## *2403A51028*

SANKU RUSHI REDDY

## Subgroup M

### M.1 — [S13M1] Stable sort employees by dept asc, salary desc

Scenario (agritech):

Context:

HR exports in agritech require deterministic sorting for payroll audits.

Your Task:

Sort employees by dept ascending and salary descending (stable), and re-emit CSV.

Data & Edge Cases:

name,dept,salary rows provided.

AI Assistance Expectation:

Use AI to outline csv.DictReader/Writer usage and key composition.

Constraints & Notes:

Stable sort within department by salary desc.

Sample Input

name,dept,salary  
Raj,Eng,120  
Maya,HR,90  
Abi,Eng,110

Sample Output

Raj,Eng,120  
Abi,Eng,110  
Maya,HR,90

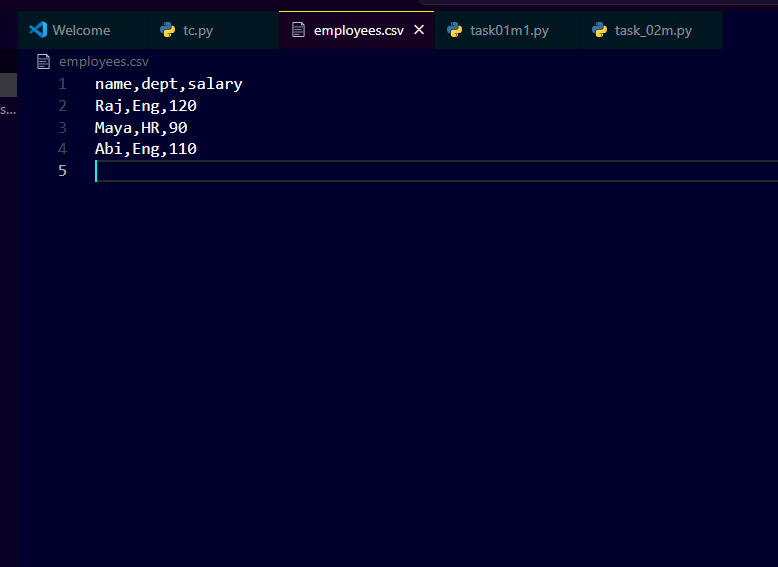
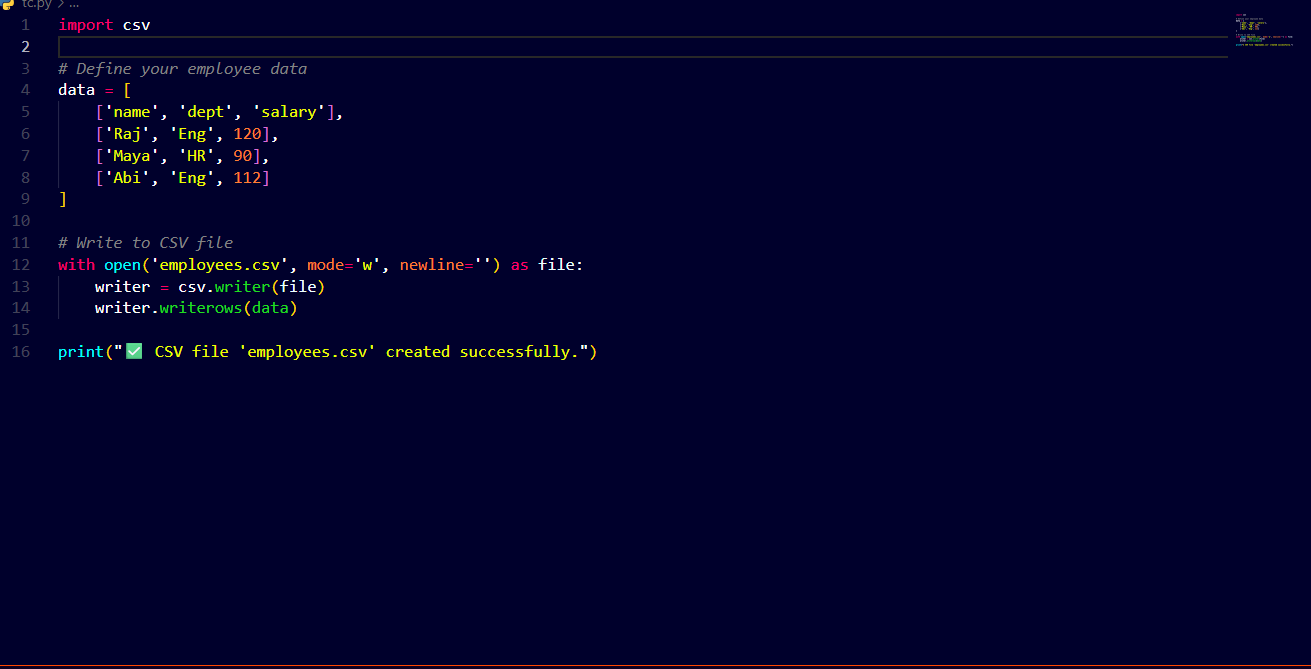
Acceptance Criteria: Stable and correct ordering

