unlink()

The `unlink()` function removes a link of a name from the filesystem. If the name is the last link to a file and no process has it open, the file is deleted.

It's defined in the “unistd.h” system header.

It applies only to files, not directories. Use “unlikat()” for directories.

```
#include <unistd.h>

// "pathname" location of the file to unlink
int unlink(const char *pathname);
```

Returns 0 on success. In case of failure, returns -1 and sets “errno”.

- Files in Unix-like operating systems exist via links (directory entries)
- This function removes a directory entry (a link to a file)
- A file is deleted when:
    - All hard links are removed
    - No process is using the file
- “delete()” can delete files or empty directories, while “unlink()” is lower-level and only removes links

Example Usage

```
#include <stdio.h>
#include <unistd.h>

int main(void)
{
    char *file;

        file = "test.txt";
    if (unlink(file) == 0)
        printf("File '%s' deleted successfully.\n", file);
    else
        perror("unlink");
    return (0);
}
```