

CIS 088: INTRODUCTION TO PROGRAMMING USING JAVA

Originator

eramirez

Co-Contributor(s)**Name(s)**

Marhuenda-Donate, Felix

Justification / Rationale

Java is one of the top programming languages used in businesses and as such, there is a need to teach students how to program in Java to be competitive in the workforce. Java is used for server-side development, front-end development and is the default programming language used in Android development. Java has a large backing from companies such as Oracle, Amazon, and Google to name a few.

Effective Term

Fall 2022

Credit Status

Credit - Degree Applicable

Subject

CIS - Computer Information Systems

Course Number

088

Full Course Title

Introduction To Programming Using Java

Short Title

INTRO TO JAVA

Discipline**Disciplines List**

Computer Information Systems (Computer network installation, microcomputer technology, computer applications)

Modality

Face-to-Face

100% Online

Hybrid

Catalog Description

This course is an introduction to the fundamentals concepts of computer programming using Java. The course focuses on learning the basic concepts of program design, problem-solving, data structures, and programming logic. The course heavily relies on hands-on experience using Java and a modern integrated development environment (IDE) such as but not limited to Eclipse or IntelliJ.

Schedule Description

This course is an introduction to the fundamentals concepts of computer programming using Java. The course focuses on learning the basic concepts of program design, problem-solving, data structures, and programming logic. The course heavily relies on hands-on experience using Java and a modern integrated development environment (IDE) such as but not limited to Eclipse or IntelliJ.

Lecture Units

2

Lecture Semester Hours

36

Lab Units

1

Lab Semester Hours

54

In-class Hours

90

Out-of-class Hours

72

Total Course Units

3

Total Semester Hours

162

Required Text and Other Instructional Materials**Resource Type**

Book

Open Educational Resource

Yes

Author

Allen B Downey and Chris Mayfield

Title

Think Java: How to Think Like a Computer Scientist

Edition

2nd Edition

City

Needham, MA

Publisher

Green Tee Press

Year

2020

College Level

Yes

Flesch-Kincaid Level

12

ISBN #

978-1492072508

Resource Type

Book

Author

Aristides S. Bouras

Title

Java and Algorithmic Thinking for the Complete Beginner

Edition

2

Publisher

Independently published

Year

2019

College Level

Yes

ISBN #

978-1698811239

Class Size Maximum

35

Course Content

1. Program design tools and programming environments
2. Using Documentation
3. Software life cycle including design, development, styles, documentation, testing, and maintenance.
4. Variables & Data Types
5. Strings
6. User input and output
7. Control Flow and Conditionals
8. Functions/Methods/Procedures
9. Loops
10. Arrays
11. Lists
12. UML to Java classes
13. Objects & Classes
14. Simulation Design
15. Algorithm Design
16. Serverless computing using Java functions

Lab Content

Lab content will be covered through individual and/or group activities. These activities are project-based and business-oriented that cater to the course content.

Reinforce the following concepts:

- Program design
- Program development lifecycle
- Requirements determinants and analysis
- Modular design
- Techniques for modeling program structures
- Programming concepts
- Variables
- Literals
- Types
- Expressions
- Procedures
- Functions
- Parameters
- Operators and operations
- Decision logic
- Looping
- Sub-procedures
- Passing parameters

- Coding
- Unit testing
- Control structures

Course Objectives

| | Objectives |
|--------------|--|
| Objective 1 | Describe the basic components of the Java software development environment. |
| Objective 2 | Develop programs that use primitive data types, data structures, and classes provided by the programming language. |
| Objective 3 | Construct Java programs that use built-in functions provided by the programming language. |
| Objective 4 | Design, implement, test, and debug a Java program that uses basic computation as well as sequencing and simple I/O. |
| Objective 5 | Design, implement, test, and debug a Java program that uses control flow. |
| Objective 6 | Design, implement, test, and debug a Java program that uses methods/functions as well as parameters to organize the flow of a program. |
| Objective 7 | Design, implement, test, and debug a Java program that uses iteration. |
| Objective 8 | Design the basic structure of a Java class with instance variables/fields, and methods from a UML diagram and write the implementation details for the methods to produce a working program. |
| Objective 9 | Define and use classes and methods to implement algorithms. |
| Objective 10 | Create pseudocode that will provide a high-level solution to a problem statement. |

