The exesheet class and package
Antoine Missier
antoine.missier@ac-toulouse.fr
2023/09/05, v2.2

Contents

1 Introduction .................................................. 2

2 Titles .................................................................. 2
   2.1 The \exercise command .................................. 2
   2.2 The \subpart command .................................. 3
   2.3 The \annex command ................................. 4
   2.4 Exercise titles in the table of contents ............... 4
   2.5 Short exercises: the \exe command ................. 4

3 Enumerations and lists ..................................... 5
   3.1 List settings ............................................. 5
   3.2 List of exercises: the exenumrate environment ..... 5
   3.3 Items aligned by row: tablenum1, tablenuma, tablitem .. 6
   3.4 Items aligned by column: colsenum, colsiem ........ 7

4 Questions and solutions ................................... 8
   4.1 Environments questions and answers ............... 8
   4.2 More about answers environments .................... 9
   4.3 Commands \question, \answer and \answerspace .... 10

5 Marginal notes for marking scheme .................. 10
   5.1 The \points command .................................. 10
   5.2 The \pts command ..................................... 11
   5.3 Commands \totalexe, \note* and \note .............. 11
   5.4 Margin notes options .................................. 13
   5.5 The \totalpoints command ........................... 14
   5.6 Marking scheme consistency checking ............... 14

6 Options .......................................................... 15
   6.1 Summary of available options ....................... 15
   6.2 Alternative (deprecated) commands .................. 15

7 Implementation .............................................. 16
   7.1 Options and required packages ...................... 16
   7.2 Internationalization .................................. 17
   7.3 Titles ..................................................... 19
   7.4 Enumerations and lists ................................ 21
   7.5 Questions and answers ................................ 23
   7.6 Marking scheme options processing ............... 26
   7.7 Marginal notes commands ............................ 32
   7.8 The correct option and other (deprecated) commands .. 34
1 Introduction

The exesheet package is designed for typesetting exercise or exam sheets. Additionally, the exesheet class loads the schooldocs package. The latter makes adjustments to margins and titles, and defines various layout styles with specific headers and footers suitable for exercise sheets, among other uses. Refer to the documentation of the schooldocs package for more details. The exesheet class is build upon the article class and forwards any unknown options to it.

There are many other packages dedicated to exercise sheets. Most of them suggest encapsulating each exercise within an environment. In contrast, exesheet starts each exercise with \exercise, which functions similarly to a subsection (with the same features) and is suitable for documents that primarily consist of exercises. The package also offers alternative ways to introduce exercises, which are more appropriate for shorter exercises.

Another distinctive feature of the exesheet package is its specific settings for enumeration lists, which are suitable for numbering questions or answers within an exercise.

While other packages often offer more or less complex mechanisms for managing the placement of answers, exesheet doesn’t aspire to such complexity. However, for all exercises within the sheet, you have the capability to display only questions, only answers, or both, all while preserving their placement as they appear in the source file. This choice allows for great flexibility: you can create a correct version for all exercises collectively, or individual corrections per exercise, per part (subpart of exercise), per question, per sub-question.

Finally this package enables to display a detailed marking scheme in the margin, with optional explanations or remarks, and offering consistency control.

Many settings can be customized, and various options are available to manage the output document. These options rely on the key-val mechanism: key=value. These options can be applied when calling the class or the package, e.g.

\documentclass[a4paper,11pt,output=answers,display=pts]{exesheet}

or later using the command \exesheetset{(options)}. In the example above, a4paper,11pt are options that are passed to the underlying article class.

In the current document, a frame is utilized to emphasize output examples.

2 Titles

2.1 The \exercise command

The \exercise[(opt)] command initiates an exercise with the title Exercise, typeset as a document subsection, followed by automatic numbering, unique to the entire document. The optional parameter (opt) is utilized to include additional text on the same title line, such as specifying a subject or a marking scheme. Thus, using \exercise[(to begin)] results in:

Exercise 1 (to begin)

Try this first command; easy!
To bring optional text closer to the exercise number, you can employ \unskip which removes any preceding space. Take a look at the following example, achieved with \exercise[\unskip*** (difficult)]:

**Exercise 2*** (difficult)

Calculate 1 + 1.

\texttt{\textbackslash exercisename}  

The term “Exercise” is automatically translated into various languages\footnote{Currently, translation is integrated into the package for the following languages: French, German, Spanish, Italian, and Portuguese.} depending on the language that is loaded (via \texttt{babel} or \texttt{polyglossia}). You can redefine it using \texttt{\renewcommand}. A better approach is to use macros from the \texttt{translations} package by Clemens Niederberger (which allows dynamic language switching), e.g. \texttt{\DeclareTranslation{Swedish}{esheet-exercise}{"Ovning}}.

\texttt{\textbackslash labelexercise}  

This command combines \texttt{\textbackslash exercisename} with the exercise number and can be redefined. For instance, if you want to include a period after the exercise number, you can redefine it as follows: \texttt{\renewcommand{\labelexercise}{\textbackslash exercisename\textbackslash theexercise.}}

\texttt{\textbackslash theexercise}  

If you wish to alter only the numbering style, you can redefine the \texttt{\theexercise} command based on the \texttt{\exercise} counter.

\texttt{\textbackslash labelexercisestyle}  

This macro, which is initially empty, enables the definition of a specific style for exercise titles. In this document, we have set the following in the preamble: \texttt{\renewcommand{\labelexercisestyle}{\textbackslash rmfamily\textbackslash color{black}}}\footnote{In this document, real section and subsection titles have been highlighted by modifying their color and font (sans serif) using the \texttt{\allsectionsfont} macro from the \texttt{sectsty} package.}

\texttt{\exercise*}  

The starred version \texttt{\exercise*[(\texttt{opt})]{(\texttt{label})}} permits the selection of an alternative \texttt{(label)} for a specific exercise while omitting the numbering. For instance: \texttt{\exercise*[(Fermat’s theorem)]{Problem}} results in:

**Problem (Fermat’s theorem)**

Prove that there are no positive integers $x, y, z$ such that $x^n + y^n = z^n$ for any integer $n$ greater than 2.

\texttt{\textbackslash subpart}  

An exercise may consist of multiple parts, which can be created using the \texttt{\subpart[(\texttt{opt})]} command. The part title is typeset similar to a sub-subsection.

**Exercise 3**

\texttt{\textbackslash subpart}[(\texttt{opt})]  

To begin, prepare your cup of tea.

\texttt{\textbackslash subpart}  

Now you are ready to proceed with the current exercise.
The following macros manage formatting in the same manner as for \exercise.

\thesubpart By default, subpart numbering employs letters: A, B, C, and so on. This numbering style can be modified using the \thesubpart command, which relies on the subpart counter. For example, you can redefine it as follows: \renewcommand\thesubpart{\arabic{subpart}}.

\subpartname The \subpart command utilizes \subpartname (with automatic translation in several languages according to the chosen language), as well as \labelsupart and \labelsupartstyle, all of which can be modified.

\subpart* Similar to \exercise*, the starred version \subpart*{⟨opt⟩}{⟨label⟩} permits an alternative ⟨label⟩ and omits the numbering. For instance, you can use \subpart*{First part}.

2.3 The \ annex command

\annex The \annex{⟨opt⟩} command composes the title ANNEX in uppercase letters, centered, using the subsection style, with an optional parameter that will be added on the same line.

\annexname The term “Annex” is automatically translated into several languages (depending on the chosen language). It can be extended to additional languages or altered by redefining \annexname or by utilizing macros from the translations package.

\annexstyle The style of the annex title is determined by the \annexstyle macro, which is defined as follows: \newcommand\annexstyle{\MakeUppercase}. This command may be redefined according to your preferences.

2.4 Exercise titles in the table of contents

\[exetoc={⟨bool⟩}\] By default, the titles Exercise, Part and Annex are included in the table of contents if there is any (or in the PDF file’s summary when the hyperref package is utilized). To prevent this, you can set the package option exetoc=false (with the default being true). However, note that optional title arguments will always be ignored in the table of contents.

2.5 Short exercises: the \exe command

\exe The \exe command initiates an exercise with the abbreviation Ex. followed by the exercise number. This is achieved without utilizing sectioning commands, and the exercise content begins on the same line. An exercise begins a new paragraph without any indentation.

\Ex. 4 — This is a brief exercise that can encompass several paragraphs or questions.

