

Adventure Game

Code Review

| CRITERIA | MEETS SPECIFICATIONS |
|--|--|
| Output text to the console. | Descriptions are printed to the console for the player to see. |
| Import modules and use functions from those modules. | <p>The <code>time.sleep</code> function is used to create delays between messages so that they aren't all printed at once.</p> <p>The <code>random.choice</code> or <code>random.randint</code> function is used to influence the game so that each game is different in some way.</p> |
| Use the <code>input</code> function in combination with conditional statements (e.g., <code>if</code> and <code>while</code>) to create an interactive program. | <p>The <code>input</code> function is used to ask the player what they would like to do.</p> <p>The player's choices affect what happens in the game, including:</p> <ul style="list-style-type: none">• Whether the player wins or loses• Whether to restart or exit after the game is over |
| Use a conditional loop to handle invalid input. | <p>If the player enters a choice that is not valid, the game gives them the chance to retry until they enter a valid option.</p> <p>The game does not crash and does not treat invalid input as a valid choice.</p> |
| Refactor code by defining and calling functions. | <p>The code includes at least four function definitions that are used to improve the code in some way, such as by:</p> <ul style="list-style-type: none">• Reducing repetition• Reducing complexity• Improving the readability or organization of the code <p>Each function should have a single purpose and a name that</p> |

| CRITERIA | MEETS SPECIFICATIONS |
|--|--|
| | describes that purpose. |
| Write code that follows the standard Python style guide. | The <code>pycodestyle</code> tool reports zero errors and zero warnings. |
| Test code and produce an error-free program. | The program is a playable game, and runs from start to finish without crashing or displaying errors. |