

L^AT_EX for linguists

Sylvia Blaho



University of Tromsø

Autumn 2006

Class 5: Fine-tuning text & tables

- Fine-tuning text

Class 5: Fine-tuning text & tables

- Fine-tuning text
 - Word and character spacing

Class 5: Fine-tuning text & tables

- Fine-tuning text
 - Word and character spacing
 - Vertical spacing

Class 5: Fine-tuning text & tables

- Fine-tuning text
 - Word and character spacing
 - Vertical spacing
 - Line breaking

Class 5: Fine-tuning text & tables

- Fine-tuning text
 - Word and character spacing
 - Vertical spacing
 - Line breaking
 - Page breaking

Class 5: Fine-tuning text & tables

- Fine-tuning text
 - Word and character spacing
 - Vertical spacing
 - Line breaking
 - Page breaking
- Tables

1. Fine-tuning text

1. Word and character spacing

- (1) $\text{\TeX}$ inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.

1. Fine-tuning text

1. Word and character spacing

- (1) T_EX inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.
- (2) T_EX interprets a period following a lowercase letter as the end of a sentence, Prof. Svenonius should be typeset as Prof._Svenonius.

1. Fine-tuning text

1. Word and character spacing

- (1) T_EX inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.
- (2) T_EX interprets a period following a lowercase letter as the end of a sentence, **Prof. Svenonius** should be typeset as **Prof.** _ **Svenonius**.
- (3) An uppercase letter followed by a period is interpreted as not being at the end of a sentence, so \@ has to be inserted before the period to ensure extra spacing: **at CASTL.** is to be typeset as **at** _ **CASTL** \@..

1. Fine-tuning text

1. Word and character spacing

- (1) T_EX inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.
- (2) T_EX interprets a period following a lowercase letter as the end of a sentence, **Prof. Svenonius** should be typeset as **Prof.** _ Svenonius.
- (3) An uppercase letter followed by a period is interpreted as not being at the end of a sentence, so \@ has to be inserted before the period to ensure extra spacing: **at CASTL.** is to be typeset as **at** _ **CASTL** _.
- (4) If you don't want extra spacing between sentences, say \frenchspacing. This will remain in effect until it is switched off by \nonfrenchspacing.

1. Fine-tuning text

1. Word and character spacing

- (1) T_EX inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.
- (2) T_EX interprets a period following a lowercase letter as the end of a sentence, **Prof. Svenonius** should be typeset as **Prof.** `\_Svenonius`.
- (3) An uppercase letter followed by a period is interpreted as not being at the end of a sentence, so `\@` has to be inserted before the period to ensure extra spacing: **at CASTL.** is to be typeset as **at** `\_CASTL\@.`
- (4) If you don’t want extra spacing between sentences, say `\frenchspacing`. This will remain in effect until it is switched off by `\nonfrenchspacing`.
- (5) A small spacing is produced by the command `\,`. It might be useful when typesetting “‘left’ and ‘right’”: “‘\,’left’ and ‘right’\,’”. If your source has “‘‘left’ and ‘right’ it will come out as “‘left’ and ‘right’”.

1. Fine-tuning text

1. Word and character spacing

- (1) T_EX inserts more space between sentences than between words (according to traditional typesetting rules). So it inserts more space after ‘.’ and ‘!’ than after ‘,’. Since ‘.’ is used to end sentences as well as after abbreviations, this sometimes leads to confusion.
- (2) T_EX interprets a period following a lowercase letter as the end of a sentence, **Prof. Svenonius** should be typeset as **Prof.** `\_Svenonius`.
- (3) An uppercase letter followed by a period is interpreted as not being at the end of a sentence, so `\@` has to be inserted before the period to ensure extra spacing: **at CASTL.** is to be typeset as **at** `\_CASTL\@.`
- (4) If you don’t want extra spacing between sentences, say `\frenchspacing`. This will remain in effect until it is switched off by `\nonfrenchspacing`.
- (5) A small spacing is produced by the command `\,`. It might be useful when typesetting “‘left’ and ‘right’”: “‘\,’left’ and ‘right’\,’”. If your source has “‘‘left’ and ‘right’ it will come out as “‘left’ and ‘right’”.

