AI ASSISTED CODING

ROLLNO:2403A52082

BATCH:04

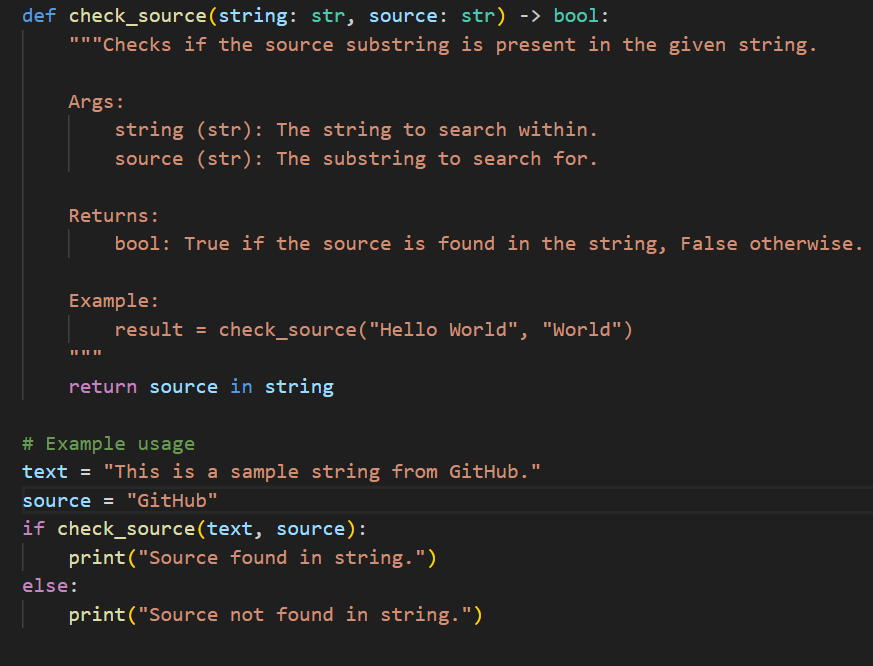
LAB-9.4

TASK-1:

PROMPT: Add Google-style docstrings to all functions in this Python script. Each docstring should include a brief description of the function, parameter names with type hints, return values with type hints, and an example usage. Do not include input-output examples. Ensure the formatting strictly follows Google-style documentation. Here's the script:def check\_source(string, source):  
return source in string.

**Example usage**

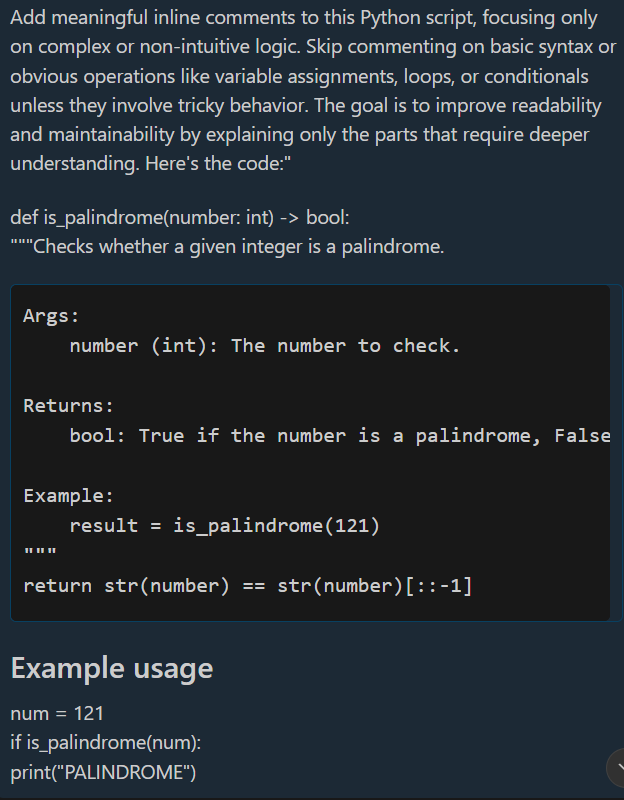
text = "This is a sample string from GitHub."  
source = "GitHub"  
if check\_source(text, source):  
print("Source found in string.")  
else:  
print("Source not found”)

