

Unit 9: Inheritance

Content Area: **Template**
Course(s):
Time Period: **Full Year**
Length: **Full Year**
Status: **Published**

UNIT RATIONALE

Creating objects, calling methods on the objects created, and being able to define a new data type by creating a class are essential understandings before moving into this unit. One of the strongest advantages of Java is the ability to categorize classes into hierarchies through inheritance . Certain existing classes can be extended to include new behaviors and attributes without altering existing code. These newly created classes are called subclasses . In this unit, students will learn how to recognize common attributes and behaviors that can be used in a superclass and will then create a hierarchy by writing subclasses to extend a superclass. Recognizing and utilizing existing hierarchies will help students create more readable and maintainable programs.

ESSENTIAL QUESTIONS

How might the use of inheritance help in writing a program that simulates crops being grown in a virtual world?
How does inheritance make programs more versatile?

STANDARDS

NEW JERSEY STUDENT LEARNING STANDARDS: CONTENT AREA

SAVED

New Jersey (NJSL) - Grades 9-12 - Computer Science and Design Thinking (2020)

8.1.12.AP.1:

Design algorithms to solve computational problems using a combination of original and existing algorithms.

8.1.12.AP.2:

Create generalized computational solutions using collections instead of repeatedly using simple variables.

8.1.12.AP.3:

Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.

8.1.12.AP.4:

Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

8.1.12.AP.5:

Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

8.1.12.AP.6:

Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

CS.9-12.8.1.12.AP.1	Design algorithms to solve computational problems using a combination of original and existing algorithms.
CS.9-12.8.1.12.AP.2	Create generalized computational solutions using collections instead of repeatedly using simple variables.
CS.9-12.8.1.12.AP.3	Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.
CS.9-12.8.1.12.AP.4	Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
CS.9-12.8.1.12.AP.5	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
CS.9-12.8.1.12.AP.6	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

NEW JERSEY STUDENT LEARNING STANDARDS: CAREER READINESS, LIFE LITERACIES AND KEY SKILLS

TECH.9.4.12.CT.1	Identify problem-solving strategies used in the development of an innovative product or practice (e.g., 1.1.12acc.C1b, 2.2.12.PF.3).
TECH.9.4.12.CT.2	Explain the potential benefits of collaborating to enhance critical thinking and problem solving (e.g., 1.3E.12profCR3.a).

NEW JERSEY STUDENT LEARNING STANDARDS: COMPUTER SCIENCE AND DESIGN THINKING

See content area standards.

PRE-ASSESSMENTS

Discussion Questions:

- Where have you heard the word “inheritance” before? What does it mean?
- Describe the relationship between a class and an object.
- Consider the following code segment:

```
public class Vehicle
{
    private String make;
    private String model;
    private int year;
```

}

- Name as many different types of vehicles that you can that could be objects of the Vehicle class.

INSTRUCTIONAL PLAN

MODULE 9

LESSON 9.5

Lesson 9.5: Creating References Using Inheritance

In this lesson, students will create references using inheritance hierarchies. An object can take on different forms depending on its implementation. Java can call the correct method even when an object is disguised as a more generic reference type. This is known as polymorphism. This lesson corresponds with AP Computer Science A topic 9.5.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none">• Define reference variables of a superclass to be assigned to an object of a subclass in the same hierarchy
Student Learning Strategies	Code tracing Create a plan Error analysis Identify a subtask Look for a pattern Pair programming Predict and compare Simplify the problem Think aloud
Success Criteria	Check for understanding Completion of below activities AP Topic Questions
Formative Assessment (drives instructional decisions)	Teacher Observation Check for Understanding

	AP Topic Questions
Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Animal Sounds example. • Explore the Shape Areas example. • Explore the Person Class example. • Complete the Pies exercise. • Complete the Creating .equals exercise. • Complete the Online Companies Revisited exercise. • Complete the Assignments exercise.
Suggested Modifications	See Lesson 9.1 above.

LESSON 9.4

Lesson 9.4: super Keyword

In this lesson, students will use the super keyword. This keyword can be used to call a superclass's constructors and methods even if the method has been overridden. This lesson corresponds with AP Computer Science A topic 9.4.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none"> • Use the super keyword to call a superclass's constructors and methods
Student Learning Strategies	<ul style="list-style-type: none"> Code tracing Create a plan Error analysis Identify a subtask Look for a pattern Pair programming Predict and compare Simplify the problem Think aloud
Success Criteria	<ul style="list-style-type: none"> Check for understanding Completion of below activities AP Topic Questions
Formative Assessment (drives instructional decisions)	<ul style="list-style-type: none"> Teacher Observation Check for Understanding

	AP Topic Questions
Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Square Class example. • Explore the Animal Class example. • Explore the Apple Pie example. • Complete the Squares exercise. • Complete the Bank Accounts exercise. • Complete the Employees exercise. • Complete the Student Test Scores exercise.
Suggested Modifications	See Lesson 9.1 above.

