

報告書作成用スタイルファイル

渡辺徹@ FUNNIST 編集委員会

2004/11/25

`<*funpro>`

リファレンス

`\thisYear{<年度>}`

‘2004’等の形式で年度を指定します。非ユーザーコマンドです。

`\midorfin{<中間>}{<最終>}`

`\midorfin`はその報告書が中間か最終かによっていずれかの引数を出力するようにします。もしもその報告書が中間ならば1つ目の引数だけが出力されるようになりますから、中間と最終で内容を変えたいときに便利です。

`\midorfin{中間}{最終}`報告書ではほげほげ。

`\type{<文字列>}`

コンソールからの入力を促すときに使われる命令です。いかなる特殊記号を書いても大丈夫です。`\type{perl -ni 'hoge'}`とすると `perl -ni 'hoge'` となります。

`\begin{TYPE} \type[<記号>]{<文字列>} \end{TYPE}`

複数行に渡るコンソール入力を促すときに使われます。このときlist環境が使われ、行の先頭に記号がつきます。標準でドル‘\$’がつきますが、[<記号>]に各々の記号を適宜バックスラッシュと共に用いると指定できます。‘%’にしたければ`\item[%]{ls -l /}`とします。

`\key{<キー>}`, `\return`, `upkey`, `downkey`, `leftkey`, `rightkey`

キーボードの特定のキーを示すときに使われます。

`\key{Delete}`, `\return`, `\upkey`, `downkey`, `leftkey`, `\key{Alt}` は次のようになります。

`Delete`, `↵`, `↑`, `↓`, `←`, `Alt`

`\dir{<ファイル>}`

ファイルやディレクトリを示すときに使われます。

`\dir{/usr/local/bin/emacs}`は `/usr/local/bin/emacs` です。

`\suf{<拡張子>}`

拡張子を示すときに使います。自動的にピリオドがつきます。`\suf{pl}`のように使います。

拡張子.plはPerlとPrologの両方で使われている。

`\begin{INPUT} {<文字列>} \end{INPUT}`

テキストファイルなどへの入力を示します。以下の先頭のパーセントは無視してください。

```
% \begin{INPUT}
% \documentclass{jsarticle}
% \begin{document}
% Hello, \LaTeX!!
% \end{document}
% \end{INPUT}
% と入力してください。
```

`\begin{CON} {<文字列>} \end{CON}`

コンソールに出力される結果を示すときに使われます。

`\ProjectLeader{<学籍番号>}{<和文氏名>}{<欧文氏名>}`

`\GroupLeader{<学籍番号>}{<和文氏名>}{<欧文氏名>}`

表紙用のプロジェクトリーダーとグループリーダーを書きます。表紙を出力した後は消されます。引数の順番は決まっています。

```
\ProjectLeader{1202000}{未来太郎}{Taro~Mirai}
```

`\GroupMember{<番号>}{<学籍番号>}{<和文氏名>}{<欧文氏名>}`

グループメンバーを複数書きます。〈番号〉は必ず 1 から始めて連番で指定します。

```
\GroupMember{1}{1202100}{函館太郎}{Taro~Hakodate}
```

`\jProjectName{<和文プロジェクト名>}`

`\eProjectName[<短いヘッダー用のプロジェクト名>]{<欧文プロジェクト名>}`

和文と欧文のプロジェクト名を書きます。欧文のプロジェクト名がヘッダーを 2 行にまたがってしまうほど長い場合は、任意引数として短い欧文のプロジェクト名を記述します。

`\jGroupName{<和文グループ名>}`

`\eGroupName{<欧文グループ名>}`

和文と欧文のグループ名を書きます。

`\ProjectNumber{<プロジェクト番号>}`

プロジェクト番号を書きます [2004/07/27]。‘<プロジェクト番号>-<グループ番号>’ としなければなりませんので、4 番の A グループならば ‘4-A’ となります。

