

reStructuredText

[Top](#) / reStructuredText

reStructuredText

[かぎマニュアルズ](#) >

[reStructuredText](#) とは、テキストを書く際の約束ごとのようなものです。実体はテキストファイルです。メモ帳でも簡単に作成できます。普通のテキストファイルとちよつと違うのは、構造化情報を持たせることができる、という点です。

物理のかぎプロジェクトでは、この [reStructuredText](#) に基づいたテキストを記述し、記事を制作します。そのために、ある程度の仕組みと書式を覚えていただく必要があります。といっても、普通に記事を書く分にはそれほど難しくありませんので、このページや [物理のかぎ記事チュートリアル](#) を参照しながら、気軽に取り組んで行って下さい♪

-
- [reStructuredText](#)
 - [どうやってつくるのか](#)
 - [構造化とは](#)
 - [リファレンス](#)
 - [注意](#)
 - [章立て](#)
 - [段落](#)
 - [リスト](#)
 - [番号付きリスト](#)
 - [定義リスト](#)
 - [文字装飾](#)
 - [強調その1](#)
 - [強調その2](#)
 - [上付き文字](#)
 - [下付き文字](#)
 - [リンク](#)
 - [空白を含む文字列へのリンク](#)
 - [ページ内の相互参照](#)
 - [表組み](#)
 - [画像](#)
 - [画像の縦横などの詳細を含める](#)
 - [インライン画像](#)
 - [キャプション](#)
 - [注釈](#)
 - [HTMLの命令を直接記述する](#)

どうやってつくるのか

メモ帳などの普通のテキストエディタで、普通に保存すればいいです。拡張子は気にしなくてかまいませんが、テキストファイルですから「.txt」としておけば良いでしょう(普通は勝手にそうなりますね)。重要なのは書き方です。

そしてその、[reStructuredText](#) という約束に基づいたテキストファイルを、たとえばHTMLに変換すると、その約束に基づいた構造が保たれたまま変換されます。変換には [rst2hooktail](#) をお使いください。ただしこれは、TeX数式処理やHTMLの表示スタイルなど、物理のかぎプロジェクトぽ用にカスタマイズが施されています(文書構造そのものは [reStructuredText](#) に基づいています)。

構造化とは

「Structured Text」とは「構造化されたテキスト」の意味ですが、厳密な構造ではなく、自由度の高い、ある一定のパターンを含んでいるというべきでしょうか。そのパターンを解析するコンバータで変換することで、XHTMLやLaTeXフォーマットのような厳密な構造を出力できます。要するに気の利いたテキストファイルです。

リファレンス

[reStructuredText](#) にしたがって文書を書く際の約束を、よく使うであろうものにかぎって説明します。さらに詳しくは [はやわかりreStructuredText](#) や [公式マニュアル](#) (残念ながら英語ですが、最も情報量が多いです)を参照してください。

注意

- 以下の説明で、「スペース」とは半角スペースのことです。数字や「.」、「_」、「:」などの記号も全て半角文字です。
- 以下の説明で、赤字 で書いたものが [reStructuredText](#) の命令です。
- 特別な理由がない限り、行の先頭にはスペースを含めないでください。行頭のスペースは [reStructuredText](#) の命令の一つですので、思わぬ結果を招くことになります。

章立て

章立てとは、文書の構造を保つためになくてはならないものです。最低限、「タイトル」、「節」、「小節」くらいを覚えておきましょう。 [reStructuredText](#) では、

```
=====
タイトル
=====
```

```
-----
セクション 1
-----
```

```
~~~~~
サブセクション 1.1
~~~~~
```

```
~~~~~
サブサブセクション 1.1.2
~~~~~
```

```
~~~~~
サブセクション 1.2
~~~~~
```

```
~~~~~
サブサブセクション 1.1.2
~~~~~
```

```
-----
セクション 2
-----
```

```
~~~~~
サブセクション 2.1
~~~~~
```

サブセクション 2.2

```
@@author:@@
@@accept:@@
```

- [この例の変換結果](#)

という感じで、章立てを記述します。ご覧の通り、見出し文の下部に「----」を付けたりして、その文が見出しであることを示します。直感的にも分かりやすい表記法ですね。

