32.- ft_putstr_fd.-

BSD Man Page References:

- `write(2)`: Used for writing data to a file descriptor.
- `ft_strlen(3)` (custom, not standard): For calculating string length.

Synopsis:

```
void ft_putstr_fd(char *s, int fd);
```

Purpose:

Writes a null-terminated string to a specified file descriptor.

Description:

The `ft_putstr_fd` function takes two arguments:

- `s`: A pointer to the null-terminated string to be written.
- `fd`: The file descriptor (an integer representing a file or an output stream) to which the string will be written.

Code explanation:

```
void   ft_putstr_fd(char *s, int fd)
{
    if (!s) // Check for null input
    {
        return;
    }

    write(fd, s, ft_strlen(s)); // Use write to write the string
}
```

- It checks for a null `s` input and returns if encountered.
- It calls the `write` system call with:
  - `fd`: The file descriptor to write to.
  - `s`: The string to be written (pointer to the first character).
  - `ft_strlen(s)`: The number of bytes to write (length of the string).

Example usage (main function):

```
int main(void)
{
    ft_putstr_fd("Hello my friend\n", 1); // Write to standard output (FD 1)
    return (0);
}
```

Key points:

- `ft_putstr_fd` efficiently writes a string to a file descriptor, including the null terminator.
- It relies on `write` for the actual writing operation.
- It's useful for printing to different output streams beyond just standard output.

Additional notes:

- Ensure proper error handling when writing to file descriptors.
- Consider using `write(fd, s, strlen(s))` directly instead of relying on `ft_strlen` if performance is critical.