Lab 2

Task 1:
Scenario: You are managing game scores for a gaming application.

1. Create a NumPy array to store player names, game scores, and levels reached.
2. Update the score of the player who has reached the highest level.
3. Append a new player's data (name, score, level) to the array.
4. Insert a new player's score at a specific index within the array.
5. Delete the player with the lowest score from the dataset.

Task 2:
Scenario: You are analyzing website traffic data.

1. Create two NumPy arrays: one for daily visitors and another for pages viewed.
2. Create a shallow copy of the visitors array and a deep copy of the pages viewed array.
3. Concatenate both arrays to form a new array.
4. Split the concatenated array into two parts based on a threshold (e.g., more than 100 pages viewed).
5. Modify the original arrays and observe the changes in the copies.

Bonus Task: Git Commands
Scenario: You are collaborating on a project with multiple branches and need to manage complex changes effectively. Answer the following questions related to Git operations.

1. Describe the process to combine changes from a feature branch into the main branch. If you encounter a conflict during this process, what steps would you take to resolve it without losing any of the modifications?
2. Compare the two methods for integrating changes from one branch into another. Explain when it might be more advantageous to reapply commits from your feature branch on top of the main branch instead of combining them directly.
3. Outline how you would temporarily save your ongoing modifications without finalizing them. Under what circumstances would this approach be particularly useful? After saving your modifications, how would you restore them back to your working environment?
4. Discuss the differences in the effects on your working directory when you undo changes selectively, partially, or completely. In what situations would each method be most applicable, and what considerations should you keep in mind when using these approaches?