必ずしもこの例で示した通りの記号を用いる必要はありません。が、一つの文書において同じレベルの見出しには、同じ記号を用いる必要があります。さらに細かくセクション分けを加える場合は、同様に適当な見出し記号を追加します。通常はサブセクションくらいまでで十分でしょう。

上の例の最後に

```
@@author:@@
@@accept:@@
```

という部分があります。ここに、書いた人の名前、正式公開日の日付を

```
@@author: 崎間@@
@@accept: 2005-01-22@@
```

などというふうに記入します。acceptの部分はいつになるか分からないので、査読に提出する際は@@accept:@@のまま構いません(@@author:@@と@@accept:@@の二つの命令は、物理のかぎプロジェクト独自の拡張です)。

以上が、文書作成の骨組みになります。

段落

連続した行は、途中で改行が入っても一つの段落とみなされます。

```
という感じで、章立てを記述します。
ご覧の通り、見出し文の下部に「----」を付けたりして、
その文が見出しであることを示します。
```

は一つの段落として、途中で改行を含めず表示されます。

```
という感じで、章立てを記述します。
ご覧の通り、見出し文の下部に「----」を付けたりして、
その文が見出しであることを示します。
```

のように、途中で空行が入ると、その上と下は違う段落として表示されます。

リスト

リスト(箇条書き)するには、

```
- このように
- 書くと
- 箇条書きされます
```

のように、行頭を - で書きます。この他の記号として、* と + が使えます。

番号付きリスト

リストの先頭に番号を付けたい場合,

- ```
1. このように
2. 書くと
3. 箇条書きされます
```

のように, 先頭を 数字. と書きます (行頭にはスペースを入れないでください). ピリオド後ろにはスペースが必要です. また,

- ```
3. このように
4. 書くと
5. 箇条書きされます
```

のように, 中途半端な番号からはじめることもできます.

定義リスト

定義リストとは「用語」と「その説明」をリストするものです.

```
用語1
ここに用語の説明を書きます

用語2
ここにその用語の説明を
書いて行きます.
```

のように, 説明部分先頭にスペースを入れ, インデントを設けて記述します.

文字装飾

強調その1

強調したい文字列の両側を

```
地の文地の文地の文 *強調したい文字列* 地の文地の文地の文
```

のように * で括ると, 変換後はその部分の文字列が

```
地の文地の文 強調したい文字列 地の文地の文
```

と, 太字になって強調されます. このとき, * と地の文の間にはスペースが入っている必要があるので, 注意してください.

強調その2

さらに強調したい場合は,

```
地の文地の文地の文 **強調したい文字列** 地の文地の文地の文
```

のように, 強調したい文字列を ** で括ります. すると, 変換後は

```
地の文地の文 強調したい文字列 地の文地の文
```

```
Java™:sup:
```

というふうに太字赤色になり、少し大きいサイズで表示されます。

上付き文字

```
Java `TM`:sup:
```

とすると、Javaの右肩にTMが乗ります。

下付き文字

```
v `1`:sub:
```

とすると、vの右下に1が付きます。

リンク

文字にリンクを張りたい場合

```
ちなみにこういった任意定数は、初期条件によって決まります。
この例題について詳しくは、物理のかぎしっぽ_をご覧ください。
```

```
.. _物理のかぎしっぽ: http://www12.plala.or.jp/ksp/
```

のように、リンクを張りたい文字列に

```
物理のかぎしっぽ_
```

と最後に「_」を付け、そのリンク先を

```
.. _物理のかぎしっぽ: http://www12.plala.or.jp/ksp/
```

どうふうに記述します。リンク先記述部分は、文書中のどこにあっても構いません。リンクを張りたい文字列と地の文との間には、スペースを入れた方が読みやすいでしょう(必ずしも入れる必要はありません)。

空白を含む文字列へのリンク

リンクにしたい文字列が「my documents」や「式(1)」のように、空白文字(スペース)や記号「()」などを含む場合、その文字列を

```
`my documents`_
```

や

```
`式(1)`_
```

のようにバッククォート(Shiftキー + @キー)で括ります。

ページ内の相互参照

「式(1)」をクリックしたら、該当の数式へ飛ばすような場合など、ページ内にリンクを張りたいことがあります。それには

```
.. _eq1:
```

```
<tex>
F=G¥frac{Mm}{r^2} ¥tag{1}
</tex>
```