Here for example a new paragraph begins.

\Ex. 5 — This is another concise exercise.
The abbreviation \texttt{Ex} can be modified by redefining \texttt{exname} or with macros from the \texttt{translations} package. The \texttt{exlabel} macro combines \texttt{exname} with a period then the exercise number (given by the \texttt{exercise} counter), while \texttt{exsepmark} typesets a long dash. You can alter these characteristics by redefining these commands.

The starred version doesn’t display a separator, as demonstrated below:

\begin{verbatim}
\exe*
\end{verbatim}

Another short exercise without a separator.

\section{Enumerations and lists}

\subsection{List settings}

Enumeration lists are used to represent questions and sub-questions within exercises. To provide clear emphasis, labels are formatted in bold. Additionally, these labels are aligned to the left, positioned at the start of the line without indentation, and the vertical spacing between items is increased compared to standard \LaTeX\ lists. These formatting adjustments are achieved using the \texttt{setlist} command, a feature from the \texttt{enumitem} package by Javier Bezos.

\begin{exercise}
1. First question
   \begin{enumerate}
   \item First sub-question
   \item Second sub-question
   \end{enumerate}
2. Second question
\end{exercise}

The \texttt{enumerate} environment takes an optional parameter, that allows, among others things, the typesetting of alternative list labels. For instance, typing \begin{enumerate}[label={\textit{\textup{A}.}},font=\textit{\textup{normalfont}}] will yield the labels “\textit{a}), \textit{b}), \textit{c})…” . There are many other options available (see the \texttt{enumitem} package documentation). You can change global label font formatting using the syntax \texttt{\setlist[enumerate]{font=\ldots}} (called after \texttt{\begin{document}}).

Lists created with the \texttt{itemize} environment retain their default configuration. The package option \texttt{setlist=false} prevents changes to enumeration and itemize lists and reverts to the default \LaTeX\ settings (the default value is \texttt{true}).

\subsection{List of exercises: the \texttt{exenumerate} environment}

When an exercise sheet consists of short, independent questions, it might be unreasonable to display the full title \texttt{Exercise} for each one. In addition to the previously mentioned \texttt{exe} command, we offer an even more streamlined solution using the

\footnote{Labels can also be modified using a “shortlabel” argument, e.g. \texttt{\begin{enumerate}[A.],} or globally through the redefinition of \texttt{\labelenumi} or \texttt{\labelenumii} commands.}

\footnote{The \texttt{french} option of the \texttt{babel} package changes the behavior of \texttt{itemize} lists and employs long dashes as labels for each list level. This behavior can cause issues when mathematical content follows the dash symbol, as it might be mistaken for the minus sign. Thus the default \texttt{itemize} list style is reinstated.}

\section{3. Enumerations and lists}

\subsection{List settings}

Enumeration lists are used to represent questions and sub-questions within exercises. To provide clear emphasis, labels are formatted in bold. Additionally, these labels are aligned to the left, positioned at the start of the line without indentation, and the vertical spacing between items is increased compared to standard \LaTeX\ lists. These formatting adjustments are achieved using the \texttt{setlist} command, a feature from the \texttt{enumitem} package by Javier Bezos.

\begin{exercise}
1. First question
   \begin{enumerate}
   \item First sub-question
   \item Second sub-question
   \end{enumerate}
2. Second question
\end{exercise}

The \texttt{enumerate} environment takes an optional parameter, that allows, among others things, the typesetting of alternative list labels. For instance, typing \begin{enumerate}[label={\textit{\textup{A}.}},font=\textit{\textup{normalfont}}] will yield the labels “\textit{a}), \textit{b}), \textit{c})…” . There are many other options available (see the \texttt{enumitem} package documentation). You can change global label font formatting using the syntax \texttt{\setlist[enumerate]{font=\ldots}} (called after \texttt{\begin{document}}).

Lists created with the \texttt{itemize} environment retain their default configuration. The package option \texttt{setlist=false} prevents changes to enumeration and itemize lists and reverts to the default \LaTeX\ settings (the default value is \texttt{true}).

\subsection{List of exercises: the \texttt{exenumerate} environment}

When an exercise sheet consists of short, independent questions, it might be unreasonable to display the full title \texttt{Exercise} for each one. In addition to the previously mentioned \texttt{exe} command, we offer an even more streamlined solution using the
exenumerate environment. This environment is essentially an enumeration list with increased spacing between items, compared to the enumerate environment. Here is an example (the main list uses the exenumerate environment, while the sub-list is created using the standard enumerate environment):

1. Translate the following sentences in English:
   (a) Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi.
   (b) Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus.

2. Translate the following sentence in German:
   Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi.

3. Translate the following sentence in French: Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

The exenumerate environment (also based on the enumitem package) accepts an optional parameter, similar to the enumerate environment.

3.3 Items aligned by row: tablenum1, tablenuma, tablitem

These three environments are employed to typeset brief questions (tablenum1), sub-questions (tablenuma) or itemize lists (tablitem) on the same line. They share the same syntax: \begin{tablenum1}(⟨opt⟩)(⟨cols⟩). The ⟨cols⟩ parameter denotes the number of columns utilized by the environment. It must be enclosed in parentheses. This parameter can be omitted, in which case its default value is 2. Similar to conventional lists, each item is initiated with the \item command.

Internally we have utilized the \NewTasksEnvironment macro from the tasks package by Clemens Niederberger. The usage of the optional argument ⟨opt⟩ is explained in the documentation of this package. For example, similar to the enumitem package, label=\arabic*) produces an Arabic numbering followed by a closing parenthesis. Additionally there are numerous possibilities for arranging items in original ways. For instance, the \item* command allows you to specify the number of columns the item is supposed to span. In the subsequent example, the five \item commands are sequentially positioned between \begin{tablenum1}(3) and \end{tablenum1}. Notice that numbering occurs line by line in this context.

Exercise 8

Calculate the derivative of the following functions:

1. \( f(x) = \frac{1 - x^2}{e^x + e^{-x}} \),
2. \( g(x) = \ln \left( \frac{1 - x}{1 + x^2} \right) \),
3. \( h(x) = \int_0^1 e^{xy} \, dy \),
4. \( k(x) = \sum_{i=1}^{\infty} \frac{1}{x^i} \),
5. \( l(x) = \int_{\frac{1}{2}}^{x} \frac{1}{\ln t} \, dt \).
For \texttt{tablenuma}, labels are letters, a, b, c,\ldots, enclosed in parentheses.

You can change the labels by redefining the macros \texttt{\labelenumone} (for \texttt{tablenum1}) and \texttt{\labelenuma} (for \texttt{tablenuma}), using the \texttt{task} counter, e.g. \texttt{\renewcommand\labelenuma{\Alph{task}.}} yields the labels A., B.,\ldots

If the \texttt{exesheet} package is invoked with the option \texttt{setlist=false}, labels within \texttt{tablenum1} and \texttt{tablenuma} environments will be presented with indentation and in normal font rather than bold. You can change the label formatting globally with the command \texttt{\settask}, e.g. \texttt{\settask{label-format=\itshape}}. You can also completely redefine the environments using \texttt{\RenewTasksEnvironment}. When \texttt{setlist=true}, place these commands after \texttt{\begin{document}}.

When you intend to utilize \texttt{tablenuma} (or \texttt{tablitem}) immediately after inserting the \texttt{\item} command within an \texttt{enumerate} environment, a vertical misplacement occurs, as demonstrated in the following example:

1. (a) One (b) Two (c) Three

To achieve proper vertical spacing in such cases, we offer the starred environments \texttt{tablenuma*} and \texttt{tablitem*}, with corrected alignment as shown below:

\begin{verbatim}
1. (a) One (b) Two (c) Three.
\end{verbatim}

If the vertical alignment is still not perfect, include \texttt{\mbox{\vspace{⟨height⟩}}} just after \texttt{\item} and before invoking \texttt{\begin{tablenuma*}} (or \texttt{\begin{tablitem*}}), where \texttt{⟨height⟩} can be a positive or negative length.

