milestone (50 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Portfolio Option #1

Submit a document with methods for your automobile class, and pseudo code indicating functionality of each method.

Example:

```
public String RemoveVehicle(String autoMake, String autoModel, String autoColor, int autoYear)
if
    values entered match values stored in private variables
    remove vehicle information
else
    return message indicating mismatch
```

Post questions to “General Course Questions” for areas where greater understanding is required.

Option #2: Portfolio Option #2

Submit a document with methods for your home class, and pseudo code indicating functionality of each method.

Example:

```
public String RemoveHome(String homeModel, String homeAddress, int homeZipCode)
```

```
if
```

```
    values entered match values stored in private variables  
    remove home information
```

```
else
```

```
    return message indicating mismatch
```

Post questions to “General Course Questions” for areas where greater understanding is required.

Mastery Exercise (10 points)

Module 7

Readings

- Chapters 8 & 9 in *Programming in Java with ZyLabs*
- Oracle. (2015). BASIC I/O. Retrieved from <https://docs.oracle.com/javase/tutorial/essential/io/index.html>
- Marian, C. (2018) Java memory management. Retrieved from <https://dzone.com/articles/java-memory-management>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Module 8

Readings

- Chapter 7 in *Programming in Java with ZyLabs*
- Oracle. (2015). Classes. Retrieved from <https://docs.oracle.com/javase/tutorial/java/javaOO/classes.html>
- Oracle. (2015). Objects. Retrieved from <https://docs.oracle.com/javase/tutorial/java/javaOO/objects.html>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Project (350 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Portfolio Option #1

Your Portfolio Project for CSC320 will consist of three components:

1. Program corrections: Make the appropriate corrections to all the programming assignments submitted as Critical Thinking assignments from Modules 1-6. You will need to submit the

- programs along with the carefully outlined corrections needed in order for the programs to run correctly.
2. Lessons learned reflection: Create a 2-3-page summary that outlines the lessons learned in this Programming I course.
 3. Final program: Create a final program that meets the requirements outlined below.

Final Program Requirements

Create an automobile class that will be used by a dealership as a vehicle inventory program. The following attributes should be present in your automobile class:

- private string make
- private string model
- private string color
- private int year
- private int mileage.

Your program should have appropriate methods such as:

- default constructor
- parameterized constructor
- add a new vehicle method
- list vehicle information (return string array)
- remove a vehicle method
- update vehicle attributes method.

All methods should include try..catch constructs. Except as noted all methods should return a success or failure message (failure message defined in "catch").

1. Create an additional class to call your automobile class (e.g., Main or AutomobileInventory). Include a try..catch construct and print it to the console any errors.
2. Call automobile class with parameterized constructor (e.g., "make, model, color, year, mileage").
 - a. Then call the method to list the values. Loop through the array and print to the screen.
3. Call the remove vehicle method to clear the variables.
 - a. Print the return value.
4. Add a new vehicle.
 - a. Print the return value.
 - b. Call the list method and print the new vehicle information to the screen.
5. Update the vehicle.
 - a. Print the return value.
 - b. Call the listing method and print the information to the screen.
6. Display a message asking if the user wants to print the information to a file (Y or N).
 - a. Use a scanner to capture the response. If "Y", print the file to a predefined location (e.g., C:\Temp\Autos.txt). Note: you may want to create a method to print the information in the main class.
 - b. If "N", indicate that a file will not be printed.

Your final program submission materials must include your source code and screenshots of the application executing the application and the results.

Compile your Module 1-6 programs with corrections, lessons learned reflection, and final program course code and application screenshots into a single document. Submit your completed Portfolio Project by the posted due date.

Option #2: Portfolio Option #2

Your Portfolio Project for CSC320 will consist of three components:

1. Program corrections: Make the appropriate corrections to all the programming assignments submitted as Critical Thinking assignments from Modules 1-6. You will need to submit the programs along with the carefully outlined corrections needed in order for programs to run correctly.
2. Lessons learned reflection: Create a 2-3-page summary that outlines the lessons learned in this Programming I course.
3. Final program: Create a final program that meets the requirements outlined below.

