Lesson 7.04: Inheritance ## Learning Objectives Students will be able to... * Define and identify: **inheritance**, **parent class**, **child class** * Create a class that inherits from anther class * Overwrite methods of parent class in a child class ## Materials/Preparation * [Do Now] * [Lab - Pokemon Child Classes] ([printable lab document]) ([editable lab document]) * [Associated Reading - section 7.4] (https://tealsk12.gitbook.io/intro-cs-2/readings#7-4) * Read through the do now, lesson, and lab so that you are familiar with the requirements and can assist students ## Pacing Guide | **Duration** | **Description** | | ---------- | ---------- | | 5 Minutes | Do Now | | 10 Minutes | Lesson | | 30 Minutes | Lab | | 5 Minutes | Debrief | ## Instructor's Notes ### 1. Do Now * Display the Do Now on the board. * Students will explore an example of a new `Dog` class inheriting the methods of the `Pet` class ### 2. Lesson #### Instruction - Inheritance * when you create a new class that is a subclass of the original (ex. the `Dog` class "inherits" the properties/methods of the `Pet` class.) #### Discussion * Ask students: what is the difference between the `Dog` declaration and the `Pet` declaration? * Discuss the error and what was missing in the original code. #### Instruction - Parent and Child Class * When a class inherits from another class, the class it inherits from is called the **parent class** and the class that inherits is called the **child class**. * Ask: in the example from the do now, which is the child and which is the parent? * Child classes gain access to all the methods of the parent class * What does `dog1.make_nosie()` print out? * Child classes can also overwrite their parent classes. Have the students practice overwriting `make_noise` in the `Dog` class so that the dog will print out `bark bark` ### 3. Lab * Given a generic Pokemon class, create three child classes that represent different types of Pokemon. * Consider demonstrating the creation of one of the child classes before having students start the lab independently. ### 4. Debrief * Go over students' questions. Ask what questions the students have and review instance, class, methods, `init`, `str`, etc. ## Accommodation/Differentiation In the Pokemon lab, students may need clarification regarding how to use `isinstance` and how to manipulate the `defend` method to meet the requirements of each child class. Consider demonstrating the creation of one child class for all students before having students work on the lab. ## Forum discussion [Lesson 7.04: Inheritance (TEALS Discourse Account Required)] (https://forums.tealsk12.org/c/2nd-semester-unit-7-classes/lesson-7-04-inheritance) [Do Now]:do_now.md.html [Lab - Pokemon Child Classes]:lab.md.html [printable lab document]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/04_lesson/lab.pdf [editable lab document]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/04_lesson/lab.docx