### 3.4 Items aligned by column: colsenum, colsitem

To achieve numbering of items by column, the \texttt{colsenum} environment is available: \texttt{\begin{colsenum}{⟨opt⟩}{⟨cols⟩}}. The mandatory parameter is the number of columns, and the optional parameter will be passed to the underlying \texttt{enumerate} environment, allowing you to change the numbering type (e.g. a, A, etc.), among other possibilities. To use this environment, you need to load the \texttt{multicol} package in the preamble. Here’s an example with \texttt{\begin{colsenum}{3}}:

**Exercise 9**

Calculate the derivative of the following functions:

1. \( f(x) = \frac{1 - x^2}{e^x + e^{-x}}, \)
2. \( g(x) = \ln \left( \frac{1 - x}{1 + x^2} \right), \)
3. \( h(x) = \int_0^1 e^{xy} \, dy, \)
4. \( k(x) = \sum_{i=1}^{\infty} \frac{1}{x^i}, \)
5. \( l(x) = \int_{\frac{1}{2}}^x \frac{1}{\ln t} \, dt. \)

We will observe that, on each line, items are not necessarily properly aligned, which can result in ungraceful effects. On the other hand, the \texttt{colsenum} environment doesn’t attempt to align columns from the bottom by adjusting the vertical spacing between items. If we desire this alignment (which is the default behavior in \texttt{multicol}), we can use the \texttt{colsenum*} environment (with the same syntax as \texttt{colsenum}). Here’s what we obtain with \texttt{colsenum*}:
Exercise 10

Calculate the derivative of the following functions:

1. \( f(x) = \frac{1 - x^2}{e^x + e^{-x}} \)
2. \( g(x) = \ln \left( \frac{1 - x}{1 + x^2} \right) \)
3. \( h(x) = \int_0^1 e^{xy} \, dy \)
4. \( k(x) = \sum_{i=1}^{\infty} \frac{1}{x^i} \)
5. \( l(x) = \int_{\frac{1}{2}}^{x} \frac{1}{\ln t} \, dt \)

We can observe that these alignments are not as elegant as those achieved through row numbering. However, column numbering might still be more suitable when dealing with numerous items of varying heights, and especially when the number of items can differ from column to column. Additionally, a benefit of `colsenum` is that the label selection is automatic, based on the list level (and the language), unlike `tablenum1` or `tablenuma`.

For `itemize` lists, the `colsitem` environment generates items aligned by column, unlike the line-by-line alignment of `tablitem`. It follows the same syntax as `colsenum`: `\begin{colsitem}[⟨opt⟩]{⟨cols⟩}`. The optional parameter, passed to the underlying `itemize` environment, permits the modification of the item label (bullet by default). Furthermore, just like `colsenum*`, the `colsitem*` environment produces column alignment from the bottom.

4 Questions and solutions

4.1 Environments questions and answers

The package offers two environments, `questions` and `answers`, which allow you to optionally show or hide questions and answers within exercises.

[output=(opt)]

The output is governed by the `output` key option which recognizes three values: `questions`, `answers`, and `both`. The `questions` value shows only questions without answers, `answers` displays answers without questions, and `both` (the default option) displays both questions and answers.

\texttt{\textbackslash correctionstyle}

In the default case where both questions and answers are displayed, the answers are typeset using the `\correctionstyle` style, which utilizes the color `correctioncolor`. You can modify this color using the `\definecolor` macro\(^5\). By default, `\definecolor{correctioncolor}{rgb}{0.0, 0.2, 0.6}` is used, resulting in a kind of dark blue.

\texttt{\textbackslash correctionname}

Furthermore, when using `output=both` the title `\texttt{Correction}` is displayed at the beginning of `answers` environments. This title is defined by the `\correctionname` macro, with translation available in several languages, and it can also be modified. For instance you might prefer “Solution” over “Correction”. The style defined by `\correctionstyle` will be applied to the title as well as the entire environment. Here’s an example to illustrate this:

\(^5\)The `\definecolor` command is provided by the `xcolor` package developed by Uwe Kern, which is automatically loaded by `exesheet`. 
Exercise 11

1. Is the \textit{exesheet} package useful?

2. Aren’t there any other packages that deal with exercises?

\textbf{Correction}

1. Yes, the \textit{exesheet} package is indeed useful for teachers.

2. There are numerous other packages that handle exercises and provide the capability to create questions and solutions separately. For instance the \textit{exercise} package by Paul Pichaureau, \textit{exercises} by Roger Jud, \textit{exsheets} (now superseded by \textit{xsim}) by Clemens Niederberger, \textit{exframe} by Niklas Beisert, \textit{exam} by Philip Hirschhorn, \textit{answers} by Mike Piff and Joseph Wright, \textit{probsoln} by Nicola Talbot, and more.

When only answers are displayed, the text color remains black and the word “Correction” is not displayed.

\subsection*{4.2 More about answers environments}

Internally, we have utilized the \texttt{\textbackslash comment} and \texttt{\textbackslash endcomment} macros from the \texttt{versions} package by Uwe Lück. Moreover, the \texttt{versions} package offers the \texttt{\excludeversion{⟨env⟩}} and \texttt{\includeversion{⟨env⟩}} macros which allow for the exclusion or inclusion of any environment \texttt{⟨env⟩}. These “optional” environments can be nested.

However the \texttt{questions} and \texttt{answers} environments serve a broader purpose beyond merely displaying or hiding text. You can choose to have a single answers environment for the entire sheet, or alternatively, have separate answers environments for each exercise, exercise part, question, or sub-question. The format in which the title \texttt{Correction} should appear in the output, and its placement in the table of contents or PDF file summary, depends on the nesting level of the environment. In fact, the rendering of the \texttt{Correction} title and its corresponding table of contents level will be automatically calculated by the environment.

\texttt{\texttt{answers[⟨level⟩]}} However, users might wish to adjust the title’s level themselves. To achieve this, you can manually set the level of the title “Correction” using an optional \texttt{⟨level⟩} argument which is defined as follows: 1 for section-level titles, 2 for subsections (akin to \texttt{Exercise}), 3 for sub-subsections (similar to \texttt{Part}), other numbers for lower levels (which won’t appear in the table of contents or in the PDF file’s summary).

Caution should be taken that, if the \texttt{questions} environment is not used beforehand in the same exercise (or part), the \texttt{answers} environment will consider the correction as global for the entire sheet (or exercise) and will reset the \texttt{exercise} (or \texttt{part}) counter. This can be managed properly with the optional argument. For example, use \texttt{\begin{answers}[2]} to prevent \texttt{exercise} counter reset, or \texttt{\begin{answers}[3]} to prevent \texttt{subpart} counter reset.

\texttt{\texttt{answers\*}} The starred version \texttt{answers\*} doesn’t display the \texttt{Correction} title.
4.3 Commands \question, \answer and \answerspace

\question
\answer

Instead of using questions and answers environments, we can also employ the simpler \question{⟨ques⟩} and \answer{⟨ans⟩} macros. The visibility of ⟨ques⟩ and ⟨ans⟩ content is regulated by the same previous output=⟨opt⟩ key option. This approach might be more fitting for brief questions and answers, such as when you wish to display the answer immediately after each question item. The title “Correction” won’t appear at the start of each answer with the \answer macro. The answers are also formatted using \correctionstyle if output=both. However these commands do not support verbatim text within them, unlike the questions and answers environments which do.

\answerspace

Some teachers are accustomed to providing their students with documents where questions are typeset, leaving blank spaces instead of answers. This layout allows students to fill in their responses on the paper. To achieve this, thanks to a suggestion from Maxime Chupin, we offer the \answerspace{⟨height⟩} macro, in which the parameter ⟨height⟩ is a valid length, e.g. \answerspace{3cm}.

[answerspace=⟨bool⟩] The blank spaces introduced by \answerspace can be displayed or hidden, controlled by the answerspace option key, which can be set to either true or false (the default). The answerspace key option has no effect (equivalent to false) when the answers are displayed (output=answers or both). Of course the \answerspace macro is not meant to be used within answers environments.

5 Marginal notes for marking scheme

The exesheet package enables the display of a detailed marking scheme in the margins, along with optional comments and explanations about answers.

5.1 The \points command

\points

The \points{⟨pts⟩} command displays the number of points awarded for an exercise. It is intended to be included in the optional argument of the \exercise command. In the following example, we used \exercise[\points{5}]:

Exercise 12
Try to read this document to the end without drinking tea and you get five points.

When only the correction is displayed in an exercise, the \points macro doesn’t show the points. Printing answers along with the point scale will be discussed in section 5.5, which includes another \totalpoints macro.