Final Program Requirements

Create a home inventory class that will be used by a national builder to maintain inventory of available houses in the country. The following attributes should be present in your home class:

- private int square_feet
- private string address
- private string city
- private string state
- private int zip_code
- private string Model_name
- private string sale_status (sold, available, or under contract).

Your program should have appropriate methods such as:

- constructor
- add a new home
- remove a home
- update home attributes.

All methods should include try..catch constructs. Except as noted, all methods should return a success or failure message (failure message defined in "catch").

1. Create an additional class to call your home class (e.g., Main or HomeInventory). Include a try..catch construct and print it to the console any errors.
2. Call home class with parameterized constructor (e.g., "square_feet, address, city, state, zip_code, Model_name, sale_status").
 - a. Then, call the method to list the values. Loop through the array and print to the screen
3. Call the remove home method to clear the variables:
 - a. Print the return value.
4. Add a new home.
 - c. Print the return value.
 - d. Call the list method and print the new home information to the screen.
5. Update the home (change the sale status).

- c. Print the return value.
 - d. Call the listing method and print the information to the screen.
6. Display a message asking if the user wants to print the information to a file (Y or N).
- c. Use a scanner to capture the response. If “Y”, print the file to a predefined location (e.g., C:\Temp\Home.txt). Note: you may want to create a method to print the information in the main class.
 - d. If “N”, indicate that a file will not be printed.

Your final program submission materials must include your source code and screenshots of the application executing the application and the results.

Compile your Module 1-6 programs with corrections, lessons learned reflection, and final program course code and application screenshots.

COURSE POLICIES

Grading Scale	
A	95.0 – 100
A-	90.0 – 94.9
B+	86.7 – 89.9
B	83.3 – 86.6
B-	80.0 – 83.2
C+	75.0 – 79.9
C	70.0 – 74.9
D	60.0 – 69.9
F	59.9 or below

Course Grading

20% Discussion Participation
0% Opening Exercises
8% Mastery Exercises
37% Critical Thinking Assignments
35% Final Portfolio Project

IN-CLASSROOM POLICIES

For information on late work and incomplete grade policies, please refer to our [In-Classroom Student Policies and Guidelines](#) or the Academic Catalog for comprehensive documentation of CSU-Global institutional policies.

Academic Integrity

Students must assume responsibility for maintaining honesty in all work submitted for credit and in any other work designated by the instructor of the course. Academic dishonesty includes cheating, fabrication, facilitating academic dishonesty, plagiarism, reusing /repurposing your own work (see CSU-Global Guide to Writing & APA for percentage of repurposed work that can be used in an assignment), unauthorized possession of academic materials, and unauthorized collaboration. The CSU-Global Library provides information on how students can avoid plagiarism by understanding what it is and how to use the Library and internet resources.

Citing Sources with APA Style

All students are expected to follow the CSU-Global Guide to Writing & APA when citing in APA (based on the most recent APA style manual) for all assignments. A link to this guide should also be provided within most assignment descriptions in your course.

Disability Services Statement

CSU-Global is committed to providing reasonable accommodations for all persons with disabilities. Any student with a documented disability requesting academic accommodations should contact the Disability Resource Coordinator at 720-279-0650 and/or email ada@CSUGlobal.edu for additional information to coordinate reasonable accommodations for students with documented disabilities.

Netiquette

Respect the diversity of opinions among the instructor and classmates and engage with them in a courteous, respectful, and professional manner. All posts and classroom communication must be conducted in accordance with the student code of conduct. Think before you push the Send button. Did you say just what you meant? How will the person on the other end read the words?

Maintain an environment free of harassment, stalking, threats, abuse, insults, or humiliation toward the instructor and classmates. This includes, but is not limited to, demeaning written or oral comments of an ethnic, religious, age, disability, sexist (or sexual orientation), or racist nature; and the unwanted sexual advances or intimidations by email, or on discussion boards and other postings within or connected to the online classroom. If you have concerns about something that has been said, please let your instructor know.