Exp:07 **Name: Hemanth kumar**

Roll no:231901010

Comparison of Two Files (Compare It)

Aim:

To compare two files using Compare It! and determine whether any modifications exist, then document the findings for forensic reporting.

Introduction:

In digital forensics, verifying whether two files are the same (or how they differ) is essential for detecting tampering, version drift, or malicious edits. Visual diff tools like Compare It! show changes side-by-side (insertions, deletions, edits) and can export an audit-friendly report. When the tool reports no differences, it supports integrity claims—ideally corroborated with cryptographic hashes.

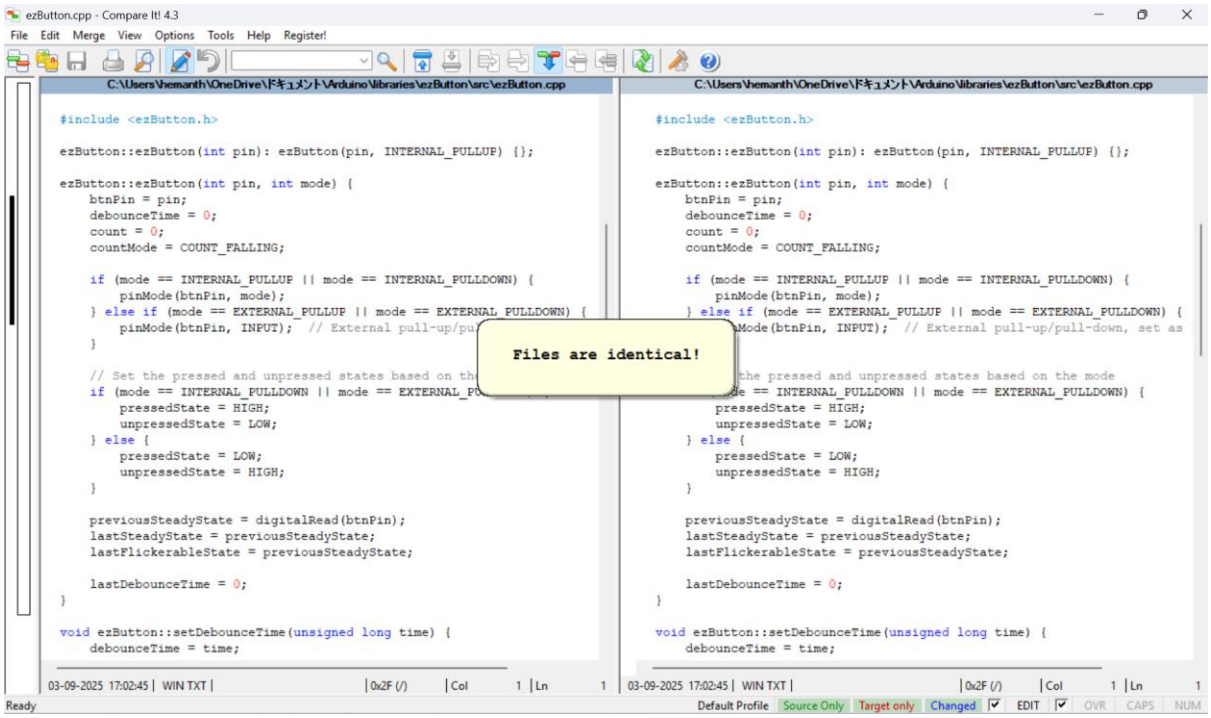
Procedure:

1. Launch Compare It!
    - Open the application (Windows).
2. Load files
    - File → Open Left → choose the first file (e.g., ezButton.cpp).
    - File → Open Right → choose the second file (e.g., backup/copy of ezButton.cpp).
3. Set comparison options (recommended)
    - Options → Compare: ensure defaults are appropriate.
    - For strict checks, disable “Ignore whitespace/case/line endings.”
    - For text-only intent, you may enable them to ignore trivial formatting.
4. Run the comparison
    - Click the Compare button (or press F9).
    - Use Next/Previous Difference arrows to navigate if any are found.
5. Evidence preservation (good practice)
    - Take a screenshot of the results view.

- File → Save Report (HTML/TXT) to export the diff summary.
- Compute and record hashes (e.g., SHA-256) of both files with a hashing tool and add them to the case notes.

Output (Observation):

- The tool displays both files side-by-side.
- A modal tooltip/message appears: “Files are identical!”
- No colored highlights or diff markers are shown; the status bar shows 0 differences.
- Line structure and content appear the same across both panes, confirming no textual changes under the chosen comparison settings.



Result

The two files compared in Compare It! are identical (no differences detected). Under the current comparison settings, there is no evidence of content modification. Integrity can be conclusively supported by matching cryptographic hashes of both files.