\pointsname \pointname \pointsstyle \pointscolor

The term “points” (or “point” in the singular if ⟨pts⟩ is less than 2) is appended and is automatically translated into several languages (and can also be modified). You can adjust the \points command’s style through \pointsstyle. The color setting (red by default) is managed by pointscolor using \definecolor, for example you can declare: \definecolor{pointscolor}{named}{blue}.

\footnote{However using \points in the optional argument of \exercise is not compatible with the memoir class, as the memoir class redefines section commands.}
5.2 The \pts command

\pts When exercises are typeset using the \exe macro or as a list with the exenumrate environment, the marking scheme can be shown in the margin, aligned with the line where the \pts\{⟨num⟩\} command is placed (typically the first line of the exercise). The ⟨num⟩ parameter represents the number of points assigned to the exercise. Here’s an example with \exe\pts{3}... \exe\pts{1.5}...

| (3 pts) | Ex. 13 — The first short exercise with a marking scheme. |
| (1.5 pt) | Ex. 14 — The second one. |

\ptsname The abbreviation “pts” (or “pt” when the number of points is less than 2) is added automatically using \ptsname or \ptname macros (translated in several languages if babel or polyglossia is loaded). The point’s display color is defined by ptscolor, changeable via \definecolor (red by default). The display style is determined by \ptsstyle, which among other things, adds parenthesis around.

[display=(opt)] The marking scheme visibility is controlled by the display option key. The default option is display=none, keeping the marking scheme hidden. To reveal the marking scheme, use display=pts. More details are available in section 5.4.

[marginpos=(opt)] The positioning of the scale is determined by the marginpos option key, typically left or right. The default value is left even though LATEX positions marginal notes on the right side by default. This option has no impact when display=none.

For a two-sided document, the default behavior is to place text in the outer margin, which is wider than the inner margin (that contains the binding). The outer margin is positioned on the right side on odd pages and on the left side on even pages. Therefore, the marginpos option can also take the values inner or outer. If you specify left or right when the twoside mode is activated, this value will be converted to outer, accompanied by a warning message.

With the twoside mode, marginal notes might occasionally appear on the wrong side of a page. This is a known LATEX bug, and the solution involves using the mparhack package (which exesheet automatically includes for documents in two-side mode) and running LATEX twice. If necessary, a warning message will prompt you to perform the re-run.

5.3 Commands \totalexe, \note* and \note

For a more comprehensive marking scheme, the following commands are available.

\totalexe The \totalexe\{⟨num⟩\} macro displays the total number of points of an exercise. By default, it appears inside an oval box, with the addition of the word “pts” (or “pt”) in bold red. In the following example, the exercise title has been generated using \exercise[\totalexe{4}].

\note* For each answer or solution in the correct version, the \note*\{⟨num⟩\} command indicates the number of points allocated to that question. The appearance slightly varies compared to \pts: by default the number is displayed in bold without the “pts” or “pt” suffix, and without parenthesis. In the following example, for answer 3, we employed \note*{1.5}, placed right after \item.
The \texttt{\textbackslash note\{\texttt{(comment)}\}} macro is utilized to provide additional information regarding the marking scheme and to explain how points are assigned. In the \texttt{(comment)} argument you can use \texttt{\textbackslash \textbackslash} to create a line break or even \texttt{\textbackslash \textbackslash[\texttt{(height)}]} to adjust the line spacing by \texttt{(height)}.

Placing \texttt{\textbackslash note\{\texttt{(num)}\} \textbackslash note\{\texttt{(comment)}\}} at the beginning of an answer is often practical. In such cases LaTeX will align the margin notes vertically, which leads to a warning like: \texttt{LaTeX Warning: Marginpar on page ... moved}. However, this warning is not an issue, as LaTeX can usually handle the arrangement of these marginal notes, stacking them one below the other. Nonetheless, to prevent unnecessary warnings, you can combine both commands into a single one by specifying the number of points as an optional argument of the \texttt{\textbackslash note}\{\texttt{(num)}\}\{\texttt{(comment)}\}.

The initial comment in the following example is generated (immediately after \texttt{\item}) using \texttt{\textbackslash note\{1\}}{0.5 for the anti-derivative}{0.5 for simplifying}.

\begin{exercise}
\hspace{40pt} 4 pts
For each subsequent question, determine whether the statement is true or false. Provide a thorough justification for your answer.

1. \[ \int_{0}^{\sqrt{3}} \frac{1}{x + \sqrt{3}} \, dx = \ln 2, \]
2. \[ \int_{2}^{e} \frac{1}{x \ln x} \, dx = -\ln 2, \]
3. The function \( F \), defined on \( \mathbb{R} \) by \( F(x) = \int_{0}^{x} \frac{1}{t^2 + t + 1} \, dt \), is increasing on \( \mathbb{R} \).

\textbf{Correction}

1. We calculate:
\[ \int_{0}^{\sqrt{3}} \frac{1}{x + \sqrt{3}} \, dx = \left[ \ln (x + \sqrt{3}) \right]_{0}^{\sqrt{3}} = \ln (2\sqrt{3}) - \ln \sqrt{3} = \ln \left( \frac{2\sqrt{3}}{\sqrt{3}} \right) = \ln 2. \]
\textbf{TRUE}.

2. We have \( \frac{1}{x \ln x} = \frac{1}{x} \frac{u(x)}{u'(x)} \) with \( u(x) = \ln x \), which is positive on \([2, e]\]. Hence
\[ \int_{2}^{e} \frac{1}{x \ln x} \, dx = \left[ \ln(\ln x) \right]_{2}^{e} = \ln(\ln e) - \ln(\ln 2) = 1 - \ln(\ln 2) = -\ln(\ln 2). \]
\textbf{FALSE}.

3. The function \( F \), defined on \( \mathbb{R} \) by
\[ F(x) = \int_{0}^{x} \frac{1}{t^2 + t + 1} \, dt, \]
is derivable on \( \mathbb{R} \) and its derivative is such that \( F'(x) = \frac{1}{x^2 + x + 1} \). The denominator is a quadratic polynomial, always positive because its discriminant is \( \Delta = -3 < 0 \). Thus \( F \) is increasing on \( \mathbb{R} \).
\textbf{TRUE}.
\end{exercise}
In the comment for answer 2, a larger vertical space is created with the optional argument `\[2ex]` for line break. The last comment, which isn’t positioned next to the points number, was produced by placing the following on the first line after the formula: `\note{0.5 for $F'$\1 for the sign of $F'$ and conclusion}.` The color and style for displaying points in `\totalex` and `\note*` can be customized using `markingcolor` and `\markingstyle`, respectively. The oval box produced by `\totalex` is created using the `\ovalbox` command from the `fancybox` package (by Timothy Van Zandt), with corner arcs set by `\cornersize{1}`. The box’s length is determined by `\ptsboxlength`, and not by the box’s content, to ensure uniformity across exercises.

By default, comment notes are typeset in a dark green color defined by `\definecolor{notecolor}{rgb}{0.0,0.4,0.0}`. The style of comments is determined by the `\notestyle` macro.

### 5.4 Margin notes options

The display key option governs the presentation of the marking scheme: as discussed previously (subsection 5.2), `display=none` shows nothing. When using `display=pts` the numbers provided as arguments to `\pts`, `\totalex`, `\note*` or as optional arguments of `\note{⟨num⟩}{...}` will be exhibited. The final option is `display=notes` which reveals the complete marginal notes, containing points and comments (the mandatory argument of `\note`), as illustrated in the previous example.

As previously mentioned in subsection 5.2, the side on which to position the scale is determined by the `marginpos` key option, with possible values of `left` and `right` (or `inner` and `outer` if the document is in `twoside` mode).

The margin layout is governed by the `marginwidth` key option, which can take one of the following values: `standard`, `expand`, or `unset`. This option has no effect when `display=none`. In this case, both the left and right margins have the same width, except in a two-sided document where the ratio between the left and right margins is 2:3. Otherwise the `marginwidth` key option behaves as follows:

- **standard** The left margin is widened, and the right margin is reduced, with a ratio of 3:2 (or 2:3 if `marginpos=right`). The text body is shifted without changing its width. The margin paragraph width remains relatively short (depends on page geometry). This option is not ideal for lengthy comments.

- **expand** (default value) The behavior is the same as with the `standard` value when `display=pts`. However, when `display=notes`, the margin expands with a ratio of 3:1 (or 1:3) and the width of margin paragraphs increases.