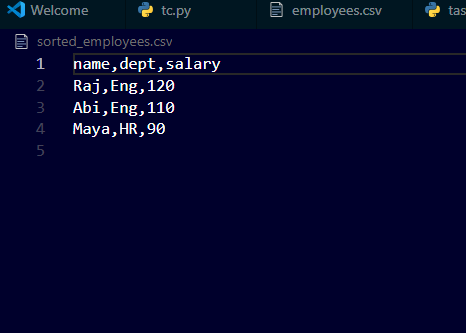
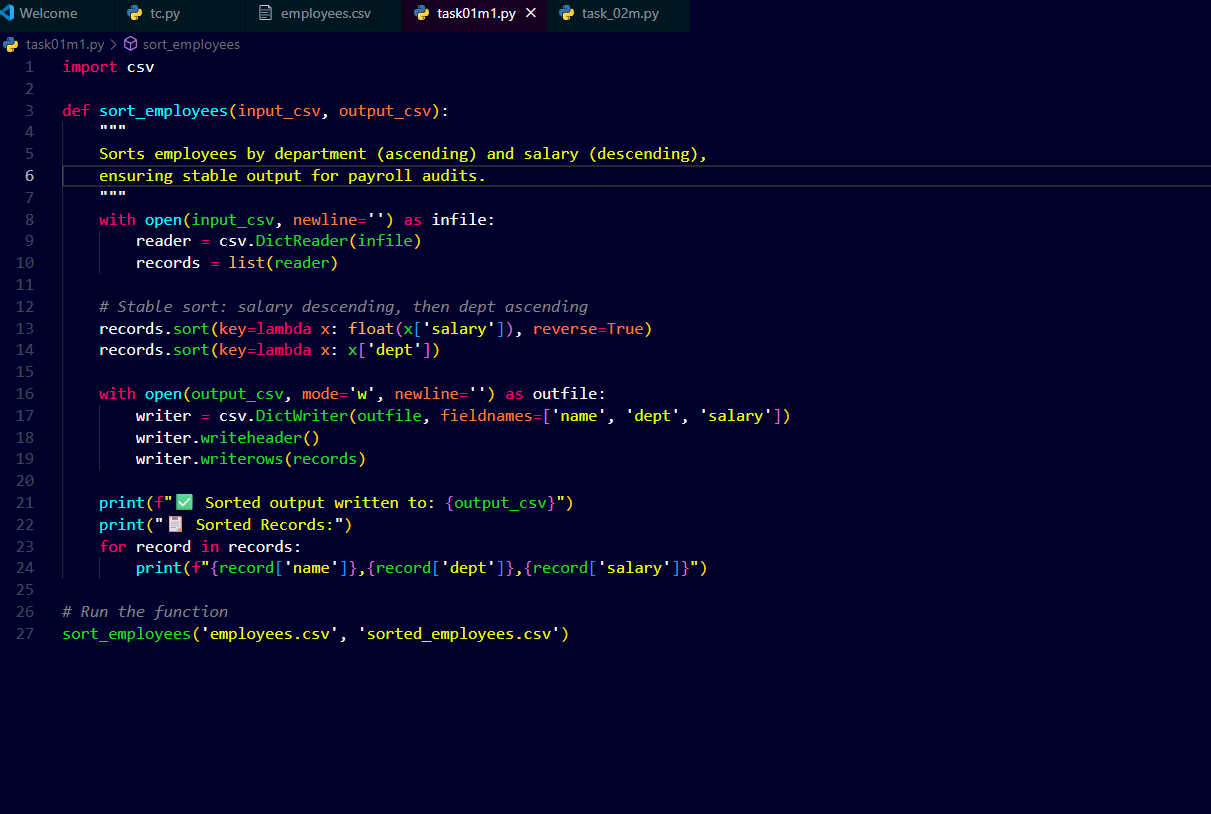
Prompt:

I have a CSV file with employee records in the format:  
name,dept,salary  
I need to sort the data for payroll audits in a deterministic way:

