Sample Rubric **Many of the python units have the option for more than one project. The rubric provided is an objective based rubric that can be used to asses either project. These rubrics could also be used as a guide to create new projects.** **This is a sample of how you might fill out the rubric to assess students progress with unit 7 skills.** ### The Student can create a class and an instance (7.01, 7.02) | 3 | 2 | 1 | 0 |Points| |:--:|:--:|:--:|:--:|:--:|:--:| |**X** Student creates classes correctly| Student usually creates classes correctly| Student attempts to create classes| No evidence that the student can create classes | 3 | |**X** Student correctly creates instances of a class| Student usually creates and instance of a class correctly| Student attempts to create a class| No evidence that the student can create a instance of a class|3 | |**X** Student instantiates a class with arguments |__| Student attempts to instantiate a class with arguments| No evidence the student can instantiate a class with arguments|3 |Student always adds attributes to an instance correctly | **X** Student usually adds attributes to an instance correctly| Student attempts to add attributes to an instance| No evidence the student can add attributes to an instance|2| |**Sub Total**|| | |**11**| ## The student can create methods for classes | 3 | 2 | 1 | 0 |Points| |:--:|:--:|:--:|:--:|:--:|:--:|:--:| |**X** Student always uses the `self` argument correctly| Student usually uses the `self` argument correctly| Student attempts to use the `self` argument| There is no evidence Student can use the `self` argument|3| |Student correctly uses the `__str__` method||Student attempts to use the `__str__` method| **X** No evidence Student can use the `__str__` method|0| |**Sub Total**|| | |**3**| ## The student can correctly use inheritance | 3 | 2 | 1 | 0 |Points| |:--:|:--:|:--:|:--:|:--:|:--:|:--:| |**X** Student uses inheritance correctly | __| Student attempts to use inheritance| No evidence that the student can use inheritance correctly|3| |**Sub Total**|| | |**3**| ### Student can decompose a problem to create a program from a brief |3 |2 |1 |0 |Points |:-:|:-:|:-:|:-:|:-:|:-:|:-:| |Student program runs without error | **X** The students program has a few errors, but it does not impact the program's functionality | Student program has errors that impact the program's functionality | Student program is not functional |2| | __ | **X** Students submitted documentation showing planning for most variables and functions.| Students submitted documentation showing planning for a few variables and functions.| No evidence of planning|2| |**Sub Total**|| | |**4**| ### Student uses naming/ syntax conventions and comments to increase readability |2 |1 |0 |Points |:-:|:-:|:-:|:-:|:-:| |Syntax conventions are generally used |**X** Sometimes syntax conventions are used| No evidence of syntax conventions to aid in code readability|2| |**X** All variables have clear names| Some variables have clear names| No evidence of using variable names to aid in code read ability|2| |__|**X** Student comments aid code readability| No evidence of using comments to aid in code readability.|1| |**Sub Total**|||**5**| **TEALS encourages you to talk with your teaching team to decide how much weight you want to give these projects. The raw score can be multiplied by a weighted value so the project fits into the grading scheme that works for your team and students.** ## Final Grade |||| | --- | --- | --- | |_ _ **31** _ _ _|__ _26_ __ __ __ X __ __3.22 __ _ __|_ _**84** (after rounding)_ _| | Points Possible | Points Earned x Weight | total Points | |Total Points| Weight| |-|-| |200|6.45| |150|4.83| |100|3.22| |50|1.661|