`\jadvisor{<和文指導教員 1, 教員 2,...>}`

`\eadvisor{<欧文指導教員 1, 教員 2,...>}`

指導教員を書きます。複数指定することが可能です。そのとき必ず半角のカンマで区切ります。それ以外の記号で区切ると正しく出力されません。

`\jadvisor{未来太郎, 北海花子, 函館太郎}`

`\jdate{〈和文提出日〉}`

`\edate{〈欧文提出日〉}`

提出日を和文と欧文で指定します。

`\SumOfMembers{〈メンバー数〉}`

メンバー数を指定します。これが与えられないとメンバーを表紙に出力できません。正の整数を与えないと正常に出力されません。

`\begin{eabstract} {〈欧文概要〉} \end{eabstract}`

欧文の概要を出力します。この環境中に後述の`\ekeyword` 命令を使うことができます。

`\begin{jabstract} {〈和文概要〉} \end{jabstract}`

和文用の概要を書きます。

`\ekeyword{〈欧文キーワード 1, キーワード 2,...〉}`

`\jkeyword{〈和文キーワード 1, キーワード 2,...〉}`

複数のキーワードを指定するときの区切りは適切なものを使ってください。そのうちこのマクロは実装を変更するかもしれません（和文の場合は‘、’で欧文の場合は‘;’になるように）。

`\bunseki{〈文責〉}`

文責を示すための命令です。`\bunseki{未来太郎}`とするだけです。

`\qu, \qq, yo, \yy, \pp, \wasyo, \yousyo`

おまけの命令です。

`\qu{hoge}` is `\qq{hoge & hoge}`. Hoge wrote `\yousyo{hoge is good.}`

`\yo{あれ}`と`\yy{それはあれです。}`ならば`\pp{あれ?}``\wasyo{ほげ学}`を
と入力すると次のようになります。

‘hoge’ is “hoge & hoge”. Hoge wrote *hoge is good*. 「あれ」と『それはあれです。』ならば（あれ？）『ほげ学』を書いた。

`\chapl原因, \chappref, \seclab, \secref, \applab, \appref, \figlab, \figref, \tablab, \tabref, \equ原因, \equref, \pref`

相互参照用の命令です。特に意識して使うこともありません。詳しくは渡辺徹が書いた『好き好き L^AT_EX 2_ε 初級編』の 6.5 節あたりを参照してください。

`\begin{mytab} [〈位置指定子〉] {〈見出し〉} {〈ラベル〉} \end{mytab}`

`\myimage [〈オプション〉] {〈ファイル名〉} {〈見出し〉} {〈ラベル〉}`

こちらの命令も『好き好き』を参照してください。

ソースの解説

`\RequirePackage` `okumacro` からは`\keytop` 命令を使っています。 `url` からは`\url` 命令などを使っています。

```
\RequirePackage[obeyspaces,spaces]{url}
\RequirePackage{okumacro}
\RequirePackage{verbatim}
```

`\RequirePackage{txfonts}` はやめました [2004/07/20]。

`\if@mid@` 中間報告書と最終報告書を区別するためのブール値 `@mid@` 定義します。 `funpro` の任意引数 (パッケージオプション) に `middle` を渡すと「中間」、 `final` を渡すと「最終」の報告書用の体裁になります。

```
\newif\if@mid@
\DeclareOption{middle}{\@mid@true}
\DeclareOption{final}{\@mid@false}
\ProcessOptions\relax
```

`\mid@fin` 中間・最終のどちらを執筆しているかによって出力形式決めます。 ユーザレベルでは

`\midorfin` `\midorfin` 命令を使います。 1 つ目の引数が中間用、 2 つ目の引数が最終用になります。

```
\def\mid@fin#1#2{\if@mid@ #1\else#2\fi}
\let\midorfin\mid@fin
```