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

`Here_is\hspace{1cm}1cm`

Here is 1cm

`Here_is_\hspace{1cm}1cm`

Here is 1cm

`Here_is_\hspace{1cm}_1cm`

Here is 1cm

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

<code>Here_is\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}_1cm</code>	Here is	1cm

(7) The length specification can also be negative, then it puts two items closer together.

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

<code>Here_is\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}_1cm</code>	Here is	1cm

(7) The length specification can also be negative, then it puts two items closer together.

(8) `\hfill` is a command for inserting a 'rubber' length: it takes up all and only the amount of horizontal space available. (It's short for `\hspace{\fill}`, where `\fill` is a special 'rubber' length).

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

<code>Here_is\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}_1cm</code>	Here is	1cm

(7) The length specification can also be negative, then it puts two items closer together.

(8) `\hfill` is a command for inserting a 'rubber' length: it takes up all and only the amount of horizontal space available. (It's short for `\hspace{\fill}`, where `\fill` is a special 'rubber' length).

`1\hfill2` produces 1

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

<code>Here_is\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}_1cm</code>	Here is	1cm

(7) The length specification can also be negative, then it puts two items closer together.

(8) `\hfill` is a command for inserting a 'rubber' length: it takes up all and only the amount of horizontal space available. (It's short for `\hspace{\fill}`, where `\fill` is a special 'rubber' length).

<code>1\hfill2</code> produces 1	2
----------------------------------	---

<code>1\hfill2\hfill3</code> produces 1	2	3
---	---	---

(6) Inserting horizontal space

`\hspace{length}` inserts the specified amount of blank space at the point where it appears in the text, but not at the beginning of an output line.

`\hspace*{length}` inserts the specified amount of space even when it's output at the beginning of a line.

A blank before or after `\hspace` also appears in the output.

<code>Here_is\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}1cm</code>	Here is	1cm
<code>Here_is_\hspace{1cm}_1cm</code>	Here is	1cm

(7) The length specification can also be negative, then it puts two items closer together.

(8) `\hfill` is a command for inserting a 'rubber' length: it takes up all and only the amount of horizontal space available. (It's short for `\hspace{\fill}`, where `\fill` is a special 'rubber' length).

<code>1\hfill2</code> produces 1	2
----------------------------------	---

<code>1\hfill2\hfill3</code> produces 1	2	3
---	---	---

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

`1\hrulefill2\hrulefill3` gives 1_____2_____3

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

`1\hrulefill2\hrulefill3` gives 1_____2_____3

(10) `\quad` inserts the amount of horizontal space equal to the current typesize (e.g., 10 pt for a 10 pt font); `\qquad` inserts the double.

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

`1\hrulefill2\hrulefill3` gives 1_____2_____3

(10) `\quad` inserts the amount of horizontal space equal to the current typesize (e.g., 10 pt for a 10 pt font); `\qquad` inserts the double.

2. Vertical spacing

(11) `\vspace{length}` inserts the specified amount of vertical space at the point where it appears in the text, but not at the beginning of a new page.

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

`1\hrulefill2\hrulefill3` gives 1_____2_____3

(10) `\quad` inserts the amount of horizontal space equal to the current typesize (e.g., 10 pt for a 10 pt font); `\qquad` inserts the double.

2. Vertical spacing

(11) `\vspace{length}` inserts the specified amount of vertical space at the point where it appears in the text, but not at the beginning of a new page.

`\vspace*{length}` inserts the specified amount of vspace even when it's output at the beginning of a page.

(9) `\dotfill` and `\hrulefill` work in the same way:

`1\dotfill2` gives 1 2

`\dotfill\dotfill\dotfill text` gives text

`1\hrulefill2\hrulefill3` gives 1_____2_____3

(10) `\quad` inserts the amount of horizontal space equal to the current typesize (e.g., 10 pt for a 10 pt font); `\qquad` inserts the double.

2. Vertical spacing

(11) `\vspace{length}` inserts the specified amount of vertical space at the point where it appears in the text, but not at the beginning of a new page.

`\vspace*{length}` inserts the specified amount of vspace even when it's output at the beginning of a page.

(12) `\vfill` inserts a rubber length. `\vfill Some text \vfill` centers `Some text` vertically:

`Some text`

- (13) The commands `\bigskip`, `\medskip` and `\smallskip` insert vertical space depending on the font size.

- (13) The commands `\bigskip`, `\medskip` and `\smallskip` insert vertical space depending on the font size.

