

Creating Java Programs with Greenfoot

Overview

This workshop engages students who understand basic programming concepts to create 2-D games using Java. Greenfoot is a free educational Java development environment (JDE) created at the University of Kent. Students will learn detailed object-oriented programming terminology and concepts while creating 2-D games in a fun and interactive environment.

Duration

- Recommended classroom instruction time: up to 16 hours
- Professional education credit hours for educators who complete Oracle Academy training: 16

Target Audiences

Educators

- Secondary school teachers who teach computer programming
- Secondary school teachers who seek to incorporate computing into their curriculum and awaken students' interest in computer science
- May also be suitable for club leaders and other general educators who wish to incorporate computer science into their curriculum

Students

- Students who wish start or further their Java programming experience and learn how to create 2-D games in a fun and interactive environment
- While Greenfoot is suitable for beginning learners, our experience shows it is most engaging for secondary school and early post-secondary students

Prerequisites

Required:

- Basic understanding of at least one programming language

Suggested:

- Oracle Academy Workshop - Getting Started with Java Using Alice

Suggested Next Courses

- Oracle Academy Course – Java Fundamentals

Lesson-by-Lesson Objectives

Getting Started With Greenfoot

- Download and install Greenfoot
- Describe the components of the Greenfoot development environment
- Create an instance of a class
- Describe classes and subclasses
- Recognize Java syntax used to correctly create a subclass

Using Methods, Variables and Parameters

- Define parameters and how they are used in methods
- Understand inheritance
- Describe the properties of an object
- Examine the purpose of a variable
- Discuss programming concepts and define terminology

Working with Source Code and Documentation

- Demonstrate source code changes to invoke methods programmatically

- Demonstrate source code changes to write an IF decision statement
- Describe a procedure to display object documentation

Developing and Testing an Application

- Demonstrate program testing strategies
- Recognize phases for developing a software application

Using Randomization and Understanding Dot Notation and Constructors

- Create randomized behaviors
- Define comparison operators
- Create IF-ELSE control statements
- Recognize and describe dot notation

Defining Methods

- Describe effective placement of methods in a super or subclass
- Simplify programming by creating and calling defined methods

Using Sound and Keyboard Control

- Write programming statements to include sound in a program
- Write programming statements to include keyboard movements in a program

Creating a World, Animating Actors, and Ending a Game

- Construct a world object using a constructor method
- Create an object using a constructor
- Define the purpose and syntax of a variable
- Recognize the syntax to define and test variables
- Write programming statements to end a game

Understanding Abstraction

- Define abstraction and provide an example of when it is used

Using Loops, Variables, and Strings

- Create a while loop in a constructor to build a world
- Describe an infinite loop and how to prevent one from occurring
- Use an array to store multiple variables used to create a world
- Create an expression using logic operators
- Describe the scope of a local variable in a method
- Use string variables to store and concatenate strings

Putting it All Together with Greenfoot

- Apply your Greenfoot knowledge to create a Java game

To search and register for events scheduled in your area, visit the [Academy events calendar](#).