

# 大学生と研究者の L<sup>A</sup>T<sub>E</sub>X 入門

言語学・応用言語学専攻

学籍番号: XXXXXXXXX

2024 年度入学

加藤幹治 (jiateng.ganzhi@gmail.com)

最終更新: 2025 年 1 月 7 日

# 目次

目次	i
<b>1 はじめに</b>	<b>1</b>
1.1 筆者について .....	1
1.2 L <sup>A</sup> T <sub>E</sub> X ってなに? .....	1
1.3 L <sup>A</sup> T <sub>E</sub> X を使うと何がいいの? .....	2
1.3.1 数式や例文が綺麗に出力できる .....	2
1.3.2 参考文献の引用がとても楽 .....	3
1.3.3 ユーザによる拡張性が高い .....	3
<b>2 L<sup>A</sup>T<sub>E</sub>X の基本的な使い方</b>	<b>3</b>
2.1 エディタの使い方 .....	3
2.2 簡単な文書を作ってみよう .....	4
2.3 文字の処理 .....	5
2.3.1 改行 .....	5
2.3.2 段落 .....	5
2.3.3 空白 .....	6
2.4 文字の装飾 .....	7
2.5 章立て .....	9
<b>3 ほげほげ</b>	<b>9</b>
<b>3 スタイルファイル (拡張パッケージ) の導入</b>	<b>9</b>
3.1 あらかじめダウンロードされているものの導入 .....	9
3.2 まだパソコンに入っていないものの導入 .....	10
<b>4 論文で使うあれこれ</b>	<b>10</b>
4.1 表の挿入 .....	10
4.1.1 普通の表 .....	10
4.1.2 セルの結合 .....	12
4.2 図の挿入 .....	13
4.3 相互参照 .....	14
4.4 脚注の挿入 .....	14
<b>5 参考文献の引用</b>	<b>15</b>

5.1	BiBTeX の使い方 .....	15
5.2	書誌情報を入力する .....	15
5.3	引用する .....	16
5.4	引用スタイルを指定する .....	17
5.4.1	natbib.sty を導入する .....	17
5.4.2	jecon.bst を導入する .....	17
5.5	書誌情報が入ったファイルを指定する .....	17
5.6	タイプセットする .....	17
5.7	文献引用のまとめ .....	18
<b>6</b>	<b>言語学用の設定</b> .....	<b>19</b>
6.1	例文の挿入 .....	19
6.2	樹形図の挿入 .....	20
6.3	IPA の挿入 .....	21
<b>7</b>	<b>FAQ</b> .....	<b>22</b>
	<b>参考文献</b> .....	<b>23</b>
<b>8</b>	<b>Abbreviations</b> .....	<b>23</b>

# 1 はじめに

この文書では、「卒論やレポートを綺麗に書きたい」「参考文献をいちいち書くのが面倒」「国際音声記号を使いたい」などなど、Word に限界を感じた大学生・大学院生 (特に人文科学系) のために、 $\text{\LaTeX}$  という文書作成システムの使い方を説明します。

既に  $\text{\LaTeX}$  がインストールされているという前提で話を進めますので、まだインストールが終わっていない方は、筆者が作成したマニュアルなどをご覧の上インストールを済ませてください。

## 1.1 筆者について

筆者は東京外国語大学の大学院生です。言語学を専攻しているので言語学関係の  $\text{\LaTeX}$  の処理にはそこそこ慣れ親しんでいますが、それ以外の分野に関しては記述が不十分な場合があります。また、筆者は Mac ユーザなので、それ以外の OS に関しても記述が不十分な場合があります。

この文書についてなにか質問があれば、[jiateng.ganzhi@gmail.com](mailto:jiateng.ganzhi@gmail.com) までメールを送っていただければ、筆者の分かる範囲で回答いたします。また、水曜日 14 時から 16 時と木曜日の 15 時から 19 時までは東京外国語大学の図書館 4 階「学習相談デスク」に座っていますので、直接聞きに来て頂いても構いません。

## 1.2 $\text{\LaTeX}$ ってなに?

$\text{\LaTeX}$  (ラテフ、ラテク、レイテック、ラテックスなどと読む) は、文書作成システムの名前です。Word のような文書作成ソフトというよりは、命令を入力して結果を出力するプログラミング言語のような感じです。

実際どのようにして文書を作っているかを、図1に示しています。左半分がエディタ画面で、右半分はそのプレビュー (このページ) です。



図1  $\text{\LaTeX}$  の編集画面 (MacOS)

編集画面で打ち込んだ文字がそのまま印刷結果になるのではない、というところが Word などのワープロソフトとの大きな違いです。自分が得たい出力結果を出すためには、このプログラミングのような書き方を覚えなければいけません。ちょっと敷居は高いですが、一緒に勉強していきましょう。

### 1.3 $\text{\LaTeX}$ を使うと何が嬉しいの？

1. 数式, 例文, 発音記号などの特殊な表記法が綺麗に出力できる
2. 参考文献の引用がとても楽
3. ユーザーによる拡張性が高い

#### 1.3.1 数式や例文が綺麗に出力できる

$\text{\TeX}$  ( $\text{\LaTeX}$  の基になっているシステム) は、元々パソコンで数式を綺麗に出力できるように開発されました。なので、数式、あるいは言語学の例文のような「ちょっと特別な書き方が必要」な分野に関してはめっぽう強いです。ちょっと例を見てみましょう。

$$s^2 = 1/n \sum_{i=1}^n (x_i - \mu)^2 \quad (1)$$

(1) Bzhedukh 語の例文 (?)

