# Lesson 7.05: Pokemon Project ## Learning Objectives Students will be able to... * Engage in **class design** before beginning coding * Apply what was learned with respect to **classes**, **methods**, and** inheritance** to create an implementation of Pokemon ## Materials/Preparation * [Project Spec - Pokemon] ([printable project Spec]) ([editable project spec]) * [Alternate Project Spec - Mailing List] ([printable alternate project Spec]) ([editable alternate project spec]) * Solution (access protected resources by clicking on "Additional Curriculum Materials" on the [TEALS Dashboard]) * Read through the project spec so that you are familiar with the requirements and can assist students * Try creating your own project so that you can Review [4 Steps to Solve Any CS Problem] * [Editable Grading Rubric](https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/rubric.docx) ### Day 1 Pacing | **Duration** | **Description** | | ---------- | ---------- | | 5 Minutes | Project Handout | | 5 Minutes | Mini-Lesson | | 15 Minutes | Project Overview | | 30 Minutes | Project Planning | ### Days 2-7 Pacing | **Duration**|**Description** | |--|--| | 5 Minutes | Planning/Questions | | 10 Minutes | Review | | 35 Minutes | Project Work | | 5 Minutes | Wrap up | ## Instructor's Notes ### Day 1 #### 1. Handout Project Specifications * Read through the Project Spec with students * Demo a completed project to show user experience. #### 2. Mini-Lesson * Discuss **Class Design** * If you find yourself creating many classes with similar methods, use inheritance! * Figure out the actual structure without writing code and use that to create your classes #### 3. Project Overview * Go over the Pokemon project spec * Review the major aspects and requirements of the game #### 4. Project Planning * Have students write down the classes and methods they need to create * Students should then outline what they will do each day in order to complete the project on time ### Days 2-7 #### 1. Planning/Questions * Have students review and update what they want to accomplish that day and any questions they have from the previous day. #### 2. Review * if necessary, review any concepts or struggles the class was having. #### 3. Project Work * students work on their projects independently. #### 4. Wrap Up * have the students write down what they struggled on or had a hard time doing. ## Grading ### Scheme/Rubric [Editable Grading Rubric](https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/rubric.docx) | Points | Percentage| Objective | Lesson | | :---: | :---: | --- | --- | |9|32%| The Student can create a class and an instance|7.01, 7.02| |6 | 21%| The student can create methods for classes|| |3 | 11%| The student can correctly use inheritance|| |5 |18% | Student can decompose a problem to create a program from a brief|| |5 |18% | Student uses naming/ syntax conventions and comments to increase readability| | | 28 | | **Total Points** || ## Forum discussion [Lesson 7.05: Pokemon Project (TEALS Discourse Account Required)](https://forums.tealsk12.org/c/2nd-semester-unit-7-classes/lesson-7-05-pokemon) [Project Spec - Pokemon]:project.md.html [Alternate Project Spec - Mailing List]:alternate_project.md.html [printable project Spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/project.pdf [editable project spec]:https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/project.docx [TEALS Dashboard]:http:/www.tealsk12.org/dashboard [4 Steps to Solve Any CS Problem]:https://github.com/TEALS-IntroCS/2nd-semester-introduction-to-computer-science-principles/raw/master/units/4%20Steps%20to%20Solve%20Any%20CS%20Problem.pdf [printable alternate project Spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/alternate_project.pdf [editable alternate project spec]:https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/7_unit/05_lesson/alternate_project.docx