Student Learning Outcomes

| | Upon satisfactory completion of this course, students will be able to: |
|-----------|--|
| Outcome 1 | Develop Java programs that use sequencing, control flow, iteration, and method/function techniques to organize code with proper documentation of class and method/functions. |
| Outcome 2 | Create classes, instance variables, and methods with the proper Java modifiers as well as the implementation details from a UML class diagram and a problem statement. |
| Outcome 3 | Construct programs that utilize object-oriented programming techniques such as information hiding and uses classes and objects provided by the Java language. |

Methods of Instruction

| Method | Please provide a description or examples of how each instructional method will be used in this course. |
|------------------------------------|--|
| Lecture | Presentation of class lectures/discussions/demonstrations to model and explain the fundamental concepts of programming, algorithms, and problem solving. |
| Laboratory | Students will create Java programs to solve real-life business needs. |
| Demonstration, Repetition/Practice | Students will have to incorporate instructor feedback on each of their Java lab activities and resubmit for grading or incorporate in future assignments. |
| Technology-based instruction | Use of Integrated Development Environment (IDE) for software development. |
| Activity | Activities focused on addressing areas of improvement in the fundamental concepts of Object-Oriented programming, such as elevator programming and algorithm flow. |
| Discussion | Students will discuss assigned Business/CIS cases. |
| Collaborative/Team | Work group consisting of developing an algorithm to fulfill business needs. |
| Participation | Students will participate in discussion regarding best practices in object-oriented programming. |

Methods of Evaluation

| Method | Please provide a description or examples of how each evaluation method will be used in this course. | Type of Assignment |
|--|--|---------------------|
| Written homework | Provide pseudocode that will solve a given problem statement. Approximately 15 hours outside of class. | Out of Class Only |
| Group activity participation/observation | Work group consisting of developing an algorithm to fulfill business needs. | In Class Only |
| Product/project development evaluation | Develop Java programs based on group work algorithms. | In Class Only |
| Mid-term and final evaluations | Students will complete quizzes and exams throughout the course. | In Class Only |
| Reading reports | Read Java documentation. Approximately 15 hours. | Out of Class Only |
| Student participation/contribution | Students will participate in discussions regarding assigned Business/CIS cases. | In and Out of Class |
| Laboratory projects | Integrated Development Learning Environment (IDLE) lab simulations. Build Java programs. | In and Out of Class |
| Presentations/student demonstration observations | Students will demonstrate to the class the function of each line of their Java source code. | In Class Only |
| Computational/problem-solving evaluations | Students will be required to create a Java program that solves a problem. Approximately 20 hours. | Out of Class Only |
| Product/project development evaluation | Students will evaluate the presented code and provide an efficiency report. Approximately 15 hours. | In and Out of Class |
| Tests/Quizzes/Examinations | The course will contain various types of examinations. Approximately 10 hours. | In and Out of Class |

Assignments
Other In-class Assignments

- Program exercises demonstrating each new concept.
- Comprehensive programming exercises demonstrating multiple concepts.
- Textbook readings, internet resources in preparation for quizzes, and in-class hands-on demonstrations.
- Use IF statements or Case structure to make decisions based on the test of a condition.
- Break down a complicated problem (calculation) into smaller units. Solve each unit using a method or function.
- Use loop structures to repeat an operation multiple times.
- Create console applications that utilize the concepts above.
- Create console applications that utilize selection control to create a 'Menu Driven' application.

Other Out-of-class Assignments

Critical Thinking: Students are expected to spend a minimum of three hours per week on outside programming assignments that involve logic and problem-solving skills.

Readings: Textbooks, Supporting references, Websites, Periodicals, course handouts, and videos. Students will be asked to read white papers, Internet-based information or videos describing algorithms in order to discuss the topic of algorithm creation during class.

