When table content is unknown, we can pass entire table to specific tag. If we want to use columns from the table, we need to specify HEADER metadata. Table will respect specified width. Simple formatting plugin can be applied to entire object to get only top 10 rows.

|  |
| --- |
| [[Table1]:header:top10] |

When table columns are know we can use column names to specify location and formatting of columns. Style can be specified for each tag individually. More generic collection plugin must be applied to get only top 10 rows.

|  |  |  |
| --- | --- | --- |
| Nr. | Middle aligned | Special colors |
| [[Table2.Col1]:limit10] | [[Table2.Col2]] | [[Table2.Col3]] |

While selecting top 10 rows in that way works, its not as clean as using navigation plugin to do the same. Since v5 navigation plugin can be applied on a path so that a different object is passed down for processing. In this case it is done via navigation plugin – limit(N). To make it more readable tag path Table2:limit(10) can be shortened via alias – t2. Due to navigation, Table2 and Table2:limit(5) are treated as different collections.

|  |  |  |
| --- | --- | --- |
| Nr. | Middle aligned | Special colors |
| [[t2.Col1]] | [[Table2:limit(5).Col2]] | [[t2.Col3]] |

Templater supports merge-nulls metadata which instructs it to merge cells which contain null values. This works on both dynamic resize and normal tables.

|  |
| --- |
| [[DynamicResize]:merge-nulls] |

When merge-nulls is used in „normal“ tables, more style customization is available for the designer. If column does not contain merge-nulls metadata it will not be eligible for merging.

|  |  |  |
| --- | --- | --- |
| Col A | Col B | Col C |
| [[Nulls.1]:merge-nulls] | [[Nulls.2]:merge-nulls] | [[Nulls.3]:merge-nulls] |

Templater also supports span-nulls metadata which instructs it to merge cells vertically (unlike merge-nulls which merges them horizontally) which contain null values. This works on both dynamic resize and normal tables.

|  |
| --- |
| [[DynamicResizeAndMerge]:span-nulls] |

For scenarios when empty table needs to have a special row; two table template can be used. Section feature can be used to hide the appropriate table. Alternatively, specific Table4 tags can be placed inside table (but that will only hide the appropriate row(s).

|  |  |
| --- | --- |
| Name | Description |
| No results found | |

[[Table4]:collapseNonEmpty]

|  |  |
| --- | --- |
| Name | Description |
| [[Table4.Name]] | [[Table4.Description]] |

[[Table4]:collapseEmpty]

It's also possible to combine static table with dynamic resize, eg. only part of table has variable number of columns:

|  |  |  |
| --- | --- | --- |
| Beer | Description | [[Combined.Headers]] |
| [[Combined.Beers.Name]] | [[Combined.Beers.Description]] | [[Combined.Beers.Columns]] |

When we have a fixed size table (or fixed input spanning over few tables) we can utilize builtin **fixed** plugin to invoke a collection processing without a resize.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Quantity | Price |
| 1. | [[Fixed.Name]:fixed] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 2. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 3. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 4. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 5. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Quantity | Price |
| 6. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 7. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 8. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 9. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |
| 10. | [[Fixed.Name]] | [[Fixed.Quantity]] | [[Fixed.Price]] |