čʔaalya-m čʔəg<sup>w</sup>o-ər ya-ž<sup>w</sup>oa  
 boy-ERG filed-ABS 3SG-plows  
 ‘The boy plows field.’

どうでしょう。Word よりも綺麗に出力できているように見えませんか?

### 1.3.2 参考文献の引用がとても楽

文中で引用した全ての文献は、特定の形式に沿って参考文献として示さないといけません。「インデントするのが面倒くさい」「タイトル、巻号なんかを書く順番がわからない」などでお悩みではありませんか?  $\text{\LaTeX}$  ならそのお悩みは一発で解決できます。文中で文献を引用するだけで、自動的に参考文献一覧を作ってくれます。

### 1.3.3 ユーザによる拡張性が高い

Word を使っていると、痒い所に手が届かなかったり、あるいは逆にどうでもいいところでお節介をされて困ることはありませんか?  $\text{\LaTeX}$  なら、全ての設定が全て自分でできます。

さあ、みなさんも  $\text{\LaTeX}$  を始めて見ませんか?

## 2 $\text{\LaTeX}$ の基本的な使い方

### この章で説明すること

- エディタを立ち上げる
- エディタを使って簡単な文書を生成する
- 文字に装飾をする

### 2.1 エディタの使い方

文書の編集をしてくれるアプリケーションのことを「エディタ」と呼びます。エディタには、 $\text{\TeX}$ Works や  $\text{\TeX}$ Shop などがあります。まずは、エディタを立ち上げます。

## Windows ユーザ

「Windows メニュー/すべてのプログラム/T<sub>E</sub>XLive 2017」の中に T<sub>E</sub>XWorks Editor というアプリケーションがあるはずです。これを選択して立ち上げると、白紙の画面が出てきます。ここに命令を入力して文書を作ります。

## Mac ユーザ

アプリケーションフォルダに T<sub>E</sub>XShop というものが入っていると思います。これを選択して立ち上げると、白紙の画面が出てきます。ここに命令を入力して文書を作ります。

## 2.2 簡単な文書を作ってみよう

エディタを立ち上げたら、簡単な文書を作ってみましょう。

まずはじめに、L<sup>A</sup>T<sub>E</sub>X で必ず行わなければならないのが、(1) 文書クラスの指定と (2) ドキュメント環境の作成です。「新規作成」を行うと白紙の文書が作成できると思いますが、そこに

```
\documentclass[11pt]{jsarticle}
\begin{document}
aiueo
\end{document}
```

のように打ち込んでください。\\ という文字 (バックスラッシュ) は ¥ マークで代用しても構いません。打ち込んだら「タイプセット」をしてください。

### タイプセットの方法

#### T<sub>E</sub>XWorks ユーザ

左上に緑色の三角のボタンがあります。ボタンの右にあるプルダウンメニューに「pLaTeX(ptex2pdf)」と表示されていることを確認し、三角ボタンを押してください。Ctrl+T でもタイプセットができます。

#### T<sub>E</sub>XShop ユーザ

左上に「タイプセット」と書かれたボタンがあります。ボタンの右にあるプルダウンメニューに「LaTeX」と表示されていることを確認し、タイプセットボタンを押してください。command+T でもタイプセットができます。

タイプセットが成功すれば“aiueo”とだけ書かれた文書が出来るはずです。

ここで行っているのは、文書クラスの指定とドキュメント環境の作成です。まず、文書クラスの指定では「どんなスタイルの文書を作るか」を指定します。特に好みが無ければ、jsarticle というクラスを指定しておくのがよいでしょう。ドキュメント環境とは \begin{document} と \end{document} で

囲まれた部分のことです。ドキュメント環境を作るというのは「ここからここまでが作りたい文書です」という宣言をするということです。二つとも絶対に必要な手順なので、何も考えずに打ち込んでください。

## 2.3 文字の処理

### 2.3.1 改行

エディタ画面で以下のように入力をして改行されません。

入力:  
いろはにほへと  
ちりぬるを

出力:  
いろはにほへとちりぬるを

もし「いろはにほへと」と「ちりぬるを」の間に改行をしたいなら、\\という記号を「いろはにほへと」の後にはさみます。

入力:  
いろはにほへと\\  
ちりぬるを

出力:  
いろはにほへと  
ちりぬるを

### 2.3.2 段落

この\\を使った改行は、新しい段落を作しません。新しい段落を作りたい時、つまり「行頭の字下げ(インデント)を行いたい時」は、行と行の間に空白の行を挿入することで新しい段落が作れます。