Writing: Analyses, Reports, Responses/Reactions, Critiques.

Students will be asked to write a detailed list of pseudocode for describing a simple process and evaluating how that would translate to the logic of object-oriented programming and using methods to organize code.

Grade Methods

Letter Grade Only

Distance Education Checklist

Include the percentage of online and on-campus instruction you anticipate.

Online %

100

On-campus %

100

Lab Courses**How will the lab component of your course be differentiated from the lecture component of the course?**

The lecture portion of the course will consist of readings and activities (e.g. pseudocode, algorithms) that will build knowledge of the Java programming language and program structure.

Students will be writing Java programs during the lab component of the course.

From the COR list, what activities are specified as lab, and how will those be monitored by the instructor?

Activities that involve the students writing Java programs will be specified as lab assignments. The course will contain Canvas weekly assignments that require students to complete working Java programs that achieve a particular goal.

How will you assess the online delivery of lab activities?

Lab activities will require that students install an Integrated Development Environment (IDE) to create their software; this has been the primary issue in all our other programming courses. We have developed a robust module centered around the IDE so students can complete their lab activities.

We realized that IDE was the issue because fewer students turned in their lab coding assignments. We will use these types of techniques to assess our delivery methods.

Instructional Materials and Resources**If you use any other technologies in addition to the college LMS, what other technologies will you use and how are you ensuring student data security?**

We will be using IDEs that is open source to develop code. It will not contain or traffic student data.

If used, explain how specific materials and resources outside the LMS will be used to enhance student learning.

The IDE is necessary for the development of all computer programs.

Effective Student/Faculty Contact**Which of the following methods of regular, timely, and effective student/faculty contact will be used in this course?****Within Course Management System:**

- Discussion forums with substantive instructor participation
- Online quizzes and examinations
- Private messages
- Regular virtual office hours
- Timely feedback and return of student work as specified in the syllabus
- Video or audio feedback
- Weekly announcements

External to Course Management System:

- Direct e-mail
- E-portfolios/blogs/wikis
- Posted audio/video (including YouTube, 3cm mediasolutions, etc.)
- Synchronous audio/video
- Teleconferencing
- Telephone contact/voicemail

Briefly discuss how the selected strategies above will be used to maintain Regular Effective Contact in the course.

There will be weekly discussions regarding topics related to the course with appropriate instructor participation. Students will create and upload Java programs of their creation. These activities will receive appropriate instructor feedback.

Other Information**Provide any other relevant information that will help the Curriculum Committee assess the viability of offering this course in an online or hybrid modality.**

This course is really meant as a sister course to our CIS 86 (Visual Basic) course but with a different language. We have been successfully delivering CIS 86 online since the inception of COD's online program.

There will be scheduled lab hours with the instructor for the course in order to facilitate student work.

Comparable Transfer Course Information

University System

CSU

Campus

CSU San Bernardino

Course Number

IST 282

Course Title

Business Systems 1

Catalog Year

2018

MIS Course Data

CIP Code

11.0201 - Computer Programming/Programmer, General.

TOP Code

070710 - Computer Programming

SAM Code

B - Advanced Occupational

Basic Skills Status

Not Basic Skills

Prior College Level

Not applicable

Cooperative Work Experience

Not a Coop Course

Course Classification Status

Credit Course

Approved Special Class

Not special class

Noncredit Category

Not Applicable, Credit Course

Funding Agency Category

Not Applicable

Program Status

Program Applicable

Transfer Status

Transferable to CSU only

General Education Status

Y = Not applicable

Support Course Status

N = Course is not a support course

Allow Audit

No

Repeatability

No

Materials Fee

No

Additional Fees?

No

Approvals**Curriculum Committee Approval Date**

3/17/2022

Academic Senate Approval Date

3/24/2022

Board of Trustees Approval Date

4/22/2022

Chancellor's Office Approval Date

4/30/2022

Course Control Number

CCC000631281