`DeclareFontShape` 不足すると思われるフォント属性を定義します。

```
\DeclareFontShape{JY1}{gt}{m}{sc}{<->ssub*gt/m/n}{ }
\DeclareFontShape{JT1}{gt}{m}{sc}{<->ssub*gt/m/n}{ }
\DeclareFontShape{JY1}{gt}{bx}{it}{<->ssub*gt/m/n}{ }
\DeclareFontShape{JT1}{gt}{bx}{it}{<->ssub*gt/m/n}{ }
```

ページレイアウト ページレイアウトは決め打ちです。 ユーザによる変更は好ましくありません。

```
\newdimen\fullwidth
\headheight=2\topskip
\headsep=1zw
\topmargin=-\headheight
\advance\topmargin by -\headsep
\advance\topmargin by -2zw
\@setfontsize\normalsize{10pt}{15pt}
\parindent=1zw
\setlength\textwidth{44zw}
\setlength\fullwidth{\textwidth}
\setlength\textheight{45\baselineskip}
\evensidemargin=-1zw
\oddsidemargin=0pt
\marginparpush=0pt
\marginparwidth=0pt
\marginparsep=0pt
\setlength\footskip{20\p@}
```

`\thisYear` ‘2004’ などの形式で年度を指定します。

```
\def\thisYear#1{\gdef\@Year{#1}}
```

`\type` 文中でのコンソールの入力を示すときには`\type` 命令を使って`\type{which perl &}` とすると which perl & となります。

```

\newif\if@TYPE
\def\type{\@ifnextchar[{\@@type}{\@type}}
\def\@hoge{\begingroup \urlstyle{tt}\Url}
\def\@@type[#1]{\if@TYPE\item[{\ttfamily#1}]\fi\begingroup \urlstyle{tt}\Url}
\def\@type#1{\if@TYPE \item\@hoge{#1}%
\else\underline{\@hoge{#1}}\fi}

```

TYPE 別行に長いコンソール入力を示すときはTYPE 環境で囲んであげます。1 行に対して必ず \type 命令を使い次のように入力します。TYPE 環境中の \type 命令の先頭の記号は任意引数に指定できます。何も表示したくないときは \type[] と記述します。

```

% \begin{TYPE}
% \type{echo "hoge" && sleep 1 && su}
% \type{cat /etc/passwd}
% \type[\%]{cat /etc/passwd}
% \type[] {cat hoge}
% \end{TYPE}

```










出力は次のようになります。段落の左側だけが字下げされます。

```

$ echo "hoge" && sleep 1 && su
$ cat /etc/passwd
% cat /etc/passwd
cat hoge

\def\TYPE{\@TYPEtrue%
\list{\mbox{\texttt{$}}}{\rightmargin\z@}}
\def\endTYPE{\endlist \@TYPEfalse}

```

key キーボードの特定のキートップを示すときには \key 命令を使います。複数のキーを同時に押すことを示すには key 命令の引数に複数のキートップを半角カンマ (,) で区切って入力します。1 つだけのときは \key{Ctrl} とすると  となります。2 つ以上の時は \key{Ctrl,Alt,F1} とすると  となります。  の後に続けて他のキーを入力することを示すためには \key{Ctrl,C}\key{b} とすると  となります。リターンキーは \return とすると  という出力が得られます。他にも \upkey、\downkey、\leftkey、\rightkey の方向キーが用意されており、 、 、 、  という出力になります。

```

\def\key{\thinspace\@key}%
\def\@key#1{\@tempcnta=\z@%
\for\member:=#1\do{%
\ifnum\@tempcnta<1%
\keytop{\member}%
\else%
\texttt{+}\keytop{\member}%
\fi%
\advance\@tempcnta\@ne}%
\thinspace}%

```

\dir ファイル名やディレクトリを示すときは \dir を使います。適宜スラッシュやドットの前で改行すべき時は改行を許します。 \dir{/usr/local/bin/emacs} とすると /usr/local/bin/emacs となります。

```

\def\dir{\begingroup \urlstyle{tt}\Url}