入力:  
いろはにほへとちりぬるをわかよたれそつねならむ  
うゑのおくやまけふこえて

出力:  
いろはにほへとちりぬるをわかよたれそつ  
ねならむ  
うゑのおくやまけふこえて

段落を変えただけでなく段落の間に空白を一行はさみたいときは、\bigskipと入力します。

入力:  
いろはにほへとちりぬるをわかよたれそつねならむ  
\bigskip  
うゑのおくやまけふこえて

出力:  
いろはにほへとちりぬるをわかよたれそつ  
ねならむ  
うゑのおくやまけふこえて



一行空けたいけれどもインデントはしたくない、という場合は段落のはじめに`\noindent`と入力します。

入力:

いろはにほへとちりぬるをわかよたれそつねならむ  
`\bigskip`

`\noindent` うゐのおくやまけふこえて

出力:

いろはにほへとちりぬるをわかよたれ  
そつねならむ

うゐのおくやまけふこえて

ここで気をつけてほしいのは、`\noindent`と入力した後にそのまま文を続けず、必ず半角スペースを一つ挟むようにするということです。半角スペースをいれないで

`\noindent`うゐのおくやまけふこえて

のように入力すると、コンソール画面で“Undefined control sequence”というエラーが返ってきてタイプセットが失敗してしまいます。これは、`\`からはじまる文字列が「コマンド」として認識されるために起こるエラーです。上の例で言えば「`noindent`」という命令を読み込ませたいところで、「`noindent` うゐのおくやまけふこえて」という命令をしてしまっているということです。しかし、「`\noindent`うゐのおくやまけふこえて」というコマンドは定義されていませので、「そんな命令知らない!」ということでエラーが返ってくるわけです。半角スペースを入力することで、「ここで命令は終わりです」ということを表します。`\noindent` 以外のコマンドでも、コマンドを入力し終わった後に文字を続けずに半角スペースを入力するよう気をつけてください。

このように、コマンドを書くのに失敗するとエラーが返ってきてしまいます。でも心配せずにコンソールに表示されたエラーをきちんと読んでください。エラーメッセージを読めば、何がダメだったのかは大体分かります。よくありがちなものとしては、括弧を閉じていない、コマンドのスペルミスをしている、コマンドのあとにそのまま文字入力を続けてしまっている、などがあります。

### 2.3.3 空白

LaTeX では、空白の処理は少し複雑です。例えば、

入力: い  ろ は  に

出力: いろはに

このように半角のスペースをいくつ重ねても全て半角のスペース一つ分として処理されてしまいます。そこで、半角スペースを続けたい時には`~`という記号を使います。

入力: い~~~~ろ~は~~~~~に

出力: い ろ は に

このように、半角スペースを複数重ねることができます。なお、全角スペースの場合は重ねたら重ねた分だけ処理されますので、心配はいりません。

## 2.4 文字の装飾

ここまででは「地の文」をどう入力するかを説明しました。このセクションでは、「地の文」に装飾をつける方法を説明していきます。

今までの部分で、\から始まっている部分がありました。これを「コマンド」と呼び、コンピュータに与える命令を記述します。例えば、Hello World! という文字列をイタリックにしたいとしましょう。その時は以下のような記述をします。

入力: `\textit{Hello World!}`

出力: *Hello World!*

この入力の意味を解説します。`\textit`というのは「コマンド」で、ある操作をコンピュータに命令します。`\textit`というのは、「{}で囲った部分をイタリックにせよ」という命令です。同様のコマンドとして、文字列を太字にする`\textbf`などがあります。表2.4は、文字の装飾コマンドをまとめたものです。和文フォントは3つ、欧文フォントは7つの命令がありますので、これらを組み合わせて使ってください。

表1で示した字体以外は**基本的に使うことができません**。

L<sup>A</sup>T<sub>E</sub>X では、フォントサイズの指定もコマンドで行います。サイズ指定には、基本サイズからの縮小率を指定する「相対指定」と、変更後のサイズをそのまま指定する「絶対指定」を行います。

表1 文字に装飾を加えるコマンド

和文	<code>\textmc{}</code>	細明朝体
	<code>\textmc{\textbf{}}</code>	<b>太明朝体</b>
	<code>\textgt{}</code>	細ゴシック体
	<code>\textgt{\textbf{}}</code>	<b>太ゴシック体</b>
	<code>\textrm{}</code>	Roman(ウロコ付き)
	<code>\textbf{}</code>	<b>Boldface(ウロコ付き太字)</b>
	<code>\textit{}</code>	<i>Italic(イタリック)</i>
	<code>\textit{\textbf{}}</code>	<b><i>Bold Italic(イタリック太字)</i></b>
欧文	<code>\textsl{}</code>	<i>Slanted(斜字体)</i>
	<code>\textsl{\textbf{}}</code>	<b><i>Slanted(斜字体太字)</i></b>
	<code>\textsc{}</code>	SMALL CAPS(スモールキャピタル)
	<code>\textsc{\textbf{}}</code>	<b>BOLD SC(スモールキャピタル太字)</b>
	<code>\textsf{}</code>	Sans Serif(ウロコなし)
	<code>\textsf{\textbf{}}</code>	<b>Bold Sf(ウロコなし太字)</b>
	<code>\texttt{}</code>	Typewriter(計算機)
	<code>\texttt{\textbf{}}</code>	<b>Typewriter(計算機太字)</b>

