

Tangent Labs Python Web Developer Pre-Interview Test

Please provide answers for the following 4 tasks. We favour clarity and brevity over verbose answers so don't feel they need to be overly long. Unless otherwise stated, for Python tasks you may assume no greater than Python 2.6 is available to you.

1)	<p>A string is a palindrome if it reads the same from left-to-right as it does right-to-left. e.g "nolemonnomelon", "racecar" & "neverodddoreven" are all palindromes.</p> <p>A string is an anagram of another string if it consists of exactly the same characters but in another order. e.g The string "carrace" is an anagram of "racecar".</p> <p>Write a function <code>def is_anagram_of_palindrome(str):</code> such that it returns True if the string <code>str</code> is an anagram of a palindrome, and otherwise returns False. You may assume that <code>str</code> contains only lowercase characters a-z and is not empty.</p> <p>e.g Given <code>str = "carrace"</code> the function should return True as it is an anagram of the palindrome "racecar". Given <code>str = "tangent"</code> the function should return False as while it is an anagram, it is not possible that it is an anagram of a palindrome. We are not dealing with English words here, "bbcbb" is still a palindrome.</p>
2)	<p>Please point out the inadequacies with the following markup. You may assume it has been served as an entire page and should validate to a W3C standard. The inadequacies entail semantics, separation, spec and style.</p> <pre data-bbox="228 1157 1421 1577"><html> <head> <title>Job Listings</title> </head> <body> <div class=container> <h1>The Latest & Newest Job Listings</h1> <!-- Start main content --> <p>Jobs <div style="font-style: italic;">here</div>!</p> <p>Timeline.</p> <p>We specialise in tech jobs.</p> </div> </body> </html></pre>
3)	<p>Write a short, succinct comparative review of the Django web framework vs another web framework that you feel strongly about (one way or the other).</p>
4)	<p>This final task is designed to show how you think; there is no right or wrong but that's not to say the implementation won't be judged.</p> <p>Write a Python file/module that is solely responsible for filling a bath. You can define any API you like to control the bath. To get you on the right path - you can do better than the</p>

following answer:

```
import time

bath = Bath()
bath.open_taps()
time.sleep(300)
bath.close_taps()
print "Bath is ready"
```

Finally, if you attain an interview, we would love to see some relevant and recent code that you are proud of; this may be in the form of a public or private github account, code sent by email or a print-out/USB stick that you bring to the interview. It doesn't have to be vast - often 50 lines of worthy, interesting code could say more than a thousand line application. Part of any interview attained may be to talk through your code and explain what makes it great. Good luck.