```

<code>\suf</code>	<p>拡張子を示す場合は<code>\suf</code> 命令を使います．引数にピリオドは省略します．<code>\suf{pl}</code>とすると‘.pl’となります。</p> <pre>\newcommand{\suf}[1]{\texttt{.#1}}</pre>
<code>INPUT</code>	<p>ファイルへの入力を示すときは<code>INPUT</code> 環境を使います。</p> <pre>\newenvironment{INPUT}{ \list{}{\leftmargin=2zw \rightmargin=0zw} \item\small\verbatimim}{\endverbatimim \endlist}%</pre>
<code>CON</code>	<p>コンソールからの出力を示すときは<code>CON</code> 環境を使います。</p> <pre>\newenvironment{CON}{% \list{}{\leftmargin=2zw \rightmargin=0zw} \item\small\verbatimim}{\endverbatimim \endlist}%</pre>
<code>\ProjectLeader</code>	<p>表紙用の情報 各種表紙用の情報を入力するためのユーザレベルのコマンドとマクロ側のコマンドを定義します。プロジェクトリーダは<code>\ProjectLeader</code> に書きます。基本的にどのメンバーも引数には「学籍番号」、「和文氏名」、「欧文氏名」を書きます。それ以外の引数の渡し方では表示がおかしくなります。</p> <pre>\def\ProjectLeader#1#2#3{% \gdef\@ProjectLeader{#1% \hskip1.5zw #2\hskip1.5zw #3}}</pre>
<code>\GroupLeader</code>	<p>グループリーダです。</p> <pre>\def\GroupLeader#1#2#3{% \gdef\@GroupLeader{#1% \hskip1.5zw #2\hskip1.5zw #3}}</pre>
<code>\GroupMember</code>	<p>グループメンバーです。グループメンバはメンバーを識別するために 1 つ目の引数に 1 からはじまる番号を書きます。2 つ目の引数から「学籍番号」、「和文氏名」、「欧文氏名」を書きます。それ以外の引数の渡し方では表示がおかしくなります。</p> <pre>\def\GroupMember#1#2#3#4{% \expandafter\gdef\csname @GroupMember#1\endcsname{% #2\hskip1.5zw \hbox{#3\hfil}\hskip1.5zw #4}}</pre>
<code>\jProjectName</code>	<p>プロジェクト名です。基本的に先頭に <code>j</code> がついている場合は和文用、<code>e</code> がついている場合は欧文用のコマンドになります。欧文用には任意引数として、ヘッダー用の短いタイトルを書くことができます。</p>
<code>\eProjectName</code>	<pre>\def\jProjectName#1{\gdef\@jProjectName{#1}} \def\eProjectName{\@ifnextchar[{\@@eProjectName}{\@@@eProjectName}} \def\@@eProjectName[#1]#2{% \gdef\short@eProjectName{#1}% \gdef\@eProjectName{#2}% } \def\@@@eProjectName#1{\gdef\@eProjectName{#1}}</pre>
<code>\jGroupName</code>	<p>グループ名も同様です。</p>
<code>\eGroupName</code>	<pre>\def\jGroupName#1{\gdef\@jGroupName{#1}} \def\eGroupName#1{\gdef\@eGroupName{#1}}</pre>
<code>\ProjectNumber</code>	<p>プロジェクト番号を示します [2004/07/27]。</p> <pre>\def\ProjectNumber#1{\gdef\@ProjectNumber{#1}}</pre>
<code>\jadvisor</code>	<p>指導教員も同様です。ただし、複数の指導教員がいる時は半角カンマ (,) で区切りま</p>
<code>\eadvisor</code>	<p>す。カンマの両側に余計な空白を挿入する必要はありません。実際に指導教員を表紙に出力する命令は<code>\show@jadvisor</code> と<code>\show@eadvisor</code> が行います。</p>

```

\def\jadvisor#1{\gdef\@jadvisor{#1}}
\def\eadvisor#1{\gdef\@eadvisor{#1}}
\def\show@jadvisor{%
  \@for\member:=\@jadvisor\do{\member\m@hskip}}
\def\show@eadvisor{%
  \@for\member:=\@eadvisor\do{\member\m@hskip}}

\jdate      提出日も同様です。
\edate      \gdef\jdate#1{\gdef\@jdate{#1}}
            \gdef\edate#1{\gdef\@edate{#1}}

\SumOfMembers      メンバー数を定義します。マクロの都合上、人数を指定しないと正しく、グループメン
                    バーを出力できません。リスト構造を使ってメンバー情報を保存できますが、マクロが大
                    掛かりになるのでこれで勘弁してください。

                    \newcount\@members
                    \def\SumOfMembers#1{\@members=#1}
                    \def\@show@members{%
                      \@tempcntb=\@members
                      \advance\@tempcntb\@ne
                      \@tempcnta=\z@
                      \@whilenum \@tempcnta<\@tempcntb
                        \do{\edef\h@ge{\@GroupMember\the\@tempcnta}%
                          \hbox{\@nameuse{\h@ge}}\relax\advance\@tempcnta\@ne}}%

\reset@title      レジスタの開放をします。表紙を出力した後は、その表紙の情報は使われ無いと思われ
                    ますので、開放します。 \maketitle 命令の後でも特定の情報を使いたいときは適当に

                    \gdef\jProjectName{ほげを考える}

などとすると\jProjectName に「ほげを考える」という用語などが保存されます。

\def\reset@title{%
  \global\let\thanks\relax
  \global\let\@thanks\@empty
  \global\let\title\relax
  \global\let\@title\@empty
  \global\let\author\relax
  \global\let\@author\@empty
  \global\let\date\relax
  \global\let\@date\@empty
  \global\let\and\relax
  % プロジェクト名
  \global\let\jProjectName\relax
  \global\let\@jProjectName\@empty
  %\global\let\@eProjectName\relax
  %\global\let\@eProjectName\@empty
  % 短い欧文のプロジェクト名はヘッダーに使用するので開放しない
  % \ud{2004/11/25}
  %\global\let\short@eProjectName\@empty
  % グループ名
  \global\let\jGroupName\relax
  \global\let\@jGroupName\@empty
  \global\let\@eGroupName\relax
  \global\let\@eGroupName\@empty
  % プロジェクト番号\ud{2004/07/27}
  %\global\let\ProjectNumber\relax
  % こちらもヘッダーに使用する\ud{2004/11/25}