3. Line breaking

(14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines.
`\\*[length]` makes sure the two lines appear on the same page.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines. `\\*[length]` makes sure the two lines appear on the same page.
- (16) A line break can be encouraged with `\linebreak[num]`, where *num* is a number between 0 and 4. *0* allows a line break where it normally wouldn't occur, such as in the middle of a word. `\linebreak[4]` is equivalent to `\linebreak`, compelling a line break. The line is justified.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines. `\\*[length]` makes sure the two lines appear on the same page.
- (16) A line break can be encouraged with `\linebreak[num]`, where *num* is a number between 0 and 4. *0* allows a line break where it normally wouldn't occur, such as in the middle of a word. `\linebreak[4]` is equivalent to `\linebreak`, compelling a line break. The line is justified.
- (17) `\nolinebreak` has the opposite effect; its syntax is identical to that of `\linebreak`.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines. `\\*[length]` makes sure the two lines appear on the same page.
- (16) A line break can be encouraged with `\linebreak[num]`, where *num* is a number between 0 and 4. *0* allows a line break where it normally wouldn't occur, such as in the middle of a word. `\linebreak[4]` is equivalent to `\linebreak`, compelling a line break. The line is justified.
- (17) `\nolinebreak` has the opposite effect; its syntax is identical to that of `\linebreak`.
- (18) `~` is a non-breakable space. `Section~1` makes sure *Section 1* appears in the same line.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines. `\\*[length]` makes sure the two lines appear on the same page.
- (16) A line break can be encouraged with `\linebreak[num]`, where *num* is a number between 0 and 4. *0* allows a line break where it normally wouldn't occur, such as in the middle of a word. `\linebreak[4]` is equivalent to `\linebreak`, compelling a line break. The line is justified.
- (17) `\nolinebreak` has the opposite effect; its syntax is identical to that of `\linebreak`.
- (18) `~` is a non-breakable space. `Section~1` makes sure *Section 1* appears in the same line.
- (19) Putting text in an `\mbox` has the same effect: `\mbox{$n$-tuple}` ensures *n-tuple* stays in the same line.

3. Line breaking

- (14) A new line can be forced with `\\` or `\newline`. The last line is left aligned.
- (15) With the optional argument `\\[length]`, you can specify space between the two lines. `\\*[length]` makes sure the two lines appear on the same page.
- (16) A line break can be encouraged with `\linebreak[num]`, where *num* is a number between 0 and 4. *0* allows a line break where it normally wouldn't occur, such as in the middle of a word. `\linebreak[4]` is equivalent to `\linebreak`, compelling a line break. The line is justified.
- (17) `\nolinebreak` has the opposite effect; its syntax is identical to that of `\linebreak`.
- (18) `~` is a non-breakable space. `Section~1` makes sure *Section 1* appears in the same line.
- (19) Putting text in an `\mbox` has the same effect: `\mbox{$n$-tuple}` ensures *n-tuple* stays in the same line.

4. Page breaking

(20) A page break is forced with `\newpage`.

4. Page breaking

(20) A page break is forced with `\newpage`.

(21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.

4. Page breaking

- (20) A page break is forced with `\newpage`.
- (21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.
- (22) Enclosing text in a `\vbox` makes sure all of it is typeset on the same page.

4. Page breaking

- (20) A page break is forced with `\newpage`.
- (21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.
- (22) Enclosing text in a `\vbox` makes sure all of it is typeset on the same page.

5. Paragraph formatting

- (23) `\parindent` is the length by which the first lines of a paragraph are indented. It can be modified by the length commands.

4. Page breaking

- (20) A page break is forced with `\newpage`.
- (21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.
- (22) Enclosing text in a `\vbox` makes sure all of it is typeset on the same page.

5. Paragraph formatting

- (23) `\parindent` is the length by which the first lines of a paragraph are indented. It can be modified by the length commands.
- (24) `\parskip` is vertical spacing between paragraphs. Should be defined in `ex` or `em`. It's a rubber length: `\renewcommand{\parskip}{length1pluslength2minuslength3}`.

4. Page breaking

- (20) A page break is forced with `\newpage`.
- (21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.
- (22) Enclosing text in a `\vbox` makes sure all of it is typeset on the same page.

