

You provided this code `"find /fs/ess/PAS2880/users/kolganovaanna -type f -name 'ERR10802885_R1.fastq.gz' 2>/dev/null"`. Please explain why what is the reason behind using `2>/dev/null` in this command?

[i](#) README.md:90

> Clarifying stderr and command usage...

I'll explain what `2>/dev/null` does, why it's often used with `find`, and the pros/cons — with a few concise examples you can copy.

Short answer

`2>/dev/null` redirects the process's standard error (file descriptor 2) to the special device `/dev/null` (discarding it). It hides error messages (e.g., "Permission denied") so your output only shows successful results written to stdout.

Why use it with find

- When you run `find /` or search large directory trees you often get many "Permission denied" messages on stderr.
- `find ... 2>/dev/null` removes that noise and leaves only matching paths on stdout.

What `2>/dev/null` means

- `0` = stdin, `1` = stdout, `2` = stderr.
- `2>/dev/null` = send stderr to `/dev/null` (discard it).
- `>file` or `1>file` would redirect stdout; `2>` specifically redirects stderr.

Examples

Hide permission errors:

```
find / -type f -name 'ERR10802885_R1.fastq.gz' 2>/dev/null
```

Keep stderr but save it to a file for inspection:

```
find / -type f -name 'ERR10802885_R1.fastq.gz' 2> find_errors.log
```

Show everything (no suppression — useful for debugging):

```
find / -type f -name 'ERR10802885_R1.fastq.gz'
```

Search a limited path (preferred when you know where to look — no need to suppress errors):

```
find ../../garrigos-data -name 'ERR10802885_R1.fastq.gz'
```

If you only want the parent directory printed:

```
find /fs/ess/PAS2880/users/kolganovaanna -type f -name 'ERR10802885_R1.fastq.gz' -printf '%h\n' 2>/dev/null | sort -u
```

Trade-offs / cautions

- Pro: Cleaner output — useful when permission-denied noise would otherwise drown results.
- Con: You may silently hide real errors (typos, filesystem problems). If something unexpected happens, redirect stderr to a file instead of `& null` so you can inspect it later.

When you probably don't need it

- When you search a directory you own (e.g., `../../garrigos-data` or your home tree) you normally won't get permission errors — so omit `2>/dev/null` for full transparency.

If you want, I can show a short demo that runs `find` with and without `2>/dev/null` in your environment so you can see the difference. Which would you like?