- **unset** This option is provided for cases where the previous settings are not suitable. In this case, no adjustments are made to the margin width. Instead, you can define your own settings using the convenient `\geometry` macro from the `geometry` package (by Hideo Umeki). For instance, you can place the following in the preamble:

  `\geometry{hmarginratio=2:1,marginparwidth=2.5cm}`.

  If `marginpos=right`, you need to invert the ratio, e.g. 1:2 instead of 2:1. If `marginwidth` is not set to `unset`, such a command will have no effect.
Margin settings are applicable to the entire document and need to be configured in the preamble.

\[\text{noteragged=(opt)}\]

The package option noteragged controls the text alignment within the margins for the mandatory argument of \texttt{\textbackslash note}. It offers the following values: \texttt{left}, \texttt{right}, \texttt{center}, \texttt{justify} or \texttt{twoside}. The default value is \texttt{noteragged=left}, resulting in right-aligned text, which is common for text in the left margin. When \texttt{noteragged=right}, the text is left-aligned. Using \texttt{justify} makes the text justified, aligning with \LaTeX's default behavior for marginal notes. Finally \texttt{noteragged=twoside} aligns text to the left on odd pages and to the right on even pages in a two-sided document. It has no effect otherwise (the default \texttt{noteragged=left} is used and a warning message appears in the terminal).

When \texttt{display} is not set to \texttt{notes}, the \texttt{noteragged} option has no impact, as it specifically applies to text within the mandatory argument of \texttt{\textbackslash note}.

### 5.5 The \texttt{\textbackslash totalpoints} command

\texttt{\textbackslash totalpoints}\{\langle num\rangle\}

The \texttt{\textbackslash totalpoints}\{\langle num\rangle\} macro serves as a replacement for \texttt{\textbackslash points} when using a comprehensive marking scheme. When the scale isn’t visible, it functions similarly to \texttt{\textbackslash points}, and when the scale is shown, it’s akin to \texttt{\textbackslash totalex}. For instance, in exercise 15, we could have used \texttt{\textbackslash totalpoints} instead of \texttt{\textbackslash totalex}. In this case, when the detailed marking scheme is not displayed, the total points would be presented similarly to exercise 5.1 rather than being absent.

### 5.6 Marking scheme consistency checking

\[\text{checkpts=(bool)}\]

The marking scheme can be checked out\(^7\) using the key-val option \texttt{checkpts=true}; the default value is \texttt{false}.

For each exercise, the cumulative points allocated to each question (via \texttt{\textbackslash pts}, \texttt{\textbackslash note*} or \texttt{\textbackslash note[ ]} are compared to the exercise’s total specified in \texttt{\textbackslash points}, \texttt{\textbackslash totalex} or \texttt{\textbackslash totalpoints}. A warning message will be displayed in the shell to indicate whether the scale is valid for the exercise or not. For example:

\texttt{Package exesheet warning: Exercise 3: Sum of points is 4.5pt instead of 5pt.}

Both comma notation (e.g. 4,5) and decimal point format (e.g. 4.5) may be accepted, depending on your chosen language. The validation occurs at the beginning of the subsequent exercise. No warning messages will be presented at this level if no points are specified for the questions.

\texttt{\textbackslash totalsheet}\{\langle total\rangle\}

At the end of the sheet, the last exercise is checked, followed by a global examination of the entire sheet. This last task requires knowledge of the total points for the sheet. To achieve this, use the \texttt{\textbackslash totalsheet\{\langle total\rangle\}} macro in the preamble; otherwise, a warning message will be displayed. If subtotals have been assigned to exercises, the overall comparison is made between the sum of these subtotals and the total points recorded using the \texttt{\textbackslash totalsheet} macro. If not, the evaluation encompasses the sum of points for each individual question. A subsequent warning message indicates the outcome of this last verification. Finally, a message indicates whether all scale controls have been successfully passed or not.

\(^7\)Thanks to Denis Bitouzé for his suggestion about this feature.
6 Options

6.1 Summary of available options

Here we provide a summary table of the available options. Details on their usage can be found in the respective sections. The default value is displayed in bold.

<table>
<thead>
<tr>
<th>Key</th>
<th>Possible values</th>
<th>See section</th>
</tr>
</thead>
<tbody>
<tr>
<td>exetoc</td>
<td>true, false</td>
<td>2.4</td>
</tr>
<tr>
<td>setlist</td>
<td>true, false</td>
<td>3.1</td>
</tr>
<tr>
<td>output</td>
<td>questions, answers, both</td>
<td>4.1</td>
</tr>
<tr>
<td>answerspace</td>
<td>true, false</td>
<td>4.3</td>
</tr>
<tr>
<td>display</td>
<td>none, pts, notes</td>
<td>5.2, 5.4</td>
</tr>
<tr>
<td>marginpos</td>
<td>left (inner), right (outer)</td>
<td>5.2, 5.4</td>
</tr>
<tr>
<td>marginwidth</td>
<td>standard, expand, unset</td>
<td>5.4</td>
</tr>
<tr>
<td>noteragged</td>
<td>left, right, center, justify, twoside</td>
<td>5.4</td>
</tr>
<tr>
<td>checkpts</td>
<td>true, false</td>
<td>5.6</td>
</tr>
<tr>
<td>correct</td>
<td>true, false, conditional</td>
<td>see below</td>
</tr>
</tbody>
</table>

When an invalid key is provided, an error is generated. However, an unrecognized value only triggers a warning message:

Value ... is not supported by ... option on input line ...

For each option, you can set them through the class or package invocation, e.g. \usepackage[output=answers,display=notes,noteragged=right]{exesheet}

You can also use the \exesheetset{list of ⟨key⟩=⟨value⟩} command. Note that some options, output, answerspace, display, and noteragged, can be changed dynamically, even within the document, while the others are applicable in the preamble exclusively. Dynamic options are processed with each call, whereas the others are processed once, at \begin{document}.

A special option, correct, can be employed when using the exesheet class or in conjunction with the schooldocs package. This option adds “Correct version” (or its translation) to the document title and headers. Possible values are: true, false (by default) or conditional. Using correct=conditional, it behaves as true when answers are displayed and false when they’re not.

6.2 Alternative (deprecated) commands

Prior to version 2.0, we used specialized commands to configure output and display options. Thanks to a suggestion from Maxime Chupin and Denis Bitouzé, we have now implemented key=value options. Although the latter is more user-friendly, the older commands are still supported for compatibility reasons and are outlined here. While these commands will trigger a warning message, they remain functional. However, the previous options nosetlist and notoc are no longer supported.

The command \questionsonly is equivalent to setting output=questions and \answersonly means output=answers.

The commands \displaypts and \displaypoints are equivalent to setting display=pts; \displaynotes means display=notes, and \displaynotesright corresponds to display=notes and marginpos=right.
7 Implementation

7.1 Options and required packages

The `exesheet` class is build upon the `article` class and transfers all its unknown options to it. The use of `\ProcessKeyvalOptions*` is unnecessary within the class as it will be managed by the package.

```latex
\begin{verbatim}
\class{exesheet}{\RequirePackage{kvoptions}
\DeclareBoolOption[true]{exetoc}
\DeclareBoolOption[true]{setlist}
\DeclareStringOption[both]{output}
\DeclareStringOption[none]{display}
\DeclareBoolOption[false]{answerspace}
\DeclareStringOption[left]{marginpos}
\DeclareStringOption[expand]{marginwidth}
\DeclareStringOption[left]{noteragged}
\DeclareBoolOption[false]{checkpts}
\DeclareStringOption[false]{correct}
\ProcessOptions \relax
\LoadClass{article}
\RequirePackage{exesheet}
\RequirePackage{schooldocs}
\end{verbatim}
```

Options are defined using the `kvoptions` package. String options are managed through distinct processing macros that are implemented in their respective sections. For options whose effects cannot be dynamically altered and must be configured in the preamble, they are processed once, at `\begin{document}`. The other options are executed when this package is loaded (at the end of the package, as `\exs@process...` commands are not recognized at the outset).