LESSON 9.3

Lesson 9.3: Overriding Methods

In this lesson, students will learn how to override methods in a superclass/subclass relationship. Method overriding occurs when a public method in a subclass has the same method signature as a public method in the superclass. This lesson corresponds with AP Computer Science A topic 9.3.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none"> • Override a method in a superclass/subclass relationship
Student Learning Strategies	<ul style="list-style-type: none"> Code tracing Create a plan Error analysis Identify a subtask Look for a pattern Pair programming Predict and compare Simplify the problem Think aloud
Success Criteria	<ul style="list-style-type: none"> Check for understanding Completion of below activities AP Topic Questions
Formative Assessment (drives instructional decisions)	<ul style="list-style-type: none"> Teacher Observation Check for Understanding

	AP Topic Questions
Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Square is a Rectangle example. • Explore the Student toString example. • Explore the Restaurant Bills example. • Complete the Dogs Bark exercise. • Complete the Electric Cars exercise. • Complete the Online Companies exercise.
Suggested Modifications	See Lesson 9.1 above.

LESSON 9.2

Lesson 9.2: Writing Constructors for Subclasses

In this lesson, students will write constructors for subclasses. The superclass constructor can be called from the first line of a subclass constructor by using the keyword `super` and passing appropriate parameters. This lesson corresponds with AP Computer Science A topic 9.2.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none"> • Write constructors for subclasses
Student Learning Strategies	<ul style="list-style-type: none"> Code tracing Create a plan Error analysis Identify a subtask Look for a pattern Pair programming Predict and compare Simplify the problem Think aloud
Success Criteria	<ul style="list-style-type: none"> Check for understanding Completion of below activities AP Topic Questions
Formative Assessment (drives instructional decisions)	<ul style="list-style-type: none"> Teacher Observation Check for Understanding AP Topic Questions

Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Student Subclass example. • Explore the Shape Class example. • Explore the Implicit Call to Super example. • Complete the Students exercise. • Complete the Instruments exercise. • Complete the Foods exercise. • Complete the Clothing Store exercise.
Suggested Modifications	See Lesson 9.1 above.

LESSON 9.1

Lesson 9.1: Inheritance

In this lesson, students will learn about superclasses and subclasses. A superclass is a parent class that contains common attributes and behaviors used by subclasses (children). Subclasses can draw upon the existing attributes and behaviors of the superclass without repeating these in the code. This lesson corresponds with AP Computer Science A topic 9.1.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none"> • Create an inheritance relationship from a subclass to the superclass
Student Learning Strategies	Code tracing Create a plan Error analysis Identify a subtask Look for a pattern Pair programming Predict and compare Simplify the problem Think aloud
Success Criteria	Check for understanding Completion of below activities AP Topic Questions
Formative Assessment (drives instructional decisions)	Teacher Observation Check for Understanding AP Topic Questions

Activities and Resources

- Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding.
- Explore the Person Superclass example.
- Explore the Vehicle Superclass example.
- Explore the High School Student example.
- Complete the Person/Student Object exercise.
- Complete the Books exercise.
- Complete the Computers exercise.
- Complete the More Animals! exercise.

Suggested Modifications

English Language Learners

Adjusted Speech: The teacher changes speech patterns to increase student comprehension. This could include facing the students, paraphrasing, clearly indicating the most important ideas, and speaking more slowly.

Visuals: The teacher uses graphics, pictures, visuals, and manipulatives. This helps ELL students better understand and comprehend the subjects at hand.

Front-Loading Vocabulary: The teacher front loads vocabulary. This means providing students with a list of important vocabulary words they will need to know for a book, lesson, etc. prior to the lesson being taught. Including pictures to go with the vocabulary words is also very beneficial for the students.

Students with Individualized Education Plans/504s

Chunking: The teacher presents information in a way that makes it easy for students to understand and remember. Chunking is based on the presumption that our working memory is easily overloaded by excessive detail. The best way to deliver information is to organize it into meaningful units. Because students with special needs get overloaded easily, chunking is an effective strategy to use with them.

