

`\par`

This command ends a paragraph and puts T_EX into vertical mode, ready to add more items to the page. Since T_EX converts a blank line in your input file into a `\par` token, you don't ordinarily need to type an explicit `\par` in order to end a paragraph.

An important point is that `\par` doesn't tell T_EX to start a paragraph; it only tells T_EX to end a paragraph. T_EX starts a paragraph when it is in ordinary vertical mode (which it is after a `\par`) and encounters an inherently horizontal item such as a letter. As part of its ceremony for starting a paragraph, T_EX inserts an amount of vertical space given by the parameter `\parskip` (p. '`\parskip`') and indents the paragraph by a horizontal space given by `\parindent` (p. '`\parindent`').

You can usually cancel any interparagraph space produced by a `\par` by giving the command `\vskip -\lastskip`. It can often be helpful to do this when you're writing a macro that is supposed to work the same way whether or not it is preceded by a blank line.

You can get T_EX to take some special action at the start of each paragraph by placing the instructions in `\everypar` (p. '`\everypar`').

See pages 283 and 286 of *The T_EXbook* for the precise effect of `\par`.

Example:

```
\parindent = 2em
‘‘Can you row?’’ the Sheep asked, handing Alice a pair of
knitting-needles as she was speaking.\par ‘‘Yes, a little%
---but not on land---and not with needles---’’ Alice was
starting to say, when suddenly the needles turned into oars.
```

produces:

```
“Can you row?” the Sheep asked, handing Alice a pair of knitting-
needles as she was speaking.
“Yes, a little—but not on land—and not with needles—” Alice was
starting to say, when suddenly the needles turned into oars.
```