A distinct case is to mention with `setlist` when utilized in conjunction with `babel-french`. In this instance, this option is processed immediately and subsequently disabled (further clarification follows below).

```latex
\begin{verbatim}
\class{exesheet}{\RequirePackage{kvoptions}
\DeclareBoolOption[true]{exetoc}
\DeclareBoolOption[true]{setlist}
\DeclareStringOption[both]{output}
\DeclareStringOption[none]{display}
\DeclareBoolOption[false]{answerspace}
\DeclareStringOption[left]{marginpos}
\DeclareStringOption[expand]{marginwidth}
\DeclareStringOption[left]{noteragged}
\DeclareBoolOption[false]{checkpts}
\DeclareStringOption[false]{correct}
\end{verbatim}
```

Options are defined using the `kvoptions` package. String options are managed through distinct processing macros that are implemented in their respective sections. For options whose effects cannot be dynamically altered and must be configured in the preamble, they are processed once, at `\begin{document}`. The other options are executed when this package is loaded (at the end of the package, as `\exs@process...` commands are not recognized at the outset).

A distinct case is to mention with `setlist` when utilized in conjunction with `babel-french`. In this instance, this option is processed immediately and subsequently disabled (further clarification follows below).
\exs@process@output
\exs@process@display
\exs@process@noteragged
} % answerspace do not need a special process macro
\AtEndOfPackage{\exs@process@dynoptions}
\AtBeginDocument{
  \newif\ifexesheet@multicol
  \@ifpackageloaded{multicol}{
    \exesheet@multicoltrue}{\exesheet@multicolfalse}
  % configuring the rule color within answers environments
  \exs@process@setlist
  \exs@process@marginpos
  \exs@process@marginwidth
  \exs@process@checkpts
  \exs@process@correct
  \DisableKeyvalOption[action=warning,package=exesheet]{exesheet}{setlist}
  \DisableKeyvalOption[action=warning,package=exesheet]{exesheet}{marginpos}
  \DisableKeyvalOption[action=warning,package=exesheet]{exesheet}{marginwidth}
  \DisableKeyvalOption[action=warning,package=exesheet]{exesheet}{checkpts}
  \DisableKeyvalOption[action=warning,package=exesheet]{exesheet}{correct}
}
\def\exesheetset#1{\setkeys{exesheet}{#1}\exs@process@dynoptions}

Now, we load several packages. If the geometry package is already loaded, it
will not be reloaded to prevent an option clash. The shortlabel option in the
enumerate package allows the use of labels similar to the enumerate package such as
i., a), A., and so on. The mparhack package (by Tom Sgouros and Stefan Ulrich)
is loaded exclusively for documents in twoside mode.
\RequirePackage{ifthen}
\@ifpackageloaded{geometry}{\RequirePackage{geometry}}
\RequirePackage{xcolor}
\RequirePackage[shortlabels]{enumerate}
\RequirePackage{tasks}
\RequirePackage{versions}
\RequirePackage{fancybox}
\RequirePackage{translations}
\RequirePackage{ragged2e}
\ifthenelse{\boolean{@twoside}}{\RequirePackage{mparhack}}{}

7.2 Internationalization
Here we define keywords along with their translations in French, German, Spanish
Italian, Portuguese. We achieve this using macros from the translations package
by Clemens Niederberger. This package automatically detects the language being used, as loaded by babel or polyglossia.

Accented characters cannot be utilized here, as they might not be recognized if exesheet is loaded before any other package (typically when it is loaded as a class). As a workaround, we rely on basic \LaTeX{} control sequences to generate them.

\begin{verbatim}
\DeclareTranslationFallback{exesheet-exercise}{Exercise}
\DeclareTranslationFallback{exesheet-subpart}{Part}
\DeclareTranslationFallback{exesheet-annex}{Annex}
\DeclareTranslationFallback{exesheet-ex}{Ex}
\DeclareTranslationFallback{exesheet-points}{points}
\DeclareTranslationFallback{exesheet-point}{point}
\DeclareTranslationFallback{exesheet-correction}{Correction}
\DeclareTranslationFallback{exesheet-annex}{Annex}
\DeclareTranslationFallback{exesheet-ex}{Ex}
\DeclareTranslationFallback{exesheet-points}{points}
\DeclareTranslationFallback{exesheet-point}{point}
\end{verbatim}

\begin{verbatim}
\DeclareTranslation{English}{exesheet-exercise}{Exercise}
\DeclareTranslation{English}{exesheet-subpart}{Part}
\DeclareTranslation{English}{exesheet-annex}{Annex}
\DeclareTranslation{English}{exesheet-ex}{Ex}
\DeclareTranslation{English}{exesheet-points}{points}
\DeclareTranslation{English}{exesheet-point}{point}
\end{verbatim}

\begin{verbatim}
\DeclareTranslation{French}{exesheet-exercise}{Exercice}
\DeclareTranslation{French}{exesheet-subpart}{Partie}
\DeclareTranslation{French}{exesheet-annex}{Annexe}
\DeclareTranslation{French}{exesheet-ex}{Ex}
\DeclareTranslation{French}{exesheet-points}{points}
\DeclareTranslation{French}{exesheet-point}{point}
\end{verbatim}

\begin{verbatim}
\DeclareTranslation{German}{exesheet-exercise}{"Ubung}
\DeclareTranslation{German}{exesheet-subpart}{Teil}
\DeclareTranslation{German}{exesheet-annex}{Anhang}
\DeclareTranslation{German}{exesheet-ex}{"Ub}
\DeclareTranslation{German}{exesheet-points}{Punkte}
\DeclareTranslation{German}{exesheet-point}{Punkt}
\end{verbatim}

\begin{verbatim}
\DeclareTranslation{Spanish}{exesheet-exercise}{Ejercicio}
\DeclareTranslation{Spanish}{exesheet-subpart}{Parte}
\DeclareTranslation{Spanish}{exesheet-annex}{Anexo}
\DeclareTranslation{Spanish}{exesheet-ex}{Ej}
\DeclareTranslation{Spanish}{exesheet-points}{puntos}
\DeclareTranslation{Spanish}{exesheet-point}{punto}
\end{verbatim}
7.3 Titles

The `exercise` counter assigns numbers to exercises throughout the entire document, regardless of sections. To reset the counter manually, simply use \`\setcounter{exercise}{0}`. For an automatic reset at each new section, include the following code in the preamble:

\`\makeatletter \@addtoreset{exercise}{section} \makeatother`.

The parts counter (`subpart`) depends on the `exercise` counter and is reset with each new exercise.

The commands \`\labelexercisestyle` and \`\labelsubpartstyle` are initially empty, but they allow you to customize the styling. For example:

\`\renewcommand\labelexercisestyle{\sffamily}\`

The \`\execheck` macro, responsible for verifying the marking scheme, will be defined in section 7.6.

By default, the table of contents includes both exercises and parts titles, as controlled by the boolean \`\ifexesheet@exetoc` and \`\ifexesheet@exetoc`. To only display exercise titles in the table of contents while omitting parts, include the following code in the preamble:

\`\setcounter{tocdepth}{2}`.
\newcommand{\labelexercise}{\exercisename\space \theexercise}
\newcommand{\labelexercisestyle}{
\ifexesheet@checkpts \exe@check{\labelexercise} \fi
% curiously, \exe@check must be performed before \refstepcounter
\refstepcounter{exercise}
\subsection*{\labelexercisestyle\labelexercise\enskip #1}
\ifexesheet@exetoc
\addcontentsline{toc}{subsection}{\labelexercise}
\fi
}
\newcommand*{\@@exercise}[2][1][]{
\ifexesheet@checkpts \exe@check{#2} \fi
\subsection*{\labelexercisestyle #2\enskip #1}
\setcounter{subpart}{0} % resets the parts counter
\ifexesheet@exetoc
\addcontentsline{toc}{subsection}{#2}
\fi
}
\newcommand{\exercise}[0][\@exercise]{\@ifstar{\@@exercise}{\@exercise}}
\newcommand{\subpart}{\@ifstar{\@@subpart}{\@subpart}}
\newcommand{\annex}{\@ifstar{\@@annex}{\@annex}}
7.4 Enumerations and lists

The \texttt{\setlist} command is part of the \texttt{enumitem} package (\texttt{\setenumerate} is deprecated). By default, \texttt{itemsep=1ex} is set for first-level lists, and \texttt{leftmargin=1.5em} is used to align labels with the start of lines.