5. Paragraph formatting

- (23) `\parindent` is the length by which the first lines of a paragraph are indented. It can be modified by the length commands.
- (24) `\parskip` is vertical spacing between paragraphs. Should be defined in `ex` or `em`. It's a rubber length: `\renewcommand{\parskip}{length1pluslength2minuslength3}`.
- (25) `\baselinestretch` is a number that magnifies the normal distance between baselines. `\renewcommand{\baselinestretch}{1.5}` enlarges line spacing by 50%. If given outside the preamble, it will only take effect when a new font size is declared.

4. Page breaking

- (20) A page break is forced with `\newpage`.
- (21) `\pagebreak` and `\nopagebreak` function just like `\linebreak` and `\nolinebreak`.
- (22) Enclosing text in a `\vbox` makes sure all of it is typeset on the same page.

5. Paragraph formatting

- (23) `\parindent` is the length by which the first lines of a paragraph are indented. It can be modified by the length commands.
- (24) `\parskip` is vertical spacing between paragraphs. Should be defined in `ex` or `em`. It's a rubber length: `\renewcommand{\parskip}{length_1pluslength_2minuslength_3}`.
- (25) `\baselinestretch` is a number that magnifies the normal distance between baselines. `\renewcommand{\baselinestretch}{1.5}` enlarges line spacing by 50%. If given outside the preamble, it will only take effect when a new font size is declared.
- (26) `\noindent` suppresses indentation of the current paragraph; `\indent` ensures that it's

indented.

indented.

- (27) Normally, the first paragraph of a section is not indented, not even with `\indent`. The package `indentfirst` changes this (just say `\usepackage{indentfirst}` in the preamble).

indented.

- (27) Normally, the first paragraph of a section is not indented, not even with `\indent`. The package `indentfirst` changes this (just say `\usepackage{indentfirst}` in the preamble).
- (28) The package `parskip` sets `\parindent` to zero and adds white space between paragraphs (and also modifies some of the parameters for lists).

indented.

- (27) Normally, the first paragraph of a section is not indented, not even with `\indent`. The package `indentfirst` changes this (just say `\usepackage{indentfirst}` in the preamble).
- (28) The package `parskip` sets `\parindent` to zero and adds white space between paragraphs (and also modifies some of the parameters for lists).

6. Page formatting

- (29) Pages in L^AT_EX are quite complex. You can get a glimpse with the package `layout`. The command `\layout` outputs a diagram of the current page.
- (30) The command `\enlargethispage{length}` adds *length* to the `\textheight` of the current page. `\enlargethispage*{length}` also shrinks interline spacing.

- (31) If a document contains big figures or tables, TeX will increase interline spacing to ensure every page is of the same height. To prevent this, say `\raggedbottom` in the preamble.

- (31) If a document contains big figures or tables, T_EX will increase interline spacing to ensure every page is of the same height. To prevent this, say `\raggedbottom` in the preamble.
- (32) However, this might cause footnotes to appear right after the text on the page ends, instead of the bottom of the page. To avoid this, say `\usepackage[bottom]{footmisc}` after `\raggedbottom`.

Customising page layout

- Headers and footers with `fancyhdr`
- Margins with `geometry`
- Section titles with `titlesec`

2. Customising headers and footers

(33) Package: `fancyhdr` (part of most T_EX distributions).

```
\usepackage{fancyhdr}
```

```
\pagestyle{fancy}
```

```
\fancyhead[L]{Your Name}
```

```
\fancyfoot[L]{\roman{page}}
```

2. Customising headers and footers

(33) Package: `fancyhdr` (part of most $\text{T}_\text{E}\text{X}$ distributions).

```
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead[L]{Your Name}
\fancyfoot[L]{\roman{page}}
```

(34) To cancel previous header/footer settings:

```
\fancyhead{}
\fancyhead[L]{Your Name}
\fancyhead[R]{The title of the paper}
\fancyfoot{}
\fancyfoot[L]{\roman{page}}
```

(35) with the option `twoside`

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(35) with the option `twoside`

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(36) Changing the rule in headers/footers:

```
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0.1ex}
```

(37) Different headers and footers on even and odd-numbered pages

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(37) Different headers and footers on even and odd-numbered pages

