

# Easytable パッケージ

@monaqa

---

---

## 目次

---

---

1. Easytable パッケージの概要 .....	1
2. Gallery .....	2
2.1. 単純な表 .....	2
2.2. 罫線の指定 .....	5
2.3. その他の機能 .....	9
3. Easytable パッケージが提供する機能 .....	11
3.1. コマンド .....	11

---

---

## 1. Easytable パッケージの概要

---

---

Easytable パッケージは、`SATySFI` 上で楽に表を組むためのパッケージです。SATySF<sub>I</sub> には既に表を組むためのパッケージがいくつか標準で用意されていますが、いずれも単純な表を描くには少々複雑な構文を要求されるものでした。本パッケージは、単純であるもののある程度の汎用性を備えた、そんな表を楽に組めるようにするためのものです。対象を「ある程度単純な表」に絞っていることから察される通り、本パッケージで組める表には以下の制約があります。

- 左揃え、中央揃えなどの条件は列ごとに設定することができるが、セル単位で変更することはできない。すなわち、同一列の中で揃え方を変更することはできない。
- MultiCell (複数の行または列にわたるセル) は使えない。

しかし、世の中で必要とされる表の多くはこれらの機能を使わずに組むことができます。

ユーザが設定しなければならないパラメータを最小限に抑えることで API の複雑化を防ぎ、組みやすいコマンドを実現しています。

## 2. Gallery

具体例から説明したほうが理解が進みやすいと考え、具体的なコード例とともに出来上がりの表を載せています。以下では `easytable` パッケージが正常にインストールされており、かつ以下のようなコードで正常にインポートされているものとします。

```
@require: easytable
open EasyTableAlias
```

なお以下のコードをより簡潔にするため、上のコードでは `easytable` のインポートだけでなく、中にある `EasyTableAlias` モジュールを `open` することでその中身に直接アクセスできるようにしています。

### 2.1. 単純な表

本パッケージを用いて表を組むためには、以下のように `\easytable` コマンドを用います。

```
\easytable[l;c;r]{
| header1      | header2      | header3
| align left   | align center  | align right
| a            | b            | c
|}
```

header1	header2	header3
align left	align center	align right
a	b	c

このように、非常に単純な文法で表を組むことができます。`\easytable` は 2 つの必須引数を取り、1 番目で各列の体裁、2 番目で表の中身を指定します。1 番目は `cellfmt` 型と呼ばれる型のリストであり、 $n$  番目の要素が「左から数えて  $n$  番目の列をどのように揃えるか」を表しています。上の例では `[l; c; r]` を指定することで、列が左から順にそれぞれ左揃え、中央揃え、右揃えとなっています。2 番目はインラインテキスト型のリストです。引数として指定するのは 1 次元配列のようなデータ構造ですが、内部で 2 次元の構造（リストの

スト) へと変換されて表として組まれます。インラインテキストのリストは普通に書くならば  $\{a\}$ ;  $\{b\}$ ;  $\{c\}$  と表すことになりますが, SATySF<sub>I</sub> にはこれを  $\{a|b|c\}$  と表記する糖衣構文があります。 `\easytable` コマンドはこの構文を積極的に活用し, Markdown や AsciiDoc のような軽量マークアップ言語に近い書き易さを実現しています。

なお, 改行を含めた余剰なスペースは単にコードの見やすさのために入れているだけであり, たとえば以下のように書いても結果は変わりません。読みやすさ, 編集のしやすさなどを考えて調整するとよいでしょう。

```
\easytable[l;c;r]{| header1 | header2 | header3 | align left
| align center | align right | a | b | c |}
```

「結果は変わらない」ということから分かる通り, 上のコマンドから得られるのは 3 行 3 列の表です。すなわち, 列の数は第 2 引数の改行の位置ではなく, 第 1 引数のリストの長さによって決定されます。この仕様は意識しなければ非直観的な組版結果を生んでしまう原因になるため注意してください。

第 2 引数に渡すのはインラインテキストですから, インラインコマンドを用いた装飾や, 数式の挿入も可能です。(使う機会があるかはともかく) 表の入れ子もできます。

```
\easytable[r; c]{
|           | 例
| Emph    | \emph{強調されたテキスト}
| Code    | \code(`some code`);
| Math    |  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ 
| Inner table |
\easytable[l; r]{
| header 1 | header 2
| a        | b
|}
|}
```

例	
Emph	強調されたテキスト
Code	some code
Math	$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$
Inner table	header 1   header 2
	a                      b

1 番目に渡す `cellfmt` 型のリストに 1 ではなく `lw (length)` を渡すことによって、表の横幅を指定することができます。このとき、表の横幅よりも長いインラインテキストが表の中にあれば、`SATySFI` の本体で用意されている行分割アルゴリズムに従って行分割されます。表の中に長い文章を書く場合に便利です。

```
\easytable[l; lw 120pt; lw 120pt;]{
| Column 1 | Column 2     | Column 3
| 通常の列 | 横幅 120pt
| 横幅 120pt で、なおかつ横幅よりも長いテキストが入っている場合
|}
```

