HyperionDev

Debugging – The Stack Trace
Model-Answer Approach

Visit our website

Auto-graded task 1

In the original code found in the provided **debugging.py** file, there are a few key issues that need to be resolved to meet the expected output. First, the function `print_values_of` is designed to print values from a dictionary based on the provided keys. However, the function signature expects `keys` to be a single argument rather than multiple key arguments, which causes a mismatch when passing multiple keys. Additionally, inside the function, the variable `k` is used to retrieve dictionary values, but this variable is undefined, leading to a `NameError`.

The next problem lies in the way the keys are passed to the function. The original implementation passes `'lisa'`, `'bart'`, `'homer'` as individual arguments rather than as a list, which is necessary for the function as currently designed. Lastly, the `simpson_catch_phrases` dictionary contains a syntax error in Homer's catchphrase. The apostrophe within `'d'oh!'` breaks the string, which leads to a `SyntaxError`.

In the model answer, the approach involves fixing these errors step by step. First, we update the function to take `keys` as a list of keys rather than individual arguments. Inside the function, we replace the undefined variable `k` with `key` to correctly reference the current key in the loop. Next, we modify the function call to pass the keys as a list `['lisa', 'bart', 'homer']` instead of individual arguments. Finally, the syntax error in Homer's catchphrase is corrected by changing the single quotes around `'d'oh!'` to double quotes, in order to avoid the conflict caused by the apostrophe.

A possible pitfall to watch for is forgetting to pass the keys as a list when calling the function, as this would result in a `TypeError`. Another common issue could be accidentally accessing keys that don't exist in the dictionary, which would raise a `KeyError`. Adding error handling, such as checking for the presence of keys or using the `.get()` method, could be useful for working on similar tasks.