table-layout: Choosing the Column-Width Algorithm

CSS 2.1 §17.5.2 defines two ways to size a table's columns. `auto` (§17.5.2.2) measures every cell in every row and lets the widest content win its column. `fixed` (§17.5.2.1) never looks at cell content at all: a column takes its width from a `<col>` element, failing that from that column's cell in the *first row*, and every column left over splits the remaining space equally. Both tables below hold exactly the same markup and the same text.

1 — table-layout: auto (the default)

Item	Description	Status
SKU-1	A deliberately long description that pulls its own column wide, squeezing everything beside it - the column is sized from this cell's content.	OK
SKU-2	Short	OK

The middle column swelled to fit its longest cell; the outer two were squeezed down to roughly their own content width.

2 — table-layout: fixed

Item	Description	Status
SKU-1	A deliberately long description that pulls its own column wide, squeezing everything beside it - the column is sized from this cell's content.	OK
SKU-2	Short	OK

Same markup, same text: no cell was measured, so all three columns are exactly one third of the table each and the long description simply wraps inside its own column.

3 — fixed with <col> and first-row widths

#	Spanned header (50%, split over two columns)		Rest
1	25%	25%	remainder
2	ignored width: 90%	ignored	—

Column 1 takes its 15% from the `<col>`. The first row's `colspan="2"` header states 50%, divided evenly over the two columns it spans (25% each). The last column takes the 35% remainder (100% - 15% - 25% - 25%). The second row's 90% is ignored entirely - under fixed layout no row after the first can change a column's width.