Column 1	Column 2	Column 3
通常の列	横幅 120pt	横幅 120pt で、なおかつ 横幅よりも長いテキスト が入っている場合

同様に列幅を指定できるオプションとして、中央揃えの `cw` と右揃えの `lw` も用意されています。ただし、これらのオプションを指定しても、長いテキストを入れたときに段落が中央揃えや右揃えになるわけではなく、段落内は全て左揃えで組まれます\*1。その点には注意が必要です。

```
\easytable[lw 120pt; cw 120pt; rw 120pt;]{
```

1 これはどちらかというと、`SATySFI` で単なるインラインテキストを右揃えや中央揃えの段落にする汎用的な手段が現状用意されていないのが原因です。

```
| Align left | Align center | Align right
| short text | short text    | short text
| 横幅 120pt で、なおかつ横幅よりも長いテキストが入っている場合
| 横幅 120pt で、なおかつ横幅よりも長いテキストが入っている場合
| 横幅 120pt で、なおかつ横幅よりも長いテキストが入っている場合
|}
```

Align left	Align center	Align right
short text	short text	short text
横幅 120pt で、なおかつ 横幅よりも長いテキスト が入っている場合	横幅 120pt で、なおかつ 横幅よりも長いテキスト が入っている場合	横幅 120pt で、なおかつ 横幅よりも長いテキスト が入っている場合

2.2. 罫線の指定

表において、罫線は視覚的に重要な意味を持つと考えられます。今までの例にて見た通り、デフォルトでは表の上下に太い線を、そして 1 行目の下に細い線を引いています。これは 1 行目がヘッダ行を、それ以降の列がコンテンツという標準的な表に対しては自然なレイアウトとなりますが、場合によってはヘッダ行がなかったり、ヘッダ列があったりするような表も描きたいことが当然考えられます。

本パッケージではそのような需要にもできるだけ簡単なインターフェースで対応できるよう、オプション引数で罫線を制御できるようにしています。オプション引数は `ruleptn` と呼ばれる型のリストであり、引きたい罫線の種類を羅列します。まず最も単純な場合としてオプション引数に空のリストを与えた場合、罫線が全く引かれない表が組まれます。

```
\easytable?:[] [r; c; l]{
| How      | I        | want
| a        | drink    | alcoholic
| of       | course   | after
| the      | heavy    | lectures
| involving| quantum  | mechanics
|}
```

How	I	want
a	drink	alcoholic
of	course	after
the	heavy	lectures
involving	quantum	mechanics

やはり少し寂しいので，上下に罫線を引くことにしましょう．オプション引数のリストに `t` を追加すると上に，`b` を追加すると下に，それぞれ太い罫線が引かれます．

```
\easytable?:[t][r; c; l]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
\easytable?:[t; b][r; c; l]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```

How	I	want	How	I	want
a	drink	alcoholic	a	drink	alcoholic
of	course	after	of	course	after
the	heavy	lectures	the	heavy	lectures
involving	quantum	mechanics	involving	quantum	mechanics

表の途中に水平線を引きたい，というのもよくあることです．その場合には `m` (`int`) を用いることができます．`m` の引数は整数であり，正の数  $n$  を指定すれば  $n$  行目の真下に線が引かれます．また，負の数  $-n$  を指定したときは，最後から数えて  $n$  行目の真下に線が引かれます．

```
\easytable?:[t; b; m 2][r; c; l]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```

```
\easytable?:[t; b; m (-2)][r; c; 1]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```

How	I	want	How	I	want
a	drink	alcoholic	a	drink	alcoholic
of	course	after	of	course	after
the	heavy	lectures	the	heavy	lectures
involving	quantum	mechanics	involving	quantum	mechanics

ここまでくれば予想がついたかもしれませんが、オプション引数のデフォルト値は `[t; b; m 1]` です。

列を分けるために縦に線を引くスタイルも、日本では特によく見かけます. `v (int)` で列の境目に鉛直方向の線を引くことができます. 引数の意味は `m` のときと同様、左から数えて何番目に線を引くかを表します。

もう少し自由度の高い罫線が引きたい場合は、`d (int, int) (int, int)` を指定します。

```
\easytable?:[t; b; d (0, 0) (1, 1); d (3, 1) (1, -1)][r; c; 1]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```

How	I	want
a	drink	alcoholic
of	course	after
the	heavy	lectures
involving	quantum	mechanics

表全体を太い外枠で囲むときは `rect` を使うことができます。

```
\easytable?:[rect][r; c; 1]{
| How | I | want | a | drink | alcoholic | of | course
```

```
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```

How	I	want
a	drink	alcoholic
of	course	after
the	heavy	lectures
involving	quantum	mechanics

罫線とは違いますが、オプション引数では表の背景色を指定することもできます。表全体の背景色を指定するには `bg-a (color)` を、特定の範囲の列の背景色を指定するには `bg-c (color) (int) (int)` を、特定の範囲の行の背景色を指定するには `bg-r (color) (int) (int)` を用います。罫線、背景色ともに指定された順にグラフィックスが上書きされていきます。従って、罫線を引きたいときは背景色を指定した後に罫線を指定するのがよいでしょう。