というふうに, .._ターゲット名: に続けて数式(や文字列)を書きます. この例では「eq1」がターゲット名です. これで

```
<tex>
F=G¥frac{Mm}{r^2} ¥tag{1}
</tex>
```

が変換された画像(の先頭部分)がリンクターゲットになります. このターゲットへリンクを張るには

```
ここで `式(1)` になんとか
.._式(1): #eq1
```

というふうに, .._ターゲット1: #ターゲット2 とします. これで「ターゲット1」をクリックすると「ターゲット2」に飛びます. この例では「式(1)」という文字をクリックすると, 上で設定した数式に飛ぶようになります.

表組み

表をつくりたい場合,

```
+-----+-----+-----+
| 見出し 1 | 見出し 2 | 見出し 3 |
+-----+-----+-----+
| 内容     | 内容     | 内容     |
+-----+-----+-----+
| 内容     | 複数セルにわたる内容 |
+-----+-----+-----+
```

のように, テキストで表した簡易的な表組みをしておきます.

画像

文書に画像を挿入するには, 挿入したい場所に

```
.. image:: 画像ファイル名
```

と書きます. .. image:: が, そこに画像を挿入するという命令です.

画像の縦横などの詳細を含める

挿入する画像について, 幅や代替テキストなどの詳細を含めるには

```
.. image:: 画像ファイル名
   :height: 100
   :width: 200
   :alt: 代替テキスト
```

とします. heightおよびwidthはピクセル値です.

インライン画像

インライン(文中)に画像を含めることもできます. それには

```
この |imgname| という画像は, なになにです.
```

```
.. |imgname| image:: 画像ファイル名
```

のように|ターゲット名|という命令でターゲットを設定しておきます。この例では「imgname」がターゲット名です。そして.. |ターゲット名| image:: 画像ファイル名 という命令で、文中のターゲット部分に画像を挿入します。

キャプション

図の下にキャプション(簡単な説明文)を入れたい場合 .. figure:: を使い

```
.. figure:: 画像ファイル名
   キャプション
```

のようにしてください。キャプションの文字が少し小さくなり、文字列の左端が画像の左端と揃えられます。.. figure:: 行の下に空白行が入ることに気を付けてください。

注釈

本文から逸れる内容は注釈として記述します。注釈を加えるには

```
この引っ張る力が重力 [*]_ であり,
.. [*] 重力についての注釈
```

のように、[*]_ を注釈を加えたい部分に記述し、.. [*] の後ろにその内容を記述します。注釈部分が複数行に渡る場合、先頭にスペースを入れてインデントします。

これらの命令と地の文の間にはスペースが必要です。注釈は .. [*] を記述した部分に表示されます。通常、[*]_ を書いた段落のすぐ後に表示されるようにすると良いでしょう。

複数の注釈を加えたい場合も同様で、

```
この引っ張る力が重力 [*]_ であり、この力の法則が、万有引力 [*]_ です。
.. [*] 重力についての注釈
.. [*] 万有引力についての注釈
```

のように[*]_ と .. [*] 命令を使います。.. [*] 命令を書いた順番に対応づけられます。

HTMLの命令を直接記述する

どうしてもreStructuredTextで用意されている形式で扱えないもの、たとえばJava AppletやFlashを記述したい場合、その命令(HTMLタグ)を直接reStructuredTextのソースに記述することになります。

そのためには、.. raw:: html という命令(rawディレクティブ)を使用します。

```
.. raw:: html
   <hr width=50 size=10>
```

のように書いた場合、

```
<hr width=50 size=10>
```

がHTMLタグとして記述されます。すなわちこの記述部分に、短い水平線が表示されます (rawディレクティブを用いず単に `<hr width=50 size=10>` と書いた場合、その文字列がそのまま出力されます)。

Java AppletやFlashを記事に埋め込みたい場合、上記の要領でHTMLの命令を直接記述してください。

この命令は文字の装飾や段落の装飾などには用いないで下さい。

[かぎマニュアルズ](#) >

最終更新: Sat, 16 Apr 2005 16:42:07 JST (1d)