

Final Project Rubric - Programming Foundations with Python

Final Project Rubric: Programming Foundations with Python

Overview

This rubric is here to help you understand the specifications for how your project will be evaluated. It is the same rubric that the person evaluating your project will use. You should look at the rubric **before you begin working** on this project **and before you submit it**.

How Grading Works

1. The project evaluator will use this rubric to evaluate the code as well as all the other required documents.
2. Your evaluation will simply be “meets specifications,” “exceeds specifications,” or “does not meet specifications.”
 - a. The project “meets specifications” if **all** the specified “meet specifications” criteria are met.
 - b. The project “exceeds specifications” if **all** specified “exceeds specifications” criteria are met; for areas where no “exceeds specifications” criterion is specified, the project should “meet specifications”.
 - c. The project “does not meet specifications” if **any** criterion “does not meet specifications.” In this case, you will have the opportunity to revise and resubmit the project based on the feedback provided in the evaluation.

Before You Submit

1. After you have completed the Problem/Idea Definition, documented a Solution Design and created a program that successfully fulfills the problem definition, go through each rubric item and do your best to honestly evaluate where you think your project stands.
2. If you think your project “does not meet

specifications” for **any** criterion, you should make the necessary changes.

3. Once you’re confident that your project “meets specifications” or “exceeds specifications,” you can submit by following the [project submission instructions](#).

The Rubric

Criterion	Does Not Meet Specifications	Meets Specifications	Exceeds Specifications (Completely Udacious)
Problem/Idea Definition			
Does the document define the problem/idea clearly?	The idea definition is vague or not provided.	The project problem or idea is clearly defined.	
How original is the problem/idea?	The project does not add enough to an existing class project, or the project idea is limited in scope.	The project is a significant enhancement of one of the course projects, and/or plans to achieve a well defined, meaningful result using objects from the Python Standard Library or an external library.	In addition to meeting specs, the project is a novel idea inspired by a circumstance faced by the student, or brings joy into the user's life.
Solution Design and README			
Does the document clearly outline the steps needed to solve the problem (including drawing class diagrams as shown in Lesson 3b)?	<p>The solution design is vague or not provided, or does not list the steps to solve the problem.</p> <p>The solution design cannot be understood without a programming background.</p> <p>Class diagrams are missing or incomplete.</p>	<p>The steps needed to solve the problem are clearly listed.</p> <p>The design includes pseudo-code that can be read and interpreted by people with no programming background.</p> <p>Class diagrams are included to explain the attributes (both data and methods) of the classes defined and the relationships between</p>	

		them.	
Is the README file present and complete, clearly providing all the necessary instructions for running the program?	The README file is missing, unclear, or incomplete.	The README file provides full and clear instructions regarding program setup.	
Code Functionality			
Does the code fully and satisfactorily solve the problem or implement the idea listed in the problem definition?	The code does not work in accordance with the ideas in the problem definition.	The code meets all the requirements listed in the problem definition.	
Does the code run without errors?	The code crashes during execution. Required inputs to the program are not validated before use.	The code runs satisfactorily and without errors. If the program requires user input, all input is appropriately validated before use.	
User Interface			
Is the program intuitive to use?	The user might be confused and not understand what is expected during program execution. User input is required, but the user cannot easily follow what is needed. If user input is required, the program does not print an appropriate message when the input is invalid.	The user can run the program straightforwardly, without finding it confusing or too complex. If input is required, the program clearly indicates to the user what input is expected. If input is required, the program prints user friendly error messages when the input is invalid.	The user interface is intuitive and polished.
Code Readability			
Are comments used effectively to explain different parts of the code?	Code is not easily readable or is not commented.	Functions, classes, and modules have adequate comments such that a fellow programmer can understand the purpose of the program.	
Are the design guidelines highlighted in the course followed?	Similar coding constructs (classes, functions, variables, etc.) do not use similar naming conventions (CamelCase, snake_case, ALL_CAPS, etc.).	Similar coding constructs (classes, functions, variables, etc.) use similar naming conventions (CamelCase, snake_case, ALL_CAPS, etc.).	

	The names for variables, classes, and procedures are inconsistent and/or not meaningful.	The naming scheme for variables, classes, and procedures is consistent with the PEP 8 Style Guide .	
Object-Oriented Programming Usage			
Does the code use Object Oriented Programming?	The code is implemented procedurally, using functions alone.	The code uses object oriented techniques.	
Does the code instantiate existing classes?	The code uses no predefined classes from the Python Standard Library or some external library.	Code creates instances or objects of at least one class from the Python Standard Library or an external library. These instances are appropriately used to accomplish or further the goal of the project.	
Does the code define its own classes?	No programmer-defined classes are used in the code.	Code defines at least one custom class that has appropriate attributes including data and methods that act upon that data. Instances of this class are appropriately used to accomplish or further the goal of the project.	
Does the code leverage principles of inheritance appropriately?	The code could have benefited from inheritance, but those techniques were not used; or inheritance is used inappropriately, and the code would be better without it.	If the code could have benefited from inheritance, those techniques were appropriately used.	