表2 文字の大きさを指定するコマンド

入力		出力
相対	<code>{\tiny M}</code>	M
	<code>{\scriptsize M}</code>	M
	<code>{\footnotesize M}</code>	M
	<code>{\small M}</code>	M
	<code>{\normalsize M}</code>	M
	<code>{\large M}</code>	M
	<code>{\Large M}</code>	M
	<code>{\LARGE M}</code>	M
	<code>{\huge M}</code>	M
	<code>{\HUGE M}</code>	M
絶対	<code>{\fontsize{11pt}{11pt}\selectfont M}</code>	M
	<code>{\fontsize{4.235truemm}{4.235truemm}\selectfont M}</code>	M
	<code>{\fontsize{8truemm}{8truemm}\selectfont M}</code>	M

`\large`や`\small`というコマンドは、それぞれ「基本のフォントサイズ (11pt) から〇〇倍する」という相対指定コマンドです。`\fontsize`というのは絶対指定ですが、後の方に`\selectfont`というよくわからないコマンドがくっついています。これは文字送りを指定するコマンドですが、あまり気にしなくてよいです。

このように、色々なコマンドを使って文字の装飾を行っていきます。他にも上付き、下付き、下線、ダイアクリティカルマークの入力などの装飾も全てコマンドで行います。文字の装飾に限らず様々なコマンドの説明があるので、LaTeX コマンド集というサイトを是非参照してみてください

## 2.5 章立て

まとまった形式の文書を書くときには、章立てが必要になってきます。具体的な使い方を見ていきましょう。

`\section`というコマンドを入力することで、そこにあらたな節を作ります。

入力: `\section{ほげほげ}`

出力:

**3 ほげほげ**

`\section{}`という命令の解説をしましょう。これは、「`{}`で囲った部分をタイトルとする新しい節を作れ」という命令で、 $\text{\LaTeX}$  が自動的に節番号も振ってくれます。この文書では既に `section` が二つありますので、ここで新たに `section` を指定したことで自動的に「3」という番号を振ってくれたわけです。文字の大きさ、太さ、フォントや前後の空白も自動で設定してくれます。`subsection` 以下についても同様に処理してください。

## 3 スタイルファイル (拡張パッケージ) の導入

参照文献のセクションでも少し触れましたが、 $\text{\LaTeX}$  では機能を拡張するためにスタイルファイルというものを導入することができます。スタイルファイルの拡張子は `sty` です。スタイルファイルの中には、 $\text{\LaTeX}$  のインストールと同時にダウンロードされるものもあれば、自分でダウンロードしてこなければいけないものもあります。ここでは、スタイルファイルの導入の仕方を説明します。

### 3.1 あらかじめダウンロードされているものの導入

前のセクションで使用した `natbib.sty` は、大方の場合あらかじめパソコンの中に入っています。このようなスタイルファイルの場合、前のセクションで行ったように、`\documentclass{}`と`\begin{document}`の間に`\usepackage{(ファイル名)}`と指定することで導入できます。

## 3.2 まだパソコンに入っていないものの導入

CTANというサイトではスタイルファイルが多数配布されています。ここでは、`natbib.sty` というスタイルファイル (次の「参考文献の引用」セクションでもつかいます) を導入してみましょう。まずCTANにアクセスし、検索フォームから `natbib` と検索してください。検索結果のどこかに「Package natbib」というリンクがあるはずなので、そこをクリックしてください。`natbib` のページに移行したら、Download と書かれたリンクがあります。そこをクリックして、zip ファイルをダウンロードして解凍してください。

ここから `.sty` ファイルを生成する作業に入ります。まず、コマンドラインのアプリケーション (Win ならコマンドプロンプト、Mac ならターミナル) を立ち上げてください。コマンドラインから、解凍した `natbib` というフォルダの中に移動してください。例えば、`/Users/staff/Downloads/natbib` というパスに当該ファイルがあるなら、コマンドラインで `cd /Users/staff/Downloads/natbib` と移動してください。移動が完了したら、コマンドラインで `latex natbib.ins` と入力してください。そうすると、`natbib.sty` というファイルが生成されているはずです。この `natbib.sty` を編集中の `.tex` ファイルと同じディレクトリに置いてください。

次に、生成した `natbib.sty` を文書に導入します。文書内で、文書クラスの指定 (`\documentclass{}` という部分) と `\begin{document}` の間に、`\usepackage{natbib}` と記入してください。

これで新しいスタイルファイルの導入が完了しました。これ以降は文書内で `natbib` で拡張された機能を使うことができます。どのようにその機能を使うかは次節以降をご覧ください。

## 4 論文で使うあれこれ

さて、ここまでで基本的な文書の作り方を解説してきました。ここからは、基本的な論文で使う色々なものの挿入の仕方について説明していきたいと思います。

### 4.1 表の挿入

表の挿入の仕方を見ていきます。

#### 4.1.1 普通の表

以下のように入力すると、

```

\begin{table}[h]
\centering
\caption{変化表}
\label{amicus}
\begin{tabular}{|l|c|c|c|c|c|c|} \hline
& 主格 & 呼格 & 属格 & 与格 & 対格 & 奪格 \\ \hline
単数 & amicus & amice & amici & amico & amicum & amico \\ \hline
複数 & amici & amici & amicorum & amicis & amicos & amicis \\ \hline
\end{tabular}
\end{table}