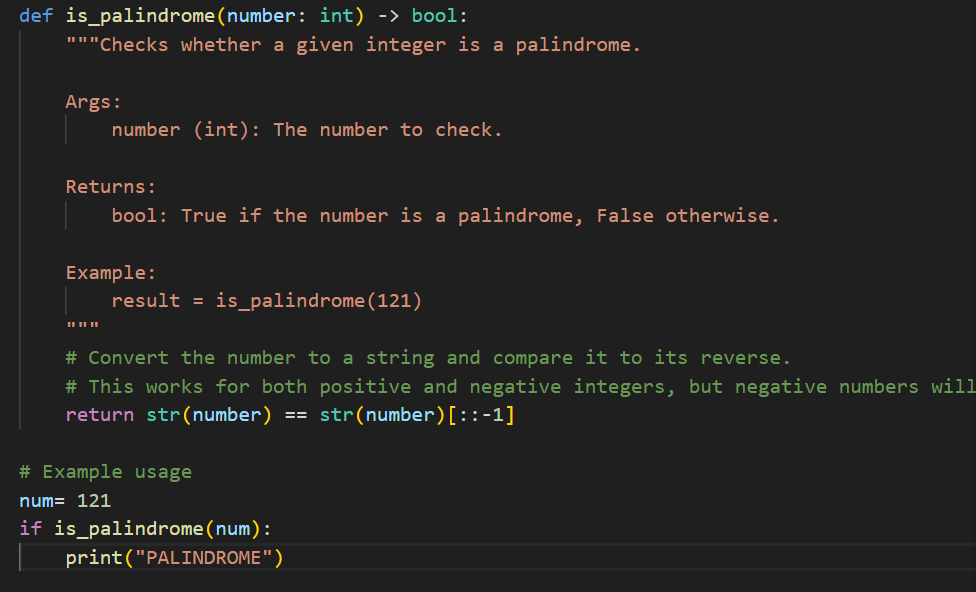
CODE:

OUTPUT:

EXPLANATION: This code defines a function [check\_source](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") that checks if a substring ([source](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)) exists within a given string ([string](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)).  
It uses Python’s in operator to return True if the substring is found, otherwise False.  
A Google-style docstring describes the function, its parameters, return value, and provides an example usage.  
The script then tests the function with the string "This is a sample string from GitHub." and the substring "GitHub", printing a message if the source is found.

TASK-2:

PROMPT:

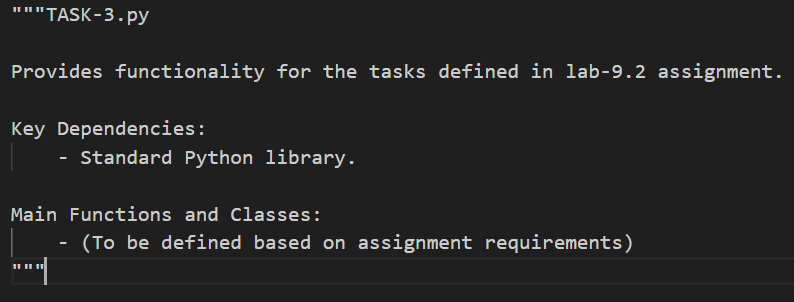
CODE:

OUTPUT:

EXPLANATION: This code defines a function [is\_palindrome](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") that checks if an integer reads the same forwards and backwards.  
It converts the number to a string and compares it to its reverse using slicing ([::-1]).  
If the number is a palindrome, it prints "PALINDROME" for the test case 121.