```

```

% \global\let\@ProjectNumber\@empty
% 指導教員
\global\let\jadvisor\relax
\global\let\@jadvisor\@empty
\global\let\eadvisor\relax
\global\let\@eadvisor\@empty
\global\let\show@jadvisor\@empty
\global\let\show@eadvisor\@empty
% 提出日
\global\let\jdate\relax
\global\let\@jdate\@empty
\global\let\edate\relax
\global\let\@edate\@empty
% リーダ
\global\let\ProjectLeader\relax
\global\let\@ProjectLeader\@empty
\global\let\GroupLeader\relax
\global\let\@GroupLeader\@empty
% グループメンバー
\global\let\GroupMember\relax
\@tempcntb=\@members
\advance\@tempcntb\@ne
\@tempcnta=\z@
\@whilenum \@tempcnta<\@tempcntb
  \do{\global\expandafter\let
    \csname @GroupMember\the\@tempcnta\endcsname \@empty
    \relax\advance\@tempcnta\@ne}}%

\s@hskip 水平方向の空白を定義します。スタイルファイルの中でいろいろ使われます。
\m@hskip \def\s@hskip{\hskip.5zw}
\l@hskip \def\m@hskip{\hskip1zw}
          \def\l@hskip{\hskip2zw}

\s@vskip 水平方向の空白を定義します。スタイルファイルの中でいろいろ使われます。
\m@vskip \def\s@vskip{\vskip.5zw}
\l@vskip \def\m@vskip{\vskip1zw}
          \def\l@vskip{\vskip2zw}

\maketitle 表紙を定義します。独断と偏見で体裁を決めていますから、若干日本人好みではない部
分がありますが勘弁してください。
          2004 年 7 月 2 日にレジスタの開放に関するバグを修正しました。 \@maketitle と
\reset@title を開放します。これは\maketitle から呼び出されます。reset@title
は相変わらず@maketitle から呼び出されます。
\def\maketitle{%
  \@maketitle
  \let\@maketitle\@empty
  \let\reset@title\@empty
  \if@openright
    \cleardoublepage
  \else
    \clearpage
  \fi
}
\def\@maketitle{%
  \begin{group}