Checking for Understanding: It is important to constantly check for understanding, especially for students who have accommodations. Teachers want to make sure students understand the concepts being covered in a way that makes sense to them.

Extra time: The teacher provides students with special needs extra time to complete work or answer questions. It is important to give students enough time to process their thoughts.

Oral Reading: The teacher will read work orally to students. Class work such as tests and literature circles may need to be read aloud to the student.

Gifted & Talented Strategies

Extensions/Enrichments: Teachers will provide gifted and talented students with extension/enrichment projects. Students will be challenged to further their understanding, to apply acquired knowledge, and/or to produce something in reference to acquired knowledge.

Modify/Change Activities: Teachers will monitor and modify activities to accommodate those students who need to be challenged further. Additional reading, problem-solving, writing, or project work is necessary for those students who are ready to move on at a rate more accelerated than their peers. In this way, G & T students are provided the same opportunity for support as special needs students.

Students at Risk of School Failure

Directions or Instructions: Make sure directions and/or instructions are given in limited numbers. Give directions/instructions verbally and in simple written format. Ask students to repeat the instructions or directions to ensure understanding occurs. Check back with the student to ensure he/she hasn't forgotten.

Peer Support: Peers can help build confidence in other students by assisting in peer learning. Many teachers use the 'ask 3 before me' approach. This is fine, however, a student at risk may have to have a specific student or two to ask. Set this up for the student so he/she knows who to ask for clarification before going to you.

Alternate or Modified Assignments: Always ask yourself, "How can I modify this assignment to ensure the students at risk are able to complete it?" Sometimes you'll simplify the task, reduce the length of the assignment or allow for a different mode of delivery. For instance, many students may hand something in, the at-risk student may jot notes and give you the information verbally. Or, it just may be that you will need to assign an alternate assignment.

Increase One to One Time: When other students are working, always touch base with your students at risk and find out if they're on track or needing some additional support. A few minutes here and there will go a long way to intervene as the need presents itself.

Contracts: It helps to have a working contract between you and your students at risk. This helps prioritize the tasks that need to be done and ensure completion happens. Each day write down what needs to be completed, as the tasks are done, provide a checkmark or happy face. The goal of using contracts is to eventually have the student come to you for completion sign-offs

Hands On: As much as possible, think in concrete terms and provide hands-on tasks. This means a child doing math may require a calculator or counters. The child may need to tape record comprehension activities instead of writing them. A child may have to listen to a story being read instead of reading it him/herself.

Tests/Assessments: Tests can be done orally if need be. Break tests down in smaller increments by having a portion of the test in the morning, another portion after lunch and the final part the next day.

Seating: Seat students near a helping peer or with quick access to the teacher. Those with hearing or sight issues need to be close to the instruction which often means near the front.

LESSON 9.6

Lesson 9.6: Polymorphism

In this lesson, students will take a deeper dive into Polymorphism. At compile time, methods in or inherited by the declared type determine the correctness of a non-static method call. At run-time, the method in the actual object type is executed for a non-static method call. This lesson corresponds with AP Computer Science A topic 9.6.

Student Learning Intentions (SLI) WALT: (We are learning to...)

- Call methods in an inheritance relationship

Student Learning Strategies

Code tracing
Create a plan

	<p>Error analysis</p> <p>Identify a subtask</p> <p>Look for a pattern</p> <p>Pair programming</p> <p>Predict and compare</p> <p>Simplify the problem</p> <p>Think aloud</p>
Success Criteria	<p>Check for understanding</p> <p>Completion of below activities</p> <p>AP Topic Questions</p>
Formative Assessment (drives instructional decisions)	<p>Teacher Observation</p> <p>Check for Understanding</p> <p>AP Topic Questions</p>
Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Using Person Methods example. • Explore the Modified Student Class example. • Explore the Vehicle Methods example. • Complete the Which Team? exercise. • Complete the Cars exercise. • Complete the Library Books exercise. • Complete the Fun with Solids exercise. • Complete the Inheritance and Polymorphism handout.
Suggested Modifications	<p>See Lesson 9.1 above.</p>

LESSON 9.7

Lesson 9.7: Object Superclass

In this lesson, students will call and use the Object superclass. The Object class is the superclass of all other classes in Java. This lesson corresponds with AP Computer Science A topic 9.7.