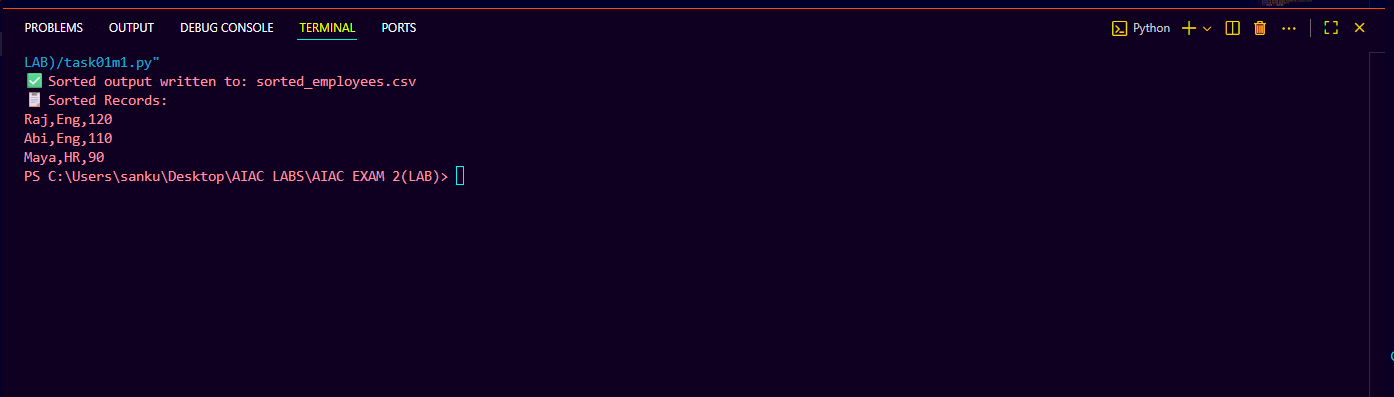
* Sort by dept in ascending order
* Within each department, sort by salary in descending order
* The sort must be **stable**, preserving original order for equal salaries

CODE:





OUTPUT:::



### M.2 — [S13M2] Validate Palindrome Function with Tests

**Scenario (AI-assisted debugging):**  
In an agritech text-analysis module, engineers need a function to check whether a given string is a palindrome (ignoring case and non-alphanumeric characters). A previous intern wrote a buggy version of the function.

**Your Task:**

1. Use AI-assisted coding tools (e.g., VSCode Copilot, Gemini, CursorAI) to:
   * Debug and correct the provided function.
   * Write **at least 5 unit tests** (positive, negative, edge cases).
2. Ensure the function passes all your test cases.

**Buggy Starter Code (given to students):**

def is\_palindrome(text: str) -> bool:

cleaned = ''.join(ch for ch in text if ch.isalpha())

return cleaned == cleaned[::-1]

**Data & Edge Cases:**

* Input may contain spaces, punctuation, and digits.
* Must be case-insensitive.
* Empty string should return True.

**Deliverables:**

* Corrected Python function.
* A test suite (you may use pytest or Python’s built-in unittest).
* Explanation of what bug(s) you fixed.

**Sample Input / Output:**

Input: "A man, a plan, a canal: Panama"

Output: True

Input: "hello"

Output: False

**Acceptance Criteria:**

* Handles mixed case + non-alphanumeric characters.
* Passes at least 5 meaningful test cases (normal, edge, tricky cases).
* Code must be readable and AI-generated documentation (docstring/comments) included.

Prompt:

I'm working on a Python function for a text-analysis module in an agritech project. I need help debugging a function that checks if a string is a palindrome.  
Requirements:

* Ignore case sensitivity
* Skip non-alphanumeric characters (punctuation, spaces, symbols)
* Return True for empty strings

Please:

1. Correct the buggy function below:

def is\_palindrome(text: str) -> bool:

cleaned = ''.join(ch for ch in text if ch.isalpha())

return cleaned == cleaned[::-1]

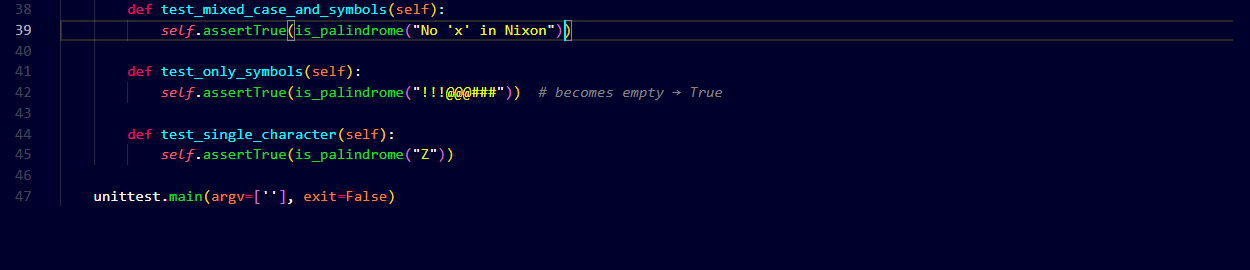
· Write at least 5 meaningful unit tests (positive, negative, edge cases) using unittest or pytest.

1. Include a docstring and inline comments explaining the logic and bug fixes.

Sample inputs:

* 1. "A man, a plan, a canal: Panama" → True
  2. "hello" → False
  3. "" → True

CODE:



OUTPUT::