TASK-3:

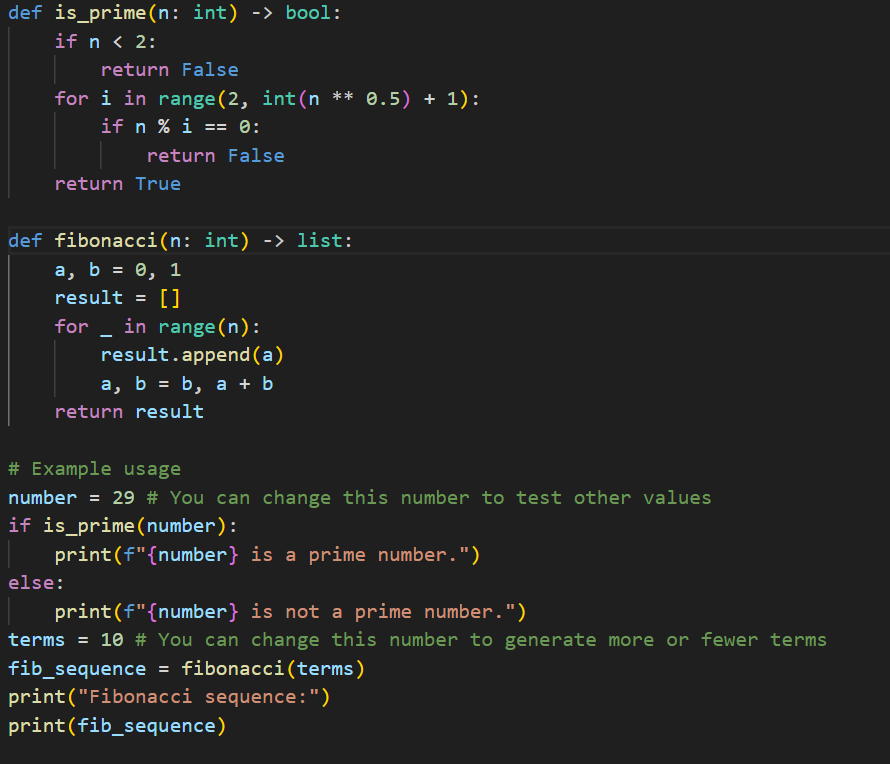
PROMPT: Write a module-level docstring for this Python file. The docstring should appear at the top of the file and clearly summarize the module’s purpose, its key dependencies, and the main functions or classes it defines. Do not rewrite or explain the entire code. Focus on clarity, conciseness, and Google-style formatting.

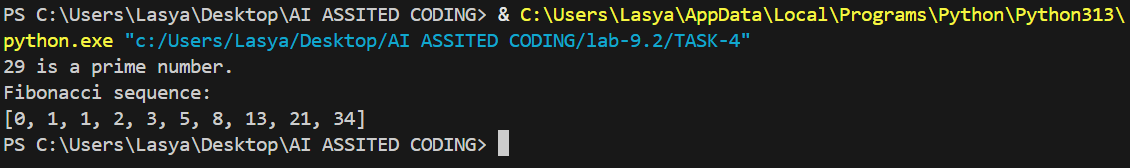
CODE:

EXPLANATION: This module-level docstring summarizes the purpose of the file, stating that it provides functionality for tasks in the [lab-9.2](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) assignment.  
It lists the standard Python library as a key dependency and notes that main functions and classes will be defined according to assignment requirements.  
The docstring follows Google-style formatting for clarity and organization.

TASK-4:

PROMPT: Transform the inline comments in this Python code into structured Google-style function docstrings. Move all relevant explanations from comments into the docstrings, and remove the original comments. Each docstring should include a brief description, parameter names with type hints, return values with type hints, and an example usage. Ensure the meaning is preserved and the formatting follows Google-style standards. Here's the code:

CODE:

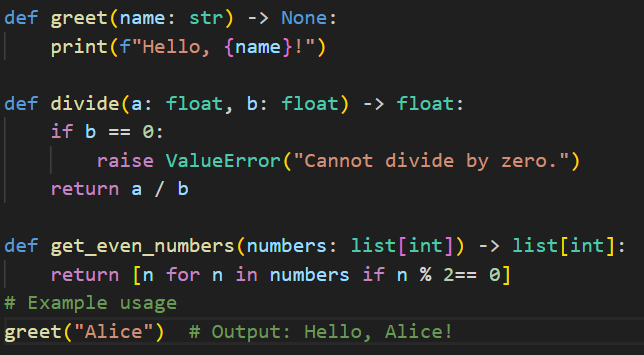
OUTPUT:

* EXPLANATION: [is\_prime(n: int) -> bool](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html): Checks if a number is prime by testing divisibility from 2 up to the square root of [n](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html). Returns True if [n](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) is prime, otherwise False.
* [fibonacci(n: int) -> list](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html): Generates a list containing the first [n](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") terms of the Fibonacci sequence.

