


1

	<code>\big</code>	<code>\Big</code>	<code>\bigg</code>	<code>\Bigg</code>
	<code>\bigl</code>	<code>\Bigl</code>	<code>\biggl</code>	<code>\Biggl</code>
	<code>\bigm</code>	<code>\Bigm</code>	<code>\biggm</code>	<code>\Biggm</code>
	<code>\bigr</code>	<code>\Bigr</code>	<code>\biggr</code>	<code>\Biggr</code>

These commands make delimiters bigger than their normal size. The commands in the four columns produce successively larger sizes. The difference between `\big`, `\bigl`, `\bigr`, and `\bigm` has to do with the class of the enlarged delimiter:

- `\big` produces an ordinary symbol.
- `\bigl` produces an opening symbol.
- `\bigr` produces a closing symbol.
- `\bigm` produces a relation symbol.

T_EX uses the class of a symbol in order to decide how much space to put around that symbol.

These commands, unlike `\left` and `\right`, do *not* define a group.

Example:

```


$$\begin{aligned}
& \$(x) \quad \text{\bigl}(x\text{\bigr)} \quad \text{\Bigl}(x\text{\Bigr)} \quad \text{\biggl}(x\text{\biggr)} \quad \text{\Biggl}(x\text{\Biggr}) \\
& [x] \quad \text{\bigl}[x\text{\bigr}] \quad \text{\Bigl}[x\text{\Bigr}] \quad \text{\biggl}[x\text{\biggr}] \quad \text{\Biggl}[x\text{\Biggr}]
\end{aligned}$$


```

produces:

$$(x) \quad (x) \quad (x) \quad \left(x\right) \quad \left(x\right) \quad [x] \quad [x] \quad [x] \quad [x] \quad [x]$$