```



```

\begin{titlepage}%
\let\footnotesize\small
\let\footnoterule\relax
\let\footnote\relax
\begin{center}%
\thispagestyle{empty}%
{\Large\headfont 公立はこだて未来大学~\@Year~年度\space
システム情報科学実習\mid@fin{I}{II}\par
% \mid@fin{中間}{最終}報告書}\par\s@vskip
% なぜかは知らないけれど、いつの間にか更新されていた
グループ報告書}\par\s@vskip
{\headfont Future~University-Hakodate\space\@Year\space
System\space Information\space Science Practice\space\mid@fin{I}{II}%
% \par\mid@fin{Midterm}{Final}~Report}\par\m@vskip
\space Group~Report}\par\s@vskip
{\large\headfont プロジェクト名}\par\@jProjectName\par
{\large\headfont Project Name}\par\@eProjectName\par\m@vskip
{\large\headfont グループ名}\par\@jGroupName\par
{\large\headfont Group Name}\par\@eGroupName\par\m@vskip
{\headfont プロジェクト番号/Project~No.}\par\@ProjectNumber\par\m@vskip
{\large\headfont プロジェクトリーダー/Project~Leader}\par
\@ProjectLeader\hfill\par\m@vskip
{\large\headfont グループリーダー/Group~Leader}\par
\@GroupLeader\hfill\par\m@vskip
{\large\headfont グループメンバー/Group~Member}\par
\mbox{\vbox{\@show@members}}\par\m@vskip
{\large\headfont 指導教員}\par\show@jadvisor\par
{\large\headfont Advisor}\par\show@eadvisor\par\m@vskip
{\large\headfont 提出日}\par\@jdate\par
{\large\headfont Date of Submission}\par\@edate
\end{center}%
\vfil\null%
\end{titlepage}%
\endgroup
\setcounter{footnote}{0}%
\reset@title% ここでレジスタの開放
}%

```

tocdepth 目次の深さは小節レベルまで出力します。

```
\setcounter{tocdepth}{2}
```

\@makechapterhead 章見出しについては番号の段階で改行しないように変更しました。

```

\def\@makechapterhead#1{%
\space*{.5\Cvs}
{\parindent \z@ \raggedright \normalfont
\ifnum \c@secnumdepth >\m@ne
\ifmainmatter
\huge\headfont \@chapapp\thechapter\@chappos
\hspace{1zw}
\fi
\fi
\interlinepenalty\@M
\huge\headfont #1\par\nobreak
\space{1.5\Cvs}}
\def\@makeschapterhead#1{%
\space*{.5\Cvs}

```

```

{\parindent \z@ \raggedright
 \normalfont
 \interlinepenalty\@M
 \huge \headfont #1\par\nobreak
 \vskip 1.5\Cvs}}

```

\headfont 見出し用のフォントを決定するコマンドです。ここでは和文はゴシック体、欧文は Times Bold になります。

```

\renewcommand{\headfont}{\reset@font\gtfamily\bfseries}