The example usage tests if 29 is prime and prints the result, then generates and prints the first 10 Fibonacci numbers.

TASK-5:

PROMPT: **Review and correct the docstrings in this Python code. Some of them may be outdated or inaccurate. Rewrite each docstring to accurately reflect the current behavior of the function, and format them using Google-style documentation. Include a brief description, parameter names with type hints, return values with type hints, and an example usage. Here's the code:"**  
*(Paste your Python code with existing docstrings below this line)*

CODE:

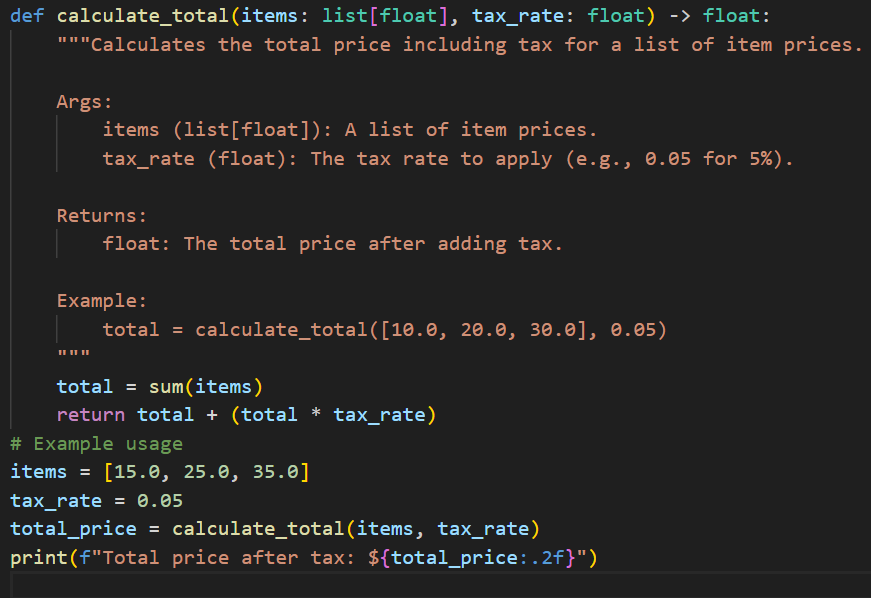
OUTPUT:

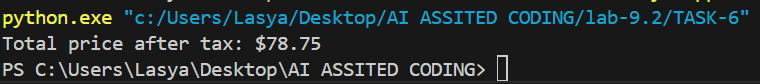
* EXPLANATION: [greet(name: str) -> None](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html): Prints a greeting message for the given name.
* [divide(a: float, b: float) -> float](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html): Returns the result of dividing [a](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) by [b](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html). Raises a [ValueError](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") if [b](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) is zero to prevent division by zero.
* [get\_even\_numbers(numbers: list[int]) -> list[int]](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html): Returns a list of even numbers from the provided list.

The example usage calls [greet("Alice")](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html), which prints a greeting for Alice.

TASK-6:

PROMPT: Add Google-style docstrings to this function. Each docstring should include a brief description of the function, parameter names with type hints, return values with type hints, and an example usage. Follow Google-style formatting and ensure the docstring accurately reflects the function's behavior

CODE:

OUTPUT:

EXPLANATION: This code defines a function [calculate\_total](vscode-file://vscode-app/c:/Users/Lasya/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") that computes the total cost of a list of item prices, including tax.  
It sums the item prices, calculates the tax by multiplying the sum by the tax rate, and adds the tax to the total.  
The example usage calculates the total price for items [15.0, 25.0, 35.0] with a 5% tax rate and prints the result.