Student Learning Intentions (SLI) WALT: (We are learning to...)	<ul style="list-style-type: none"> • Call Object class methods through inheritance <p>Utilize the Object class through inheritance</p>
Student Learning Strategies	<p>Code tracing</p>

	<p>Create a plan</p> <p>Error analysis</p> <p>Identify a subtask</p> <p>Look for a pattern</p> <p>Pair programming</p> <p>Predict and compare</p> <p>Simplify the problem</p> <p>Think aloud</p>
Success Criteria	<p>Check for understanding</p> <p>Completion of below activities</p> <p>AP Topic Questions</p>
Formative Assessment (drives instructional decisions)	<p>Teacher Observation</p> <p>Check for Understanding</p> <p>AP Topic Questions</p>
Activities and Resources	<ul style="list-style-type: none"> • Watch the lesson video and take the corresponding quiz. This quiz is a quick check for understanding. • Explore the Default Values example. • Explore the Override toString example. • Explore the Override equals example. • Complete the Equal? exercise. • Complete the Equals? - Part 2 exercise. • Complete the Equal Rectangles exercise. • Complete the 2D Array Tester exercise.
Suggested Modifications	<p>See Lesson 9.1 above.</p>

INTERDISCIPLINARY CONNECTIONS: NEW JERSEY STUDENT LEARNING STANDARDS FOR ELA, SOCIAL STUDIES, SCIENCE AND/OR MATHEMATICS

CCSS - English-Language Arts

Key Ideas and Details:

CCSS.ELA-LITERACY.RL.11-12.1 Cite strong and thorough textual evidence to support analysis of what the text says explicitly as well as inferences drawn from the text, including determining where the text leaves matters uncertain.

Integration of Knowledge and Ideas:

CCSS.ELA-LITERACY.W.11-12.1 Write arguments to support claims in an analysis of substantive topics or texts, using valid reasoning and relevant and sufficient evidence.

Production and Distribution of Writing:

CCSS.ELA-LITERACY.W.11-12.4 Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

Research to Build and Present Knowledge:

CCSS.ELA-LITERACY.W.11-12.7 Conduct short as well as more sustained research projects to answer a question (including a self-generated question) or solve a problem; narrow or broaden the inquiry when appropriate; synthesize multiple sources on the subject, demonstrating understanding of the subject under investigation.

Range of Writing:

CCSS.ELA-LITERACY.W.11-12.10 Write routinely over extended time frames (time for research, reflection, and revision) and shorter time frames (a single sitting or a day or two) for a range of tasks, purposes, and audiences

CCSS - Mathematics

Reason quantitatively and use units to solve problems:

CCSS.MATH.CONTENT.HSN-Q.A.2 Define appropriate quantities for the purpose of descriptive modeling.

Create equations that describe numbers or relationships:

CCSS.MATH.CONTENT.HSA-CED.A.1 Create equations and inequalities in one variable and use them to solve problems.

Analyze functions using different representations:

CCSS.MATH.CONTENT.HSF-IF.C.7 Graph functions expressed symbolically and show key features of the graph.

Apply geometric concepts in modeling situations:

CCSS.MATH.CONTENT.HSG-MG.A.1 Use geometric shapes, their measures, and their properties to describe objects

Calculate expected values and use them to solve problems:

CCSS.MATH.CONTENT.HSS-MD.A.1 Define a random variable for a quantity of interest by assigning a numerical value to each event in a sample space;

CCSS.MATH.CONTENT.HSS-MD.A.2 Calculate the expected value of a random variable;

English Language Arts

- Journal writing
- Close reading of industry-related content
- Create a brochure for a specific industry
- Keep a running word wall of industry vocabulary

Social Studies

- Research the history of a given industry/profession
- Research prominent historical individuals in a given industry/profession
- Use historical references to solve problems

World Language

- Translate industry-content
- Create a translated index of industry vocabulary
- Generate a translated list of words and phrases related to information technology

Math

- Compare and contrast use of equations and variables in algebra and programming.
- Program graphics and use the properties of geometric shapes
- Compare the computer graphic coordinate system with the Cartesian coordinate plane in math
- Compare probability and the use of random numbers in computer programming.
- Track and track various data, such as industry's impact on the GDP, career opportunities or among of individuals currently occupying careers

Fine & Performing Arts

- Create a poster recruiting young people to focus their studies on a career in Information Technology

Science

- Research the environmental impact of a given career or industry
- Research latest developments in Information technology
- Investigate applicable-careers in STEM fields

REFLECTIONS