```

eabstract 欧文の概要を書くための環境です。フォントサイズは 1 段階小さくしていますし、行間も狭くなります。最後に改ページします。

```

\newenvironment{eabstract}{%
 \centerline{\Large\headfont Abstract}\par%
 \begin{list}{}{%
 \leftmargin=3zw
 \rightmargin\leftmargin
 \listparindent0pt}\item[]\relax
 \small\narrowbaselines}%
 {\end{list}\par\clearpage}%

```

jabstract 和文の概要を書くための環境です。フォントサイズは 1 段階小さくなりますが、行間は和文用の行間のままです。結びに改ページします。

```

\newenvironment{jabstract}{%
 \centerline{\Large\headfont 概要}\par%
 \begin{list}{}{%
 \leftmargin=3zw
 \rightmargin\leftmargin
 \listparindent0pt}\item[]\relax
 \small\widebaselines}%
 {\end{list}\par\clearpage}

```

ekeyword 概要においてキーワードを出力したいときは \jkeyword 命令や \ekeyword 命令を使い

jkeyword ます。それぞれ和文、欧文に対応します。

```

\newenvironment{ekeyword}{%
 \m@vskip{\headfont\small Keyword\quad}\small}{}
 \newenvironment{jkeyword}{%
 \m@vskip{\headfont\small キーワード\quad}\small}{}

```

\paragraph 段落、小段落用の環境を再定義します。

\subparagraph

```

\if@twocolumn
 \renewcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
 {\z@}{-1zw}% 改行せず 1zw のアキ
 {\normalfont\normalsize\headfont}}
 \else
 \renewcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
 {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
 {-1zw}% 改行せず 1zw のアキ
 {\normalfont\normalsize\headfont}}
 \fi

```

\pagestyle ページスタイルは柱には何も出力せず、ノンブルにページ番号を出力するシンプルな物を採用します。しかし、後期になってなんだかいろいろ変更されたようです。

```

\def\ps@plainfront{%
 \let\@mkboth\@gobbletwo

```

```

\let\@oddhead\@empty
\let\@evenhead\@oddhead
\def\@oddfoot{%
    \normalfont\hfil\lower-.2ex\hbox{-}\space\thepage
    \space\lower-.2ex\hbox{-}\hfil}%
\let\@evenfoot\@oddfoot
}
%
\global\let\ps@plainhead\@empty
\global\let\ps@plainfoot\@empty
\global\let\ps@plain\@empty
\global\let\ps@myheadings\@empty
\global\let\ps@headings\@empty
%
\def\plainifnotempty{%
    \if@mainmatter
        \thispagestyle{plainmain}%
    \else
        \thispagestyle{plainfront}%
    \fi
}
%
\def\ps@plainmain{%
    \let\@mkboth\@gobbletwo
    \def\@oddhead{\ifx\short@eProjectName\@undefined
        \@eProjectName\else\short@eProjectName\fi\hfill}%
    \let\@evenhead\@oddhead
    \def\@oddfoot{\normalfont
        Group~Report~of~\@Year~SISP\space\hfil
        \lower-.2ex\hbox{-}\space\thepage\space\lower-.2ex\hbox{-}\hfil
        Group~Number\space\@ProjectNumber}%
    \let\@evenfoot\@oddfoot
}
%
\def\mainmatter{%
    \if@openright
        \cleardoublepage
    \else
        \clearpage
    \fi
    \@mainmattertrue
    \pagestyle{plainmain}%
    \pagenumbering{arabic}%
}
%
\pagestyle{plainfront}

```

\bunseki 節や小節の末尾に文責を示すための命令です。

```

\def\bunseki#1{%
    \begin{flushright}%
        \mbox{( 文責:~#1 )}%
    \end{flushright}}%

```

\qu 欧文の引用には\qu と\qq を使います．単語の引用は\qu で，文の引用には\qq を使
\qq うようにします．和文の引用には\yo と\yy を使います．これも同じように単語には\yo
\yo
\yy

