Lesson 1.06: Project ## Learning Objectives * Apply basic Python knowledge about inputs/outputs and variables to create a game of Madlibs * Practice good debugging skills ## Materials * [Project Spec - Mad Libs] ([Printable Project Spec]) ([Editable Project Spec]) * [Alternate Project Spec - Magic Square] ([Printable Alternate Project Spec]) ([Editable Alternate Project Spec]) * Solution (access protected resources by clicking on "Additional Curriculum Materials" on the [TEALS Dashboard]) * [Editable Grading Rubric] (https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/rubric.docx) ## Preparation * Practice running the example code * Read through the project specifications so that you can completely communicate the requirements of the project * Review [4 Steps to Solve Any CS Problem] ## Day 1 Pacing | **Duration** | **Description** | | ------------ | ---------- | | 5 Minutes | Quiz Debrief | | 10 Minutes | Project Overview | | 40 Minutes | Project Work | ## Day 2 Pacing | **Duration** | **Description** | |------------- | ------------------- | | 45 Minutes | Project Work | | 10 Minutes | Wrap Up - Student Demos | ## Instructors Notes ### 1. 4 Steps to Solve Any CS Problem * Introduce students to the [4 Steps to Solve Any CS Problem] ### 2. Project Overview * Introduce students to the Mad Lib concept by using a short, written out Mad Lib on the whiteboard, poster paper, or projector. * Pass out and the project specification and walk students through all the requirements and potential challenges. * Emphasize that prompts must ask for the correct noun-verb combinations. * Encourage students to look at the grading rubric on page two repeatedly throughout the project to ensure they are meeting all the requirements. * Demo a sample project solution (access protected resources by clicking on "Additional Curriculum Materials" on the [TEALS Dashboard](https://www.tealsk12.org/dashboard/)) for students to see how a completed program should function. * Identify the sub problems of Mad Libs * Have students list what variables, inputs, and print statements they will need ### 3. Project Work * This project is a summative assessment for the unit. Students should be demonstrating mastery of all the skills covered. * Most students will require roughly 1 hour of total work time to complete the project * Assess the progress of your students regularly using such techniques as asking them to demonstrate their incomplete programs, tracking questions asked during lab time, and/or utilizing peer reviews. * Adjust the amount of time allowed for the project to fit the needs of your students * It is vital that nearly all students complete the project before moving on * If most students have the ability to work on assignments at home, the amount of in-class time provided can be reduced if necessary. * If this approach is taken, be sure to make accommodations for students who not able to work at home, such as after school lab hours * Ensure that students are able to ask questions in class throughout the project ### 4. Wrap Up - Student Demos * Celebrate and showcase student work once projects are completed. * Have students demonstrate their Mad Libs for the class, with the class choosing what nouns/verbs/etc. to use for the story. ## SNAP Flashback - MadLibs ! [MadLibs](lesson1.06%20-%20code.png) ## Accommodation/Differentiation Ask students to research casting. Have them add, subtract, or multiply values as part of the story. ## Grading ### Objective Scoring Breakdown [Editable Grading Rubric](https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/rubric.docx) | Points | Percentage | Objective | Unit Location | | :---: | :---: | --- | :---: | | 2 | 10% | Students can correctly use the IDE | 1.01 | | 6 | 28% | Student can correctly identify and store variable types | 1.02 1.04 | | 3 | 14% | Student can use the print function | 1.03 | | 5 | 24% | Student can decompose a problem to create a program from a brief | | 5 | 24% | Student uses naming/ syntax conventions and comments to increase readability | | **21** | **Total points** | | | ### Scoring Consideration You may need to adjust the points in order to fit your class. Treat the percentages as a guide to determine how to weight the objectives being assessed. ## Forum discussion [Lesson 1.06: Mad Libs (TEALS Forums Account Required)] (https://forums.tealsk12.org/c/2nd-semester-unit-1/1-06-madlibs) [Mad Libs - Example Code]: project_file.py [Project Spec - Mad Libs]: project.md.html [Alternate Project Spec - Magic Square]: alternate_project.md.html [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 project Spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/project.pdf [editable project spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/project.docx [printable Alternate project Spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/alternate_project.pdf [editable Alternate project spec]: https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/1_unit/06_lesson/alternate_project.docx