Assignment P1

Due Date: September 24

Purpose
Yippee! It’s time for you to try out some basic Python constructs in a short, but exciting, program.

Problem
The Clueless Consulting Company (CCC) of Walla Walla Washington does a thriving business supporting the government. With all the work they have, owners Will and Wendy Wonka are having trouble billing their clients properly. The work that they do is billed by the hour; the current rate is a very reasonable $129/hour. CCC is very picky about things, so all the time that Will and Wendy spend consulting is kept track of meticulously. You are hired to write a Python program that prompts for the time that they spend on a job, computes the total time they’ve worked, computes the amount they will bill, and finally displays all the information in a neat and orderly way.

Input
The program should prompt the user for the number of hours and minutes Will and Wendy have worked on the project. Thus, you are getting four values: the number of hours and minutes for Will and the number of hours and minutes for Wendy. For example:

Enter Will’s hours: 13
Enter Will’s minutes: 45
Enter Wendy’s hours: 18
Enter Wendy’s minutes: 54

Output
The output should display an aligned table containing the days, hours, and minutes that Will and Wendy worked as well as the totals for each of those. Below this, the total billable hours should be shown; a billable hour is defined as any full hour plus one hour for any number of minutes ≥ 30. All numeric values should be right justified. There should be a dollar sign in front of the billable total. For the input shown above, the output might look like the following (you do not have to follow this example exactly):

<table>
<thead>
<tr>
<th>Days</th>
<th>Hours</th>
<th>Minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Will</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>45</td>
</tr>
<tr>
<td>Wendy</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>54</td>
</tr>
<tr>
<td>Totals</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>39</td>
</tr>
</tbody>
</table>

Total billable hours: 33 @ $129/hour = $4257

If any value is input as 0, then it’s fine to show a zero in the output.
Specifics

- Be sure to adequately test your program! Any valid\(^1\) number of hours and minutes should produce a correct bill. The program will be tested only with valid data.

- Use **named** “constants” (variables) for appropriate values.

- Include a good introductory comment, as described in class.

- Choose good (mnemonic) identifiers.

- The program should be properly indented using spaces **only** (no tabs!).

Notes

- For maximum points, write efficient code. Note that this does not mean fewer lines or unreadable code. Find the best *algorithm* for the problem.

- You do **not** need if (selection) or repetition statements in any part of this program.

- The name of your **source code** file (the one that ends with “.py”) should be your last name and project number; e.g., gousieP1.py.

- To turn in the project, send your source code as a plain text attachment via email to mgousie@wheatoncollege.edu. Submit by 11:59:59 on the due date for the project to be on time.

- A printed version of your source code is due at the **beginning of class** on September 25\(^{th}\). Write/print and **hand sign** the Wheaton Honor Code Pledge on what you turn in: “I have abided by the Wheaton College Honor Code in this work.”

- Be sure to test with the Wing IDE if you are using another environment for development before turning in your project.

- Remember to save all of your work until your project is returned.

\(^1\)It is valid for the input hours and minutes to be more than 24 and 60, respectively. A negative number, for example, would not be valid.

*It’s almost too easy.*

– Dr. Schulz in my Calculus III class