When you modify a file after staging it, here's what happens:

**1. Initial State:**

* You have a file called example.txt.
* You make some changes to example.txt.
* You stage the file with git add example.txt.

**2. Staging Area vs. Working Directory:**

* **Staging Area**: When you run git add example.txt, Git takes a snapshot of the file as it exists at that moment and stores this snapshot in the staging area. This version of the file is what will be committed to the repository when you run git commit.
* **Working Directory**: After staging the file, you might continue to edit example.txt. These changes are made in your working directory, which is the current state of the files on your filesystem.

**3. File Modification After Staging:**

* Let's say you modify example.txt again after running git add example.txt. Now, there are two versions of example.txt:
  + The **staged version**: This is the version of example.txt that you added to the staging area.
  + The **working directory version**: This is the version of example.txt that now reflects the latest changes you've made since staging.

**4. Git's Behavior:**

* Git recognizes that the version of example.txt in your working directory is different from the version in the staging area.
* As a result, git status will show example.txt as "modified." This indicates that while a version of the file is staged and ready to be committed, there are additional changes in the working directory that have not been staged yet.

**5. What This Means:**

* **Staged Changes**: If you commit at this point, only the version of example.txt in the staging area (the version you initially staged) will be committed.
* **Unstaged Changes**: The additional modifications in the working directory will not be included in the commit unless you stage them again using git add example.txt.

**Example:**

Let's go through an example step by step:

1. **Modify and Stage**:

echo "Initial content" > example.txt

git add example.txt

At this point, the staging area contains "Initial content."

1. **Modify Again**:

echo "Additional content" >> example.txt

Now, example.txt in the working directory contains both "Initial content" and "Additional content," but the staging area still contains only "Initial content."

1. **Check Status**:

git status

Git will show example.txt as "modified" because the working directory version is different from the staged version.

1. **Commit**:

git commit -m "Commit staged version"

The commit will only include "Initial content," not the additional content you added afterward.

To include the latest changes in the commit, you would need to run git add example.txt again before committing.