\begin{verbatim}
\newenvironment{enumerate}[1][]{\setlist[enumerate]{font=\bfseries}}
\setlist[enumerate,1]{topsep=1.5ex plus 1ex minus 1ex,leftmargin=1.5em,
itemsep=3ex plus 1ex minus 1ex,topsep=3ex plus 1ex minus 1ex}
\setlist[enumerate,3]{noitemsep,nolistsep}
\setlist[itemize]{noitemsep,nolistsep}
\begin{enumerate}[#1]}
\end{enumerate}
\end{verbatim}

When using the \texttt{babel} package with the \texttt{french} option, \texttt{itemize} lists are altered to use the same dash label for each list level. These modifications are undone here to revert to the default \LaTeX\ \texttt{itemize} lists, including labels and spaces. We have created the \texttt{\standardfrenchlists} command, which should be invoked within the \texttt{\AtBeginDocument} command, depending on whether \texttt{exesheet} is loaded before or after \texttt{babel}.

\begin{verbatim}
\newcommand\standardfrenchlists{% must be executed at begin document
\ifexesheet\@setlist
\standardfrenchlists
\setlist[enumerate,1]{topsep=1.5ex plus 1ex minus 1ex,leftmargin=1.5em}
\fi
\end{verbatim}
The \texttt{NewTasks} command is part of the \texttt{tasks} package. It enables the definition of the environments \texttt{tablenum1}, \texttt{tablenuma} and \texttt{tablitem}. Horizontal spacing is adjusted to ensure proper alignment with items in other \texttt{enumerate} (or \texttt{itemize}) environments.

\begin{verbatim}
\ifx\setlist\x@process@setlist
\settasks[label-format=\bfseries]
\NewTasksEnvironment[label=\labelenumone, column-sep=1em, label-align=right, item-indent=1.5em, label-width=1em, label-offset=0.5em, after-item-skip=0.5em plus 0.5ex minus 0.5ex]{tablenum1}\item \\
\NewTasksEnvironment[label=\labelenuma, column-sep=1em, label-align=right, item-indent=2.15em, label-width=1.6em, label-offset=0.5em, after-item-skip=0.5em plus 0.5ex minus 0.5ex]{tablenuma}\item \\
\else
\NewTasksEnvironment[label=\labelenumone, column-sep=1em, label-align=right, label-width=1em, label-offset=0.5em, after-item-skip=0.5em plus 0.5ex minus 0.5ex]{tablenum1}\item \\
\NewTasksEnvironment[label=\labelenuma, column-sep=1em, label-align=right, item-indent=2.15em, label-width=1.6em, label-offset=0.5em, after-item-skip=0.5em plus 0.5ex minus 0.5ex]{tablenuma}\item \\
\fi
\end{verbatim}

\begin{verbatim}
\NewTasksEnvironment[label=\labelitemi, label-align=right, item-indent=2.5em, label-offset=0.5em, after-item-skip=0.5em plus 0.5ex minus 0.5ex]{tablitem}\item \\
\end{verbatim}

The starred environments \texttt{tablenuma*} and \texttt{tablitem*} are designed to be employed within an \texttt{enumerate} environment, precisely at the outset of an \texttt{item}, in order to achieve correct horizontal alignment. The length of \texttt{\baselineskip} has been tested with various font families and sizes. The alignment is generally good.

\begin{verbatim}
\newenvironment{tablenuma}{% \\
\vspace{-1.667\baselineskip}
\begin{tablenuma} \item \\
\end{tablenuma}}{\end{verbatim}

\begin{verbatim}
\newenvironment{tablitem}{% \\
\vspace{-1.667\baselineskip}
\begin{tablitem} \item \\
\end{tablitem}}{\end{verbatim}

\texttt{colsenum}

\begin{verbatim}
\newenvironment{colsenum}[2][]{% \\
\setlength{\multicolsep}{2ex} \\
\raggedcolumns % default is \flushcolumns \\
\begin{multicols}{#2} % #2 = number of columns \\
\begin{enumerate}[#1] % #1 = options of enumerate \\
\end{enumerate} \\
\end{multicols}}{\end{verbatim}

22


7.5 Questions and answers

\exs@process@output The booleans \texttt{exesheet@questions} and \texttt{exesheet@answers} governs the visibility of their corresponding environments. These booleans are configured through the \texttt{output} key option within the \texttt{exs@process@output} macro.

\newboolean{exesheet@questions}\setboolean{exesheet@questions}{true}
\newboolean{exesheet@answers}\setboolean{exesheet@answers}{true}
\def\exs@process@output{
  \ifthenelse{\equal{\exesheet@output}{questions}}{
    \setboolean{exesheet@questions}{true}
    \setboolean{exesheet@answers}{false}
  }{% else if
    \ifthenelse{\equal{\exesheet@output}{answers}}{
      \setboolean{exesheet@questions}{false}
      \setboolean{exesheet@answers}{true}
    }{% else
      \setboolean{exesheet@questions}{false}
      \setboolean{exesheet@answers}{false}
    }
  }
}
We utilize the `versions` package developed by Uwe Lück, which introduces the macros \comment and \endcomment. These macros facilitate conditional displays, a technique also employed in the `verbatim` and `version` packages. Additionally, the notable `codesection` package offers the capability to enclose optional code between \BeginCodeSection{(skip)} and \EndCodeSection{(skip)} macros, both in the text body and the preamble. However, these macros cannot be used within an environment as we have done here with \comment and \endcomment. Several of our tests use the \LaTeX syntax \ifthenelse{\bolean{...}} since \comment and \endcomment can sometimes interfere with the \LaTeX structure \if \else \fi.

The two counters `exe@ini` and `subpart@ini` are employed in the subsequent `\set@toclevel` macro.

The internal macro `\set@toclevel` calculates the title level (counter `toc@level`) to ensure correct typesetting of “Correction” at the start of an `answers` environment, when `questions` and `answers` are displayed together. It involves comparing the `exercise` and `subpart` counters with their values at the time of the `questions` environment call. The `@enumdepth` counter indicates the current `enumerate` list level (with 0 indicating outside of any list). The optional parameter of the `answers` environment permits the explicit specification of this title level.
The internal macro \texttt{\textbackslash typeset@correctionname}, displays the term “Correction” at the appropriate level.

\definecolor{correctioncolor}{rgb}{0,0.2,0.6} % kind of dark blue
\newcommand{\correctionstyle}{\color{correctioncolor}}

\newcommand{\typeset@correctionname}{
\ifthenelse{\value{@toclevel} = 1}{
\section*{\correctionstyle\correctionname}
\ifexesheet@exetoc
\addcontentsline{toc}{section}{\correctionname}
\fi
\setcounter{exercise}{0}
}{% else if
\ifthenelse{\value{@toclevel} = 2}{%
\subsection*{\correctionstyle\correctionname}
\ifexesheet@exetoc
\addcontentsline{toc}{subsection}{\correctionname}
\fi
\setcounter{subpart}{0}
}{% else
\par\textbf{\correctionstyle\correctionname}\par
}}}

Then we proceed to define the \texttt{answers} environment.

\newenvironment{answers}{% #1 is the optional level
\ifthenelse{\boolean{exesheet@answers}}{% #1 is the optional level
\ifthenelse{\boolean{exesheet@questions}}{
\set@toclevel[#1]
\typeset@correctionname
\correctionstyle
\ifexesheet@multicol
\renewcommand{\columnseprulecolor}{\color{correctioncolor}}
\fi
\begin{comment}
\end{comment}
}{% endcomment}
}{% endcomment
}{% endcomment
}{% endcomment
}{% endcomment
}{% endcomment
}
In the \texttt{answers} environment, when placing \texttt{\correctionstyle before \thesubsubsection} (as in the case of \texttt{\typeset@correctionname}), the preceding vertical space may become too wide.

\texttt{\newcommand{\question}{[1]\{\ifexesheet@questions #1\fi}}

\texttt{\newcommand{\answer}{[1]{\%}}}

\texttt{\ifsfs @answers%}
\texttt{\ifsfs @questions \{\correctionstyle #1\} \else #1 \fi}
\texttt{\fi}

The \texttt{\answerspace} macro was suggested by Maxime Chupin to allow students space for writing their answers on the provided paper.

\texttt{\newcommand{\answerspace}{[1]{\ifexesheet@answerspace \par \vspace{#1} \fi}}}

\subsection{Marking scheme options processing}

The options \texttt{display}, \texttt{marginpos}, \texttt{marginwidth} and \texttt{noteragged} are handled using the following internal commands.