```

表3 変化表

	主格	呼格	属格	与格	対格	奪格
単数	amicus	amice	amici	amico	amicum	amico
複数	amici	amici	amicorum	amicis	amicos	amicis

このように出力されます。コマンドの意味を確認して行きましょう。

- まず、全体を `table` 環境で囲みます。
- 一行目の `h` という部分は「この場所に出力せよ」という意味です。
- 二行目の `\centering` というのは、「表全体を中央揃えにせよ」という意味です。これを入力しないと左寄せになります。
- 三行目の `\caption` というのは、表の上に表示される名前です。
- 四行目の `\label` というのは、また後で説明しますが、この表の番号を参照する時に使います。この `label` はオプションなので、入力しなくても良いです。
- `tabular` 環境で囲まれている部分が表の本体になります。

```
{|l|c|c|c|c|c|}
```

というのは、縦の罫線と各セル内での文字の位置を設定している箇所です。`|`でどこに罫線を引くかを決定し、`l`と`c`と`r`で「列の数」と「各セル内での文字の位置」を決定します。`l`なら左寄せ、`c`なら中央揃え、`r`なら右寄せです。

```
{|lcccc|}
```

例えば、以上のように記述すると、5列で左端と右端にだけ縦罫線を引くという指定になります。

- `\\ \hline`というのは、「行の下側に横罫線を引き、ここでこの行は終われ」という意味です。`\\`を入力しないとコンパイルエラーになりますので、行の終わりには必ず入力してください。

- 6 行目から 8 行目からは各セルの内容です。セル同士は & で区切ってください。空白にするセルがあれば、何も入力しないまま & を入力してください。& の数と列数が合わない場合コンパイルエラーになります。

#### 4.1.2 セルの結合

セルを結合させたい場合は、横に結合する場合 `\multicolumn{}{}{}`、縦の場合 `\multirow{}{}{}` を使います。

```
\begin{table}[h]
\centering
\caption{横結合セル}
\label{amicus}
\begin{tabular}{|l|c|c|} \hline
\multicolumn{3}{|c|}{A} \\ \hline
\multicolumn{2}{|c|}{B} & C \\ \hline
D & E & F \\ \hline
\end{tabular}
\end{table}
```

表4 横結合セル

A		
B		C
D	E	F

`\multicolumn{}{}{}` の引数は、一番目から順に「横に結合するセルの数」、「セル内の文字の位置と罫線」、「セルの内容」です。

```
\begin{table}[h]
\centering
\caption{縦結合セル}
\label{multirow}
\begin{tabular}{|l|c|c|} \hline
\multirow{3}{*}{A} & \multirow{2}{*}{B} & C \\ \cline{3-3}
& & D \\ \cline{2-3}
& E & F \\ \hline
\end{tabular}
\end{table}
```

`\multicolumn{}{}{}` の引数は、順番に「縦に結合するセルの数」、「列の位置 (通常は \* であり)」、「セルの内容」です。結合されたことで入力の必要が無くなったセルは空白にしておいてください。`\cline{}` の括弧の中身は、「罫線を引き始めるセル-引き終わるセル」で指定してください。

表5 縦結合セル

A	B	C
		D
	E	F

以上が表の組み方です。

## 4.2 図の挿入

図を挿入するには `figure` 環境と `\includegraphics` を使います。挿入したい画像は、`.tex` が入っているフォルダと同じディレクトリに入れておくようにしてください。

```
\begin{figure}[h]
\caption{肝臓}
\centering
\includegraphics[width=50truemm]{liver}
\label{liver}
\end{figure}
```

図2 肝臓



`\caption{}` と `\label{}` は表の挿入の時と同じです。 `width=` では、挿入する画像の横幅がどれくらいの長さになるかを指定します。 `\includegraphics{}` の括弧の中には、挿入する画像のファイル名の拡張子を外した部分を書いてください。



### 4.3 相互参照

さて、ここまでで表や図を挿入する方法を見てきました。論文中では、これらの図に言及して、「表5を見ると…」などと書くことがあるでしょう。その時はいちいち表の番号を手打ちしてもよいですが、途中で別の表を挿入したりすると表番号が変わってしまい、いちいち修正するはめになります。それを防ぐために活躍するのが「相互参照」という機能です。

今まで作って来た図表では、`\label{}`というコマンドを書いていたと思いますが、これはこの相互参照の時に活躍します。例えば、表5の「縦結合セル」では`\label{multirow}`というラベルを振っています。“multirow”というのがこの表の「名前」になるわけです。この表の番号を参照したければ、`\ref{}`というコマンドを使って以下のように記述します。

入力: 表`\ref{multirow}`によると

出力: 表5によると

`\ref{}`で参照してくれるのは番号の数字だけなので、「表」とか「図」とかいった部分は自分で入力してください。

また、`\ref{}`で参照できるのは図や表だけではなくありません。節や頁など、あらゆるものを参照することができます。例えば、この文書ではセクション4.2に`\label{figins}`というラベルを振っています。これを利用して以下のように記述できます。

