

1. ページ番号

文書内でページ番号がリセットされるということがあります。例えば前付きでは独立なページ数を設定しておき、本文ではまた 1 から始まる、という具合です。さらに、ページ数の出力も前付きではローマ数字だが本文内ではアラビア数字になる、というように、出力形式が変わることもあります。本パッケージ内の `PageNumber` モジュールを使うと以下のページ数の管理が行えます。

- 真のページ数：先頭ページから常に一つずつ増えていくページ数。
- みかけのページ数：必要に応じてリセットされているかもしれないページ数。ノンブルとして出力されるのはこっち。
- ページ数文字列：実際に出力されるページ数を文字列で表したもの。

まず `PageNumber` モジュールを読んでおきます。

```
@require: pagestyle/pagnumber
```

現在のみかけのページ数を変更するには、`set-page-number` を使います。

```
let page-number-b = PageNumber.set-page-number pn % このページのページ数
を pn に変更
% page-number-b は inline-nil 相当の inline-boxes
% これを本文に埋め込む。
```

文書内で変更するための `\set-page-number` もあります。

```
+p{
...
\PageNumber.set-page-number(1);
...
}
```

ページ出力形式は `int -> string` という関数です。これは `set-page-format` で設定します。

```
let page-number-b = PageNumber.set-page-format arabic
% 本文に埋め込むのを忘れないように。
```

現在のみかけのページ数を取得するには `get-page-number` を、現在のページ数文字列を取得するには `get-page-string` を使います。引数には真のページ数を与えます。`page-break` プリミティブの第二引数に渡される引数を `pbinf` とすると `pbinf#page-number` で取得できます。

```
let page-number = PageNumber.get-page-number pbinf#page-number
let page-str = PageNumber.get-page-string pbinf#page-number
```