Cerro Coso College

Course Outline of Record Report

10/13/2021

CSCIC267 : Introduction to JAVA Programming

General Information

Author:	-
Course Code (CB01) :	CSCIC267
Course Title (CB02) :	Introduction to JAVA Programming
Department:	Business Information Technolog
Proposal Start:	Fall 2013
TOP Code (CB03) :	(0702.00) Computer Information Systems
SAM Code (CB09) :	Clearly Occupational
Distance Education Approved:	Yes
Course Control Number (CB00) :	CCC000232355
Curriculum Committee Approval Date:	10/31/2014
Board of Trustees Approval Date:	12/18/2014
External Review Approval Date:	02/09/2015
Course Description:	This course is designed to introduce techniques and principles of problem solving using computer systems with the Java computer programming language. The development of applications and applets with Java is covered using object-oriented programming techniques.
Submission Type:	New Course
Author:	No value

Faculty Minimum Qualifications

Master Discipline Preferred:	No value
Alternate Master Discipline Preferred:	• Computer Science • Computer Science
Bachelors or Associates Discipline Preferred:	• Computer Information Systems (Computer network installation, microcomputer technology, computer applications)
Additional Bachelors or Associates Discipline Preferred:	No value

Course Development Options

Basic Skills Status (CB08)

Course is not a basic skills course.

Course Special Class Status (CB13)

Course is not a special class.

Grade Options

- Letter Grade Methods
- Pass/No Pass

☐ Allow Students to Gain Credit by Exam/Challenge

Allowed Number of Retakes

0

Course Prior To College Level (CB21)

Not applicable.

Rationale For Credit By Exam/Challenge

No value

Retake Policy Description

Type:|Non-Repeatable Credit

☑ Allow Students To Audit Course

Course Support Course Status (CB26)

No value

Associated Programs

☐ Course is part of a program (CB24)

Associated Program

No value

Award Type

No value

Active

Transferability & Gen. Ed. Options

Course General Education Status (CB25)

No value

Transferability

Transferable to both UC and CSU

Transferability Status

Approved

Units and Hours:

Summary

Minimum Credit Units (CB07)	3
Maximum Credit Units (CB06)	3
Total Course In-Class (Contact) Hours	90
Total Course Out-of-Class Hours	72
Total Student Learning Hours	162
Faculty Load	0

Credit / Non-Credit Options

Course Credit Status (CB04)

Credit - Degree Applicable

Course Non Credit Category (CB22)

Credit Course.

Non-Credit Characteristic

No Value

Course Classification Status (CB11)

Credit Course.

Funding Agency Category (CB23)

Not Applicable.

☐ Cooperative Work Experience Education Status (CB10)

☐ Variable Credit Course

Weekly Student Hours

	In Class	Out of Classs
Lecture Hours	2	4
Laboratory Hours	3	0
Activity Hours	0	0

Course Student Hours

Course Duration (Weeks)	18
Hours per unit divisor	0
Course In-Class (Contact) Hours	
Lecture	0
Laboratory	0
Activity	0
Total	90
Course Out-of-Class Hours	
Lecture	0
Laboratory	0
Activity	0
Total	72

Time Commitment Notes for Students

No value

Faculty Load

Extra Duties: 0

Faculty Load: 0

Units and Hours: - Weekly Specialty Hours

Activity Name	Type	In Class	Out of Class
No Value	No Value	No Value	No Value

Pre-requisites, Co-requisites, Anti-requisites and Advisories

Prerequisite

MATHC055 - Intermediate Algebra

Students are expected to solve many complex word problems based on algebraic construction and work with proportions, ratios, exponents, and percentages. In addition, students are expected to create, understand, and debug programs that read and interpret tabular data as well as work with complex formulas and logic. These are skills that students learn in the MATH C055 course.

AND

Prerequisite

CSCIC101 - Introduction to Computer Information Systems

Students need to be able to install their own software and understand what memory is, how to zip and unzip files, how to save and find their files, and how a computer works. This material is covered in the CSCI C101 course.

Entrance Skills

Entrance Skills	Description
No value	No value

Limitations on Enrollment

Limitations on Enrollment	Description
No value	No value

Specifications

Methods of Instruction

Methods of Instruction	Lecture
Rationale	PowerPoint lectures demonstrating the logic, syntax and use of programming controls, properties, structures, and classes.
Methods of Instruction	Written work
Rationale	No value
Methods of Instruction	Project-based learning
Rationale	No value
Methods of Instruction	Outside reading
Rationale	No value

Methods of Instruction	Instruction through examination or quizzing
Rationale	No value

Methods of Instruction	Laboratory
Rationale	No value

Methods of Instruction	Demonstration
Rationale	No value

Methods of Instruction	Discussion
Rationale	No value

Assignments

A. Chapter reading (Example: Reading the assigned chapters from the textbook based on the topics for the week).
B. Coding assignments (Example: Write and test program coding assignments).
C. Application design (Example: Design program flow and control for a variety of programming tasks).

Methods of Evaluation	Rationale
Final Exam	Final Exam demonstrating comprehensive mastery of material presented. Example: Multiple choice and essay question exam covering all concepts of the course.
Participation	Discussion Participation demonstrating understanding of Java concepts. Example: Discussion regarding when to use different control structures with examples.
Tests	Weekly Quizzes demonstrating understanding of new material presented. Example: Multiple choice question quiz covering the use of expressions.
Homework	Weekly Programming Assignments demonstrating mastery of new programming material. Example: Program coding assignment on use of control structures.
Tests	Midterm Exam demonstrating mastery of material of the first half of instruction. Example: Multiple choice and essay question exam covering variables, types, expressions, and methods.