```
\easytable?:[bg-c (Color.red) 1 2][r; c; l]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
\easytable?:[
bg-a (Color.gray 0.8);
bg-c (Color.red) 1 2;
bg-r (Color.yellow) 1 3;
t; b; v 1;
][r; c; l]{
| How | I | want | a | drink | alcoholic | of | course
| after | the | heavy | lectures | involving | quantum | mechanics
|}
```



How	I	want	How	I	want
a	drink	alcoholic	a	drink	alcoholic
of	course	after	of	course	after
the	heavy	lectures	the	heavy	lectures
involving	quantum	mechanics	involving	quantum	mechanics

## 2.3. その他の機能

長い文章を入れたい場合などは、箇条書き形式で表に要素を入れることのできる `\list-table` コマンドが便利です。

```
\list-table?:[t; b; m 1; v 1; v 2][rw 90pt; l; lw 180pt;]{
*
** Treat
** Quantity
** Description

*
** Albatross
** 2.99
** On a stick!

*
** Crunchy Frog
** 1.49
** If we took the bones out,
    it wouldn't be crunchy, now would it?

*
** Gannet Ripple
** 1.99
** On a stick!
}
```

Treat	Quantity	Description
Albatross	2.99	On a stick!
Crunchy Frog	1.49	If we took the bones out, it wouldn't be crunchy, now would it?
Gannet Ripple	1.99	On a stick!

以下は単純な例で、なんとなく数学書の最初に載っていそうな表です。これぐらいシンプルな表は、やはりシンプルに書きたいものです。

```
\easytable?:[t; b][c; l]{
|  $\mathbb{N}$  | 自然数全体の集合
|  $\mathbb{Z}$  | 整数全体の集合
|  $\mathbb{Q}$  | 有理数全体の集合
|  $\mathbb{R}$  | 実数全体の集合
|  $\mathbb{C}$  | 複素数全体の集合
|}
}
```

$\mathbb{N}$	自然数全体の集合
$\mathbb{Z}$	整数全体の集合
$\mathbb{Q}$	有理数全体の集合
$\mathbb{R}$	実数全体の集合
$\mathbb{C}$	複素数全体の集合

以下も比較的単純な表です。easytable パッケージでは、表の格子点から格子点をつなぐ線をひくことができます。これを用いることで、表の左上によく見るバックスラッシュを引くことができます。

```
\easytable?:[
d (0, 0) (1, 1);
v 1; h 1;
rect;
```

```

][c; c; c; c;]{
  |      | ${e} | ${g} | ${h}
  | ${e} | ${e} | ${g} | ${h}
  | ${g} | ${g} | ${h} | ${e}
  | ${h} | ${h} | ${e} | ${g}
  |}
\easytable?:[
  bg-r (Color.gray 0.9) 0 1;
  bg-c (Color.gray 0.9) 0 1;
  v 1; h 1;
  rect;
][c; c; c; c;]{
  |      | ${e} | ${g} | ${h}
  | ${e} | ${e} | ${g} | ${h}
  | ${g} | ${g} | ${h} | ${e}
  | ${h} | ${h} | ${e} | ${g}
  |}

```

	<i>e</i>	<i>g</i>	<i>h</i>			<i>e</i>	<i>g</i>	<i>h</i>
<i>e</i>	<i>e</i>	<i>g</i>	<i>h</i>	<i>e</i>	<i>e</i>	<i>g</i>	<i>h</i>	
<i>g</i>	<i>g</i>	<i>h</i>	<i>e</i>	<i>g</i>	<i>g</i>	<i>h</i>	<i>e</i>	
<i>h</i>	<i>h</i>	<i>e</i>	<i>g</i>	<i>h</i>	<i>h</i>	<i>e</i>	<i>g</i>	

### 3. Easytable パッケージが提供する機能

#### 3.1. コマンド

本パッケージが提供するコマンドは2つです. `ruleptn` 及び `cellfmt` 型の説明は後述します.

- `\easytable?:[rule-pattern][cell-format]{| contents |}`  
`{| contents |}` の内容を表にします.

引数	型	種類	内容
<code>rule-pattern</code>	<code>ruleptn list</code>	Optional	罫線の設定
<code>cell-format</code>	<code>cellfmt list</code>	Needed	$n$ 番目の列の体裁 (揃え方)
<code>contents</code>	<code>inline-text list</code>	Needed	表の中身

- `\list-table?:[ruleptn list][cellfmt list]{* itemize }`  
`{* itemize }` の内容を表にします.

引数	型	種類	内容
<code>rule-pattern</code>	<code>ruleptn list</code>	Optional	罫線の設定
<code>cell-format</code>	<code>cellfmt list</code>	Needed	$n$ 番目の列の体裁 (揃え方)
<code>contents</code>	<code>itemize</code>	Needed	表の中身

もう少し具体的な使い方は、後の具体例を見たほうが早いでしょう.