`\pp` で、文には`\yy` です。丸括弧で括る場合は`\pp` 命令を使います。全角丸括弧が使われます。雑誌名や書籍名を参照するときは`\wasyo` と`\yousyo` 命令を使います。和書の場合は`\yousyo` `\wasyo`、洋書の場合は`\yousyo` です。わかりやすいでしょう？

```
\newcommand{\qu}[1]{‘#1’}
\newcommand{\qq}[1]{‘‘#1’’}
\newcommand{\yo}[1]{「#1」}
\newcommand{\yy}[1]{『#1』}
\newcommand{\pp}[1]{(#1)}
\newcommand{\wasyo}[1]{『#1』}
\newcommand{\yousyo}[1]{\emph{#1}}
```

`\chaplab` 章にラベルをつける場合は`\chaplab` 命令を使います。接頭語`chap:` が自動的に付加されますので`\chaplab{chap:hoge}` とはせずに、`\chaplab{hoge}`としてください。これを参照するときは`\chapref{hoge}`としてください。

```
\newcommand{\chaplab}[1]{\label{chap:#1}}
\newcommand{\chapref}[1]{第~\ref{chap:#1}~章}
```

`\seclab` 節も同様に`\seclab` と`\secref` の 2 つの命令を使います。

```
\secref \newcommand{\seclab}[1]{\label{sec:#1}}
\newcommand{\secref}[1]{\ref{sec:#1}~節}
```

`\applab` 付録の特定の章を参照するときは`\applab` と`\appref` 命令を使います。付録中の節を参照するには前述の`\seclab` と`\secref` を使います。

```
\newcommand{\applab}[1]{\label{app:#1}}
\newcommand{\appref}[1]{付録~\ref{app:#1}}
```

`\figlab` 図にラベルを貼るときは`\figlab`、参照するときには`\figref` を使います。

```
\figref \newcommand{\figlab}[1]{\label{fig:#1}}
\newcommand{\figref}[1]{図~\ref{fig:#1}}
```

`\tablab` 表も同様に`\tablab` と`\tabref` を使います。

```
\tabref \newcommand{\tablab}[1]{\label{tab:#1}}
\newcommand{\tabref}[1]{表~\ref{tab:#1}}
```

`\equlab` 式については参照方法が色々あるのですが、次のような体裁にします。

```
\equref \newcommand{\equlab}[1]{\label{equ:#1}}
\newcommand{\equref}[1]{式~(\ref{equ:#1})}
```

`\pref` ページ番号を参照するには`\pref` 命令を使います。通常、節や章を参照するときは、その章節の番号のみを示します。ページ番号を示す必要はありません。

```
\newcommand{\pref}[1]{\pageref{#1}~ページ}
```

`mytab` 表は通し番号をつけ中央揃えにし、表の上部に表見出しをつけます。そのため任意の表 (`tabular` 環境などで作成した) を`mytab` 環境に入れ子にします。

```
\newenvironment{mytab}[3][htbp]{%
\begin{table}[#1]\begin{center}\caption{#2}\tablab{#3}%
{\end{center}\end{table}}%
```

`\myimage` EPS などの画像に関しては`\myimage` 命令を使います。1 つ目の引数に画像を張り込むときのオプション、2 つ目にファイル名、3 つ目にキャプション、4 つ目にラベルをつけます。ラベルには自動的に接頭語`fig:` が付加されますので、`{fig:hoge}` のような指定にはしません。

```
\newcommand{\myimage}[4][width=.8\textwidth]{%
\begin{figure}[htbp]%
\centering\includegraphics[#1]{#2}%
}
```

```
\caption{#3}\figlab{#4}%  
\end{figure}}%
```

hissu ユーザは使わない命令です。

```
\def\hissu{\par\noindent\color{red}/*\space}  
\def\endhissu{\space*/\par\color{black}}  
  
\</funpro>  
\endinput
```