入力: `\ref{figins}`節を参照

出力: 4.2節を参照

### 4.4 脚注の挿入

LaTeX で文書に脚注を挿入するためには、`\footnote{}`というコマンドを使います<sup>1</sup>。脚注を付けたい直後の箇所でコマンドを使い、括弧の中に注の内容を書き込みます。

ダース・ヴェイダーはシス`\footnote{ダークサイドの`

ダース・ヴェイダーはシス<sup>a</sup>の暗黒卿である  
フォースを信奉する集団}の暗黒卿である  
<sup>a</sup> ダークサイドのフォースを信奉する集団

---

<sup>1</sup> ここで示した例は特殊な環境で使っているため番号表示がアルファベットになっていますが、通常に使った場合このように数字表記になります

## 5 参考文献の引用

### 5.1 BiBTeX の使い方

1. 書誌情報が入力されたファイルを作る
2. 入力した書誌情報を文中で引用する
3. 引用スタイルを指定する
4. 文中で書誌情報を入力したファイルを指定する
5. タイプセットする
6. 参考文献リストが作成される

LaTeX で参考文献を引用するには、ちょっとした手間が必要です。最初に「参考文献リストを作るのが Word に比べて楽」と言ったじゃないか、と思うかもしれませんが、正確にいうと「最初はちょっと手間がかかるが、一回設定してしまえばあとは楽」という意味です。上に書いたステップごとに進んで行きましょう。なお、まだ前セクションの「スタイルファイルを導入する」をご覧ください。なっていないかたは、そちらをご覧ください。なってからもう一度ここに戻ってきてください。

### 5.2 書誌情報を入力する

LaTeX で文献を引用するためには、まず書誌情報がつまったファイルを作成する必要があります。ファイルの作成には、エディタとは別のアプリケーションを使います。Mac なら BibDesk というアプリケーションが同時にインストールされているので、それを使ってください。他の OS なら、JabRef という Java アプリがどの OS でも使えるはずなので、それをインストールしてください。自分のお好みのもの (Mendeley, Zotero など) がある方はそれで結構です。

これらのアプリケーションを立ち上げ、新しいファイルを作成してください。BibDesk なら、File→New Bibliography を押してください。JabRef なら、左上の “New BibTeX database” というボタン (白ヌキの紙のようなアイコン) を押してください。

それでは、お手元にある適当な本の書誌情報を入力してみてください。BibDesk なら上の方にある緑色の + ボタンを押し、JabRef なら上の方にある白抜き色の + ボタンを押します。JabRef であれば、最初に「エントリタイプを入力する」画面が出ます。これは、これから入力するものがどんな種類の本なのかを決定する場面です。本の種類によって引用の仕方が変わります (例えば、収録されている論集の名前を入力する必要があるかどうか、など) ので、最適なものを選ぶようにしてください。後から変更もできますので、分からない時はとりあえず book か article を選択しておきましょう。BibDesk ではこの画面は表示されませんが、右上にデフォルトでは book と表示されている欄がありますので、そこで最適なものを選んでください。

書誌情報に入力する情報には、「required field」と「optional field」と「cite-key(BiBtex-key)」の3種類があります。required field とは、必ず入力しなければいけないもの(著者名、タイトルなど)です。optional field とは、入力しなくてもよいもの(発行月や出版地など)です。cite-key は半角英数ならなんでもよいのですが、自分が後で分かりやすくなるような名前を付けておきましょう。私はいつも「名字年」という形式にしています。書誌情報を入力する際に絶対に気をつけて欲しいのが以下の3点です。

1. 日本語文献の場合、必ず Yomi フィールドに著者の読みを**名姓の順**で記入すること。これを入力しないと、参考文献リストが名前順に表示されなくなりますので、気をつけてください。
2. 漢字の著者を入力する時には**姓名の順**で苗字と名前の間に半角スペースを入れることを注意してください。こうしないと苗字と名前が一つながりの名前として認識されます。
3. **cite-key(BiBtex-key)** を必ず入力すること。これを入力しないと、文書中で引用することができません。

書誌情報の入力が終わったら、編集中の.tex ファイルがあるのと同じディレクトリに.bib ファイルを保存してください。この時、.bib ファイルの名前はまた後で使いますので、わかりやすいものにしておいてください。

### 5.3 引用する

BiBTeX で書誌情報の入力が終わったら、そこで入力した cite-key を使って引用していきます。

表6 citation

<code>\citet{jakobson1963}</code>	?
<code>\citep{jakobson1963}</code>	(?)
<code>\citet[cf.] [12]{jakobson1963}</code>	?:12
<code>\citealp{jakobson1963}</code>	?

```
\citet{sakuma1941}, \citet{hattori1976},
\citet{postal1970}, \citet{paul1978},
\citet{aikhenvald2006}, \citet{kindaichi1955},
\citep{ueno1997}, \citep{kiparsky1968},
\citep{shibatani1978}, \citep{haegeman1994},
\citep{jakobson1963}, \citep{haegeman1994},
\citep[.cf] [112]{nansei2005}, \citet[] [12]{urabe2016}
```