Equipment

No Value

Textbooks

Author	Title	Publisher	Date	ISBN
	Malik, D. S . (2012) Java Programming: From Problem Analysis to Program Design , 5th , Cengage Learning			

Other Instructional Materials

Description	Software: Oracle. JDeveloper, 2014 ed. -This course requires a Java Development Environment. Students may choose this one or the one below. Both are free. This one can be found @ http://www.oracle.com/technetwork/developer-tools/jdev/overview/index.html
Author	
Citation	Introduction to JAVA Programming

Materials Fee

No

Learning Outcomes and Objectives

Course Objectives

No value

CSLOs

Define and apply the fundamentals, structure, logic, and syntax of Java programming. Expected SLO Performance: 70.0

Identify the terminology associated with object-oriented programming and Java. Expected SLO Performance: 70.0

Develop, design and code simple to moderate applications using Java. Expected SLO Performance: 70.0

Science Liberal Arts: Mathematics & Science AA Degree	Apply algebraic, graphical, numerical, and other methods to solve applied problems in the areas of mathematics, natural sciences, computer graphics, and computer animation.

Analyze and debug program code for logic, and run-time errors, and intended design accuracy. Expected SLO Performance: 70.0

Interpret and use strings, variables, repetition structures, selection, arrays, methods, inheritance, classes, and objects. Expected SLO Performance: 70.0

Outline

Course Outline

I. Overview of Java
A. Using an integrated developer environment
B. Writing and running a basic Java application.
II. Variables; types; and expressions
A. primitive types
1. Double Type
2. Int Type
3. Char Type
4. Boolean Type
B. Reference types

1. Reference variable initialization
2. The String Type
3. Input and output
III. Methods
A. non-void vs. void Methods
B. non-void Methods
C. Method design
D. Method libraries
E. instance methods
IV. Control Structures
A. Selective Execution
1. If Statement
2. Switch Statement
B. Repetitive Execution
1. While Statement
2. For Statement
3. Do Statement
V. Files and Exceptions
A. Managing Files
1. Opening a file
2. Reading from a file
3. Writing to a file
4. Closing a file
B. Exceptions
1. Throwing exceptions
2. Handling exceptions: try-catch blocks
3. Cleaning up exceptions: finally block
VI. Data Structures
A. Arrays
1. Arrays and memory
2. Multidimensional arrays
B. Linked List Class
C. Array List Class
VII. Object-Oriented Programming
A. Class Hierarchies
B. Bottom-Up vs. Top-Down Design
C. Event-Driven Programming
D. The main() Method

Lab Outline

I.Java Programming Overview
A.Using an integrated developer environment
B.Writing and running a basic Java application
II.Programming with Variables; Types; and Expressions
A.Primitive types
B.Reference types
C.Expressions
III.Programming with Control Structures
A.Selection
B.Repetition
IV.Working with Files and Exceptions
A.Managing Files
B.Exceptions
V.Working with Data Structures
A.Arrays
B.Linked List Class
C.Array List Class
VI.Fundamentals of Object-Oriented Programming
A.Class Hierarchies
B.Bottom-Up vs. Top-Down Design
C.Event-Driven Programming

Delivery Methods and Distance Education

Delivery Method: Please list all that apply -Face to face -Online (purely online no face-to-face contact) -Online with some required face-to-face meetings (“Hybrid”) -Online course with on ground testing -iTV – Interactive video = Face to face course with significant required activities in a distance modality -Other

Face 2 Face
Online
Hybrid
Interactive

Rigor Statement: Assignments and evaluations should be of the same rigor as those used in the on-ground course. If they are not the same as those noted in the COR on the Methods of Evaluation and out-of-class assignments pages, indicate what the differences are and why they are being used. For instance, if labs, field trips, or site visits are required in the face to face section of this course, how will these requirements be met with the same rigor in the Distance Education section?

All assignments in distance education courses (online, hybrid and iTV) of CSCI C267 are of the same rigor as those in the on-ground course, except that students in purely online sections will submit all of their assignments virtually. Instructor evaluatio

Effective Student-Instructor Contact: Good practice requires both asynchronous and synchronous contact for effective contact. List the methods expected of all instructors teaching the course. -Learning Management System -Discussion Forums -Moodle Message -Other Contact -Chat/Instant Messaging -E-mail -Face-to-face meeting(s) -Newsgroup/Discussion Board -Proctored Exam -Telephone -iTV -Interactive Video -Other (specify)

contact_moodle_forums
contact_moodle_message
contact_email
contact_face2face
contact_discussion
contact_itv

Software and Equipment: What additional software or hardware, if any, is required for this course purely because of its delivery mode? How is technical support to be provided?

No Value

Accessibility: Section 508 of the Rehabilitation Act requires access to the Federal government's electronic and information technology. The law covers all types of electronic and information technology in the Federal sector and is not limited to assistive technologies used by people with disabilities. It applies to all Federal agencies when they develop, procure, maintain, or use such technology. Federal agencies must ensure that this technology is accessible to employees and the public to the extent it does not pose an "undue burden". I am using -iTV—Interactive Video only -Learning management system -Publisher course with learning management system interface.

s508_itv
s508_moodle
s508_publisher

Class Size: Good practice is that section size should be no greater in distance ed modes than in regular face-to-face versions of the course. Will the recommended section size be lower than in on-ground sections? If so, explain why.

No Value