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(38) Adding vertical rules to headers and footers

```
\fancyfoot{}
\fancyfoot[OL,ER]{\hrulefill\\ This is page \roman{page}%
\\ \vspace{-1ex}\hrulefill}
\fancyfoot[OR,EL]{\hrulefill\\ \LaTeX{} rocks!\\ \vspace{-1ex}\hrulefill}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
```

(39) Alternative ways to define text in headers and footers with fancyhdr:

```
\lhead{Left head}   \chead{Centre head}   \rhead{Right head}  
\lfoot{Left foot}   \cfoot{Centre foot}   \rfoot{Right foot}
```

(39) Alternative ways to define text in headers and footers with `fancyhdr`:

```
\lhead{Left head}   \chead{Centre head}   \rhead{Right head}  
\lfoot{Left foot}   \cfoot{Centre foot}   \rfoot{Right foot}
```

(40) If you define the headers and footers in the preamble, the definitions will have scope over the whole document. If you change them in the body of the document, the changes will take effect from the next page onwards.

3. Changing margins

(41) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

3. Changing margins

(41) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

(42) Changing the margins

```
\usepackage{geometry}  
\geometry{a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm}
```

3. Changing margins

(41) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

(42) Changing the margins

```
\usepackage{geometry}  
\geometry{a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm}
```

(43) alternatively:

```
\usepackage[a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm]{geometry}
```

(44) If all margins are the same:

```
\usepackage[a4paper,margin=1in]{geometry}
```

(45) For many more things **geometry** can do, see the documentation.

4. Redefining section titles

(46) Package: titlesec

```
\titleformat{command}[shape]{format}{label}{sep}{before}
```

command is the sectioning command to be redefined, e.g. `\section`.

shape sets the paragraph shape. See the package documentation for possible values.

format is the set of commands that are applied to the whole title (the label and the text as well).

label defines the label of the title.

sep is the horizontal space between the label and the text in the title. It must be defined as a length.

before is the code preceding the title body.

4. Redefining section titles

(46) Package: titlesec

```
\titleformat{command}[shape]{format}{label}{sep}{before}
```

command is the sectioning command to be redefined, e.g. `\section`.

shape sets the paragraph shape. See the package documentation for possible values.

format is the set of commands that are applied to the whole title (the label and the text as well).

label defines the label of the title.

sep is the horizontal space between the label and the text in the title. It must be defined as a length.

before is the code preceding the title body.

(47) An example

```
\usepackage{titlesec}  
\titleformat{\section}{\itshape}{\thesection.}{1em}{\bfseries}
```

(48) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

(48) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

```
\usepackage{titlesec}  
\titlespacing{\section}{3em}{1em}{1em}
```

(48) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

```
\usepackage{titlesec}  
\titlespacing{\section}{3em}{1em}{1em}
```

Using the starred version of `titlespacing` makes sure the paragraph following the title is not indented.

```
\usepackage{titlesec}  
\titlespacing*{\section}{3em}{1em}{1em}
```

(49) Fixed width labels

This might be necessary if a journal's style sheet requires paragraph indentation to be the same as the indentation of the title text.

```
\titleformat{\section}{\itshape }{\makebox[3em]{\thesection.}}{0em}{\bfseries}  
\titlespacing{\section}{0em}{1em}{1em}  
\setlength{\parindent}{3em}
```

(50) This package can do much, much more – check out the documentation if you're interested.

Homework

Take your sample article (HW 1) and modify it in the following ways.

1. Headers and footers:
 - (a) Odd page header: your name on the right in italics, the page number in capital Roman numbers on the left, slanted
 - (b) Even page header: the title of your article on the left in bold italics, the page number in capital Roman numbers on the right, slanted
 - (c) Odd page footer: the name of the journal on the left
 - (d) Even page footer: the starting and ending page numbers of the article on the right (automatically, not entered manually). The article should start on page 24.
 - (e) The header and the footer are separated from the text body by a 1pt wide rule.
2. All margins should be 2.5 cm. The header and footer should be included in the total body of the text (there should be 2.5 cm between the top of the header and the top of the page, as well as the bottom of the footer and the bottom of the page).
3. Section titles

- (a) Section titles should be slanted and bold, section numbers should be slanted but not bold. Section numbers should be followed by period. The same applies to subsection titles.
- (b) There should be one empty line between (sub)section titles and preceding text, and no lines between (sub)section titles and the following text.