?, ?, ?, ?, ?, (?), (?), (?), (?), (?), (?), (.cf?:112), ?:12

`\citet{}`は、発行年のみを括弧で囲うコマンドです。`\citep{}`は、著者と発行年を括弧で囲うコマンドです。コマンドの後の最初の引数は、“see”や“.cf”などを入れるためのオプションです。二つ目の引数はページ数を指定したい時のオプションです。

## 5.4 引用スタイルを指定する

文献を引用スタイルには、分野、雑誌などの違いによって多岐に渡ります。ここでは、言語学、社会学、経済学などの理系寄りの人文系で標準的なスタイルを入力する方法を紹介します。つまり、文中の引用は姓(発行年)、参照文献リストは姓名(発行年)「論文のタイトル」『収録されている書籍、雑誌のタイトル(あれば)』, 巻号, ページ, 出版社という形です。

### 5.4.1 natbib.sty を導入する

科学系の論文の引用スタイルを導入するために、`natbib.sty`を導入します。

文書クラスの指定(`\documentclass{}`という部分)と`\begin{document}`の間に、`\usepackage{natbib}`と記入してください。

### 5.4.2 jecon.bst を導入する

`natbib.sty`だけでは、日本語の文献に対応しません。そこで、武田史郎先生が作成した“`jecon.bst`”を導入します。

武田史郎先生のウェブサイト、「`jecon.bst`: 経済学用 BibTeX スタイルファイル」から、「ダウンロード」のページに行って、`jecon.bst`の最新版をダウンロードしてください。ダウンロードした zip ファイルを解凍すると、中に“`jecon.bst`”というファイルがあります。それを、編集時の `.tex` ファイルと同じディレクトリに置いてください。

ファイルの移動が終わったら、文書の`\begin{document}`の後に、`\bibliographystyle{jecon}`と記入してください。

## 5.5 書誌情報が入ったファイルを指定する

参照文献リストを挿入したい場所に`\bibliography{(自分が入力した.bibファイルの名前)}`と入力してください

## 5.6 タイプセットする

参照文献リストを表示するためには、ちょっとしたタイプセットの手間が必要です。ここまできたらあと少しなので、頑張ってください。

まず、タイプセットボタンの横のプルダウンメニューに「LaTeX」(TeXWorks の場合は pLaTeX(ptex2pdf)) と表示された状態で一回タイプセットを行ってください。その次に、プルダウンメニューから「BibTeX」(TeXWorks の場合は pBibTeX) と書かれたものを選択して、一回タイプセットボタンを押してください。その後、プルダウンメニューから LaTeX を再度選択し、あと二回タイプセットを行ってください。つまり、「LaTeX→(p)BibTeX→LaTeX→LaTeX」と計四回タイプセットを行ってください。これで、参考文献リストが表示されます。

サイトコマンドで引用した文献は全て自動的に「参考文献」に載せられます。記載は引用した順番ではなく、著者の姓のアルファベット順です。同じ著者の文献が複数引用されている場合、発行年が古い順に載せられます。

## 5.7 文献引用のまとめ

まず、ファイル構成が以下になっているか確認してください。

(適当なフォルダ)

- ├─ aiueo.tex (編集中のソースファイル)
- ├─ shoshi.bib (書誌情報を書いたファイル)
- └─ jecon.bst (日本語文献の引用スタイルを指定するためのファイル)

次に、文書の中身が以下になっているか確認してください。

```
\documentclass{jsarticle} %文書クラスの指定
\usepackage{natbib} %natbib.styの導入

\begin{document}

\bibliographystyle{jecon} %jecon.bstの導入
\cite{payne1997} %ここに入力するのは自分が入力したcite-keyです
\bibliography{shoshi} %書誌ファイルの指定

\end{document}
```

次に、「LaTeX→BibTeX→LaTeX→LaTeX」の順番でタイプセットを行ってください。

これで参考文献リストが作成できるはずです。それでもダメな場合、以下のことをためしてみてください。

1. 作業ファイル (フォルダの中にある、.tex, .bib, .bst 以外の全てのファイル。例えば、.aux, .bbl, .blg, .log, .out, .pdf, .synctex.gz など) を削除する
2. cite-key を正しく入力しているか確認する。特に、大文字小文字に気を付けてください。

## 6 言語学用の設定

### 6.1 例文の挿入

例文の挿入を補助するスタイルファイルはいくつかありますが、ここでは expex.sty を紹介します。expex.sty の詳しい使い方は、ExPex のドキュメントを参照してください。

(2) hoge

`\ex`と`\xe`で囲った中に、始まる例文を入れます。すると、例文番号が自動でつきます。

入力:

```
\ex  
arma virumque cano  
\xe
```

出力:

(3) arma virumque cano

丸括弧の下に a, b... と続ける場合、`\pex`と`\xe`の間に`\a`から例文を始めます。

入力:

```
\pex  
  \a Graecia capta ferum victorem cepit  
  \a tu fui ego eris  
\xe
```

出力:

(4) a. Graecia capta ferum victorem cepit  
b. tu fui ego eris

インターニアースグロスを表示する場合、`\ex`または`\pex`の`\xe`の間に、`\beginl`と`\endl`を書き、その中に`\glpreamble`, `\gla`, `\glb`, `\glc`, `\glft`という要素を入れます。`\glpreamble`は例文の上につける説明文（省略可）、`\gla`はグロスの1段目、`\glb`はグロスの2段目、`\glc`はグロスの3段目（省略可）、`\glft`は訳文です。各要素の最後には`//`をつけます。

