Course: AP Computer Science A (Mocha) | Module: Classes and Object-Oriented Programming

Lesson 4.4: Writing Classes

https://codehs.com/course/53/lesson/4.4

Description	Now that students have had practice *using* classes without actually changing the source code for a class, they dive in and actually start writing our own classes. In this lesson, students will learn the basics of writing classes including implementing constructors, using instance variables, and writing a `toString` method.
Objective	Students will be able to: • Create their own class • Create a constructor for a class • Determine what instance variables a class should have and create them • Create a toString method for a class so that it can be printed out • Create a program that uses their own class as a client
Activities	4.4.1 Video: Constructors 4.4.2 Check for Understanding: Quiz: Constructors 4.4.3 Example: Rectangle toString 4.4.4 Example: Student toString 4.4.5 Exercise: toString for Flowers 4.4.6 Exercise: Instance Variables for Your Dog 4.4.7 Exercise: Student GPA Field 4.4.8 Free Response: Free Response: What instance variables? 4.4.9 Exercise: Pizza Time! 4.4.10 Exercise: Fractions
Prior Knowledge	• Read documentation for a class to determine how to use it, how to make objects of it, and what methods are available to be called on those objects. • Write programs that use other classes as a client • Describe the relationship between classes and objects
Planning Notes	• There is a handout that accompanies this lesson. It can be used as an in-class activity or a homework assignment. Determine how and if this handout will be used and make the appropriate number of printouts prior to the class period. • Students will be creating their own classes in this lesson, which allows for more creativity, but also limits what can be tested in the autograders. Consider manually grading the *Pizza Time* exercise. • This is a longer lesson. You may want to consider breaking it up over multiple class periods. .
Standards Addressed	CR1: Teaches students to design and implement computer-based solutions to problems. CR4: Teaches students to code fluently in an object-oriented paradigm using the programming language Java. CR5: Teaches students to use elements of the standard Java library from the AP Java subset in Appendix A of the AP Computer Science A Course Description. CR6: Includes a structured-lab component composed of a minimum of 20 hours of hands-on lab experiences.
Teaching and Learning Strategies	**Lesson Opener:**

	<ul><li>Have students brainstorm and write down answers to the discussion questions listed below. Students can work individually or in groups/pairs. Have them share their responses. [5 mins]</li></ul> **Activities:** <ul><li>Watch the lesson video and complete the corresponding quiz. This quiz is a quick check for understanding [12-15 mins]</li><li>Explore the *Rectangle toString* example. [5-7 mins]<ul><li>Have students play around with the toString and change what is outputted.</li></ul></li><li>Explore the *Student toString* example. [5-7 mins]<ul><li>Challenge students to try adding a new instance variable to the Student class. Have them add that value to the constructor and the toString.</li></ul></li><li>Complete the *toString for Flowers* exercise. [5 mins]<ul><li>Have students look back on the examples if needed.</li><li>Make sure students pay attention to details, included spaces.</li></ul></li><li>Complete the *Instance Variables for Your Dog* exercise. [5 mins]<ul><li>The existing code will not work until the students make an update to add to their constructor.</li><li>The order of the variables is important. Make sure thy look at the `DogTester` to see which order to put the variables.</li></ul></li><li>Complete the *Student GPA Field* exercise. [5 mins]<ul><li>The existing code will not work until the students make an update to add to their constructor.</li><li>Have students look back on examples for more help.</li></ul></li><li>Complete the *What instance variable* free response. [3 mins]<ul><li>Answers will vary.</li><li>Possible answers are things like size, toppings, crust type, etc.</li></ul></li><li>Complete the *Pizza Time* exercise. [7-10 mins]<ul><li>Have students look back on examples for more help.</li><li>Answers will vary. and the autograder is limited. Consider grading these exercises manually.</li></ul></li><li>Complete the *Fractions* exercise. [10-12 min]<ul><li>Have students look back on examples for more help.</li></ul></li></ul> **Lesson Closer:** <ul><li>Have students reflect and discuss their responses to the end of class discussion questions. [5 mins]</li></ul>
Discussion Questions	**Beginning of Class:** <ul><li>Can you think of a Class that doesn't exist yet that you would find useful? What instance variables would it have?<ul><li>*Answers will vary.*</li><li><ul><li>Example: `Car` class. Some instance variables may be Make, Model, Color, MPG, etc.*</li></ul></li></ul></li><li>What are some important things to think about when deciding what instance variables a class should have?<ul><li>*Think about what the state of the object is. What values are needed to describe the object.*</li></ul></li><li>What is the purpose of a constructor?<ul><li>*The constructor helps create the object and provides initial code that is run at the time of instantiation. Often times instance variable values are assigned in the constructor.* **End of Class:**</li></ul></li><li>What is the purpose of a toString method? Why is it useful to write a toString method for your class?<ul><li>*The `toString` provides a String representation of an object. It is useful so that you can print your object out easily without having to write long print statements each time.*</li></ul></li><li>Why might it be important to keep instance variables private?<ul><li>*Instance variables are kept private so that we can control how they are accessed. This can be important because sometimes we may have multiple variables that need to be updated. For example, if you had a student class with a String level to represent*</li></ul></li></ul>

	middle or high school and a int grade, you would not want the user to be able to change the grade without changing the level. • How do we access instance variables if they are private? ◦ *We use accessor (getter) and mutator (setter) methods. Often times these just return a value or set a new value, but we can add additional logic to these methods.*
Resources/Handouts	Constructors (Student) Constructors (Teacher) Writing a Class - Student

Vocabulary

Term	Definition
Class	A class is a template, or a blueprint, from which Java objects are created. All Java programs start with a class.
Method	A method is a way to teach the computer a new command
String	String is a Java type that represents a string of characters (text)
Object	An object is a single instance of a Java class. An object has both state and behavior.
State	The state of an object is all of the object's associated data. It is the *state* that the object is in.
Behavior	The behavior of an object is what the object is able to do. It is the actions that can be performed by the object.
Instance	Instance is what you call a specific object constructed from a class. Instance and object generally refer to the same thing. An object is a specific instance of a class.
Instance Variable	A variable defined in a Class, for which each object of the class has its own copy.
Constructor	A constructor is a special method of a Class that constructs a new object (a new instance) of the Class and sets the initial values for the instance variables.
toString	toString is a special method you write in your class that returns a String representation of the object.

Modification: Advanced	Modification: Special Education	Modification: English Language Learners