入力:

```
\ex
\beginl
\glpreamble 糸満方言の例文//
\gla uree namaa neen//
\glb uri=ja nama=ja nee-n//
\glc それ=\Top{} 今=\Top{} ない-\Ind{}//
\glft 「それは今はない」//
\endl
\xe
```

出力:

(5) 糸満方言の例文

```
uree      namaa  neen
uri=ja    nama=ja nee-n
それ=TOP 今=TOP  ない-IND

「それは今はない」
```

## 6.2 樹形図の挿入

樹形図の挿入を行うにはいくつか方法があります。一つは、上山先生のサイトで配布されているTreeDrawerで樹形図を書いた後、スクリーンショットを取って図として挿入する方法です。これが余計なことをせず、ソースと格闘する必要もないかと思うのでオススメです。図の挿入は4.2を参照してください。TreeDrawerのインストールと使用法は上山先生のサイトを見てください。

他には、 $\text{\LaTeX}$ でスタイルファイルをインストールして直接樹形図を書く方法がいくつかあります。詳しくは? 「新版 $\text{\LaTeX}$ で言語学の論文を書くために」を見てください。このページでは他にも言語学の論文に有用な機能が紹介されています。

ここでは、“qtree”というスタイルファイルを用いた樹形図の書き方を紹介します。ソースと出

力は以下ようになります。

`\Tree`と入力した後に樹形図を書いてください。

[は node を一つ作ります。[の後にドットを打ってから文字を打つと、その文字がその node のラベルになります。

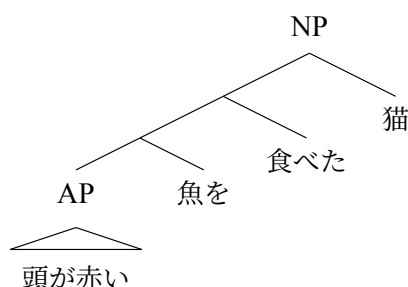
]で node を閉じます。]で node を閉じる時は、必ず]の前に半角スペースを入れてください。

`\qroof{}`で樹形図の省略を示す三角形が出力できます。

入力:

```
\Tree [.NP [[\qroof{頭が 赤い}.AP 魚を ]食べた ] 猫 ]
```

出力:



### 6.3 IPA の挿入

XeLaTeX では Unicode の文字をそのまま使えるので、IPA 気泡をエディタに貼り付ければそのまま使えます。貼り付けたりキーボードから入力するのが面倒な人は、`tipa.sty` というスタイルファイルを使えば、コマンドで IPA 記号を表示できます。やり方には色々あるのですが、`\textipa{}` というコマンドを使うのが一つの方法です。`\textipa{}` の中で表示させたい記号に対応するコマンドを入力します。例えば、`{}` の中に大文字の E を入力すると、 $\varepsilon$  が返ってきます。

入力: `\textipa{["Eksploʰneɪʃən]}`

出力:  $[\varepsilon ksplo^hneɪʃən]$

このように、「普通のアルファベットがそのまま IPA に使われているもの (例えば無声両唇破裂音など)」と、「特殊な記号を使う必要があるもの (無声後部歯茎摩擦音など)」と「音声記号を囲むブラケット」を一緒に入れてしまえば、全て整形して表示してくれます。

？“TIPA Manual”は、記号ごとにどのコマンドを使えばよいか詳しく解説しています。IPA の各記号の名称を英語で検索すると目当てのコマンドが探せます。たとえば、このページ内で“glottal stop”と検索すると、 $[ʔ]$  の記号がヒットします。 $[ʔ]$  では入力方法が 2 つ記載されていますが、Input1 の `\textglotstop` は `\textipa{}` の括弧の中に入れなくても単独で `ʔ` と入力できるコマン



ドで、Input2 のPというのは`\textipa{}`の括弧の中にいれないと使えないコマンドです。好きな方を選んで使ってください。

## 7 FAQ

Q: Word や Excel のファイルをそのまま貼り付けたりできる？

A: できません。

Q: コンパイルエラーが出た

A: エラーメッセージでウェブ検索してください。

Q:  $\text{\LaTeX}$  ってどうやって打ってるの？

A:  $\text{\LaTeX}$ です。

Q: このファイルみたいな装飾がしたい

A: ここに元ファイルが置いてあるので、当該箇所を見てみてください。

Q: その他のコマンドなどが知りたい

A: コマンドについては $\text{\LaTeX}$  コマンド集を、便利なスタイルファイルについては天地有情を、その他一般については $\text{\TeX}$  Wikiをごらんください。

## 8 Abbreviations

本文中でグロスの略語を使った場合、付録として略語の説明をアルファベット順に並べるのが通例です。leipzig.sty を使えば自動で集計し略号一覧を作ってくれますが、ここでは説明しません。leipzig.sty の導入と使い方をご覧ください。

ACC	対格 (accusative)
NOM	主格 (nominative)
PRS	現在 (present)