The \texttt{display} key option determines the value of the two booleans \texttt{exesheet@pts} and \texttt{exesheet@notes}. The \texttt{exesheet@pts} boolean controls the display of the content of \texttt{pts} and optional arguments of \texttt{note}, while the \texttt{exesheet@notes} boolean controls mandatory arguments of \texttt{note}.

\texttt{\newboolean{exesheet@pts}}
\texttt{\newboolean{exesheet@notes}}

\texttt{\def{\exs@process@display}{\ifthenelse{\equal{\exesheet@display}{pts}}{\setboolean{exesheet@pts}{true}}{\setboolean{exesheet@notes}{false}}\%% else if
\ifthenelse{\equal{\exesheet@display}{notes}}{\setboolean{exesheet@pts}{true}}{\setboolean{exesheet@notes}{true}}\%% else if
\ifthenelse{\equal{\exesheet@display}{none}}{\setboolean{exesheet@pts}{false}}{\setboolean{exesheet@notes}{false}}\%% else
\PackageWarning{exesheet}{Value ‘\exesheet@display’ is not supported by ‘display’ option}))))
The `marginpos` key option takes the values `left` (the default value) or `right` (or `inner` and `outer`). In practice, `inner` is equivalent to `left`, but in two-sided mode, the values `left` or `right` are converted to `outer` (which is then the default value for two-sided mode).

\newboolean{exesheet@leftmargin}
\def\exs@process@marginpos{\ifthenelse{\equal{\exesheet@marginpos}{left}}{\if@twoside% \PackageWarningNoLine{exesheet}{The default ‘marginpos’ option for two-sided documents is ‘outer’.\MessageBreak To change the side, use ‘inner’}%\def\exesheet@marginpos{outer} \setboolean{exesheet@leftmargin}{false} \normalmarginpar \else% default \setboolean{exesheet@leftmargin}{true} \reversemarginpar \fi}{% else if \ifthenelse{\equal{\exesheet@marginpos}{right}}{\if@twoside% \PackageWarningNoLine{exesheet}{The default ‘marginpos’ option for two-sided documents is ‘outer’.\MessageBreak To change the side, use ‘inner’}%\def\exesheet@marginpos{outer} \setboolean{exesheet@leftmargin}{false} \else% default \setboolean{exesheet@leftmargin}{true} \reversemarginpar \fi}{% else \PackageWarningNoLine{exesheet}{The value ‘\exesheet@marginpos’ is not supported by the ‘marginpos’ option}}}\}}

The `marginwidth` option adjusts the ratio between left and right margins based on what needs to be displayed in the margin (points only or full notes)\(^8\).

\(^8\)To ensure the accurate effect on the margin ratio, this option is processed at the beginning of the document, after other commands that could potentially alter the page geometry.
When \texttt{display=notes}, the additional length of \texttt{1in} corresponds to the default free space to the left of \texttt{oddsidemargin}.

The macros \texttt{standardmarginwidthfactor} and \texttt{largemarginwidthfactor} represent the ratios between the total margin width and \texttt{marginparwidth}.

\begin{verbatim}
\def\standardmarginwidthfactor{0.6}
\def\largemarginwidthfactor{0.8}
\newcommand*{\leftnotemarginwidth}[1]{
  \setlength{\marginparwidth}{\oddsidemargin}
  \addtolength{\marginparwidth}{1in}
  \addtolength{\marginparwidth}{-\marginparsep}
  \setlength{\marginparwidth}{#1\marginparwidth}
}
\newcommand*{\rightnotemarginwidth}[1]{
  \setlength{\marginparwidth}{\paperwidth}
  \addtolength{\marginparwidth}{-\textwidth}
  \addtolength{\marginparwidth}{-\oddsidemargin}
  \addtolength{\marginparwidth}{-\marginparsep}
  \addtolength{\marginparwidth}{-1in}
  \setlength{\marginparwidth}{#1\marginparwidth}
}
\def\exesheet@smallmargins{
  \geometry{hmarginratio=1:1}
  \leftnotemarginwidth{\standardmarginwidthfactor}
}
\def\exesheet@standardmargins{
  \ifexesheet@leftmargin
    \geometry{hmarginratio=3:2}
    \leftnotemarginwidth{\standardmarginwidthfactor}
  \else
    \geometry{hmarginratio=2:3}
    \rightnotemarginwidth{\standardmarginwidthfactor}
  \fi
}
\def\exesheet@largemargins{
  \ifexesheet@leftmargin
    \geometry{hmarginratio=3:1}
    \leftnotemarginwidth{\largemarginwidthfactor}
  \else
    \geometry{hmarginratio=1:3}
    \rightnotemarginwidth{\largemarginwidthfactor}
  \fi
}
\def\exs@process@marginwidth{
  \ifthenelse{\equal{\exesheet@marginwidth}{standard}}{
    \ifthenelse{\equal{\exesheet@display}{none}}{
      \if@twoside
        \exesheet@standardmargins
      \else
        \exesheet@smallmargins
      \fi
    }{
      \else
        \exesheet@standardmargins
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
      \fi
    }{
For a two-sided document, the geometry package does not correctly set the width of the margin paragraph by default; it’s too wide. Therefore, we provide an explicit setting here, which is useful when marginwidth=unset. Otherwise, the setting is handled by the marginwidth key option.

\newcommand{\noteragged}{\noteraggedleft}{\noteraggedright}

\edef{\exs@process@noteragged}{\ifthenelse{\equal{\exesheet@noteragged}{left}}{
  \if@twoside
    \renewcommand{\noteraggedleft}{\RaggedLeft}
    \renewcommand{\noteraggedright}{\RaggedLeft}
  \else
    \noteraggedleft{\Centering}
    \noteraggedright{\justifying}
  \fi
}{
  \PackageWarningNoLine{exesheet}{The value \verbatim{\exesheet@marginwidth}
is not supported by the \verbatim{marginwidth} option}
}}

The noteragged option can take one of the following values: left, right, center, justify or twoside. When working with a two-sided document, marginpar can be used with an optional parameter to distinguish left from right contents. In this context, we employ \noteraggedleft and \noteraggedright instead of \noteragged. The ragged2e package by Martin Schröder offers the commands \RaggedLeft, \RaggedRight, \Centering, and \justifying. These commands yield better results compared to the standard \raggedleft, \raggedright and \centering commands. Margin paragraphs are justified by default in \LaTeX.
The scale control option relies on calculations with lengths, which need to have a global scope. To achieve this, we first define the macros \texttt{\gsetlength} and \texttt{\gaddtolength}. These macros include \% symbols at the end of lines to avoid expanded blank spaces.

For questions, assigned points will be added in \texttt{\sum@pts}, while for exercises, points accumulate in \texttt{\sum@exe}. These lengths are compared against \texttt{\exe@total} and \texttt{\sheet@total}. The \texttt{\exe@check} macro validates the calculations of the previous exercise when triggered by \texttt{\points}, \texttt{\totalexe} or \texttt{\totalpoints} macros. It is also invoked within \texttt{\exs@process@checkpts} at the document’s end for a final check on the last exercise.
\newlength{\sum@pts}
\def\exe@label{none}
\newboolean{scale@valid}
\setboolean{scale@valid}{true}
\gdef\gsetlength#1#2{\% for obtaining global length values
  \begingroup
    \setlength\skip@{#2}\% local assignment to a scratch register
    \global#1=\skip@\% global assignment to #1
  \endgroup \% \skip@ is restored at the end of the group}
\gdef\gaddtolength#1#2{\% percent symbol necessary here!
  \begingroup
    \setlength\skip@{#1}\
    \addtolength\skip@{#2}\
    \global#1=\skip@\
  \endgroup}
\def\exe@check#1{\% do not check, no \pts or first exercise begins
  \ifthenelse{\lengthtest{\sum@pts = 0pt}\or\equal{\exe@label}{none}}{
    \PackageWarningNoLine{exesheet}{\exe@label: The scale of \the\exe@total\space is valid}
  }{
    \PackageWarningNoLine{exesheet}{\exe@label: Sum of points is \the\sum@pts\space instead of \the\exe@total}
    \setboolean{scale@valid}{false}
  }
  \gsetlength{\sum@pts}{0pt}
  \def\exe@label{#1} \% for the upcoming exercise}
\def\exs@process@checkpts{\% do not check, no \pts or first exercise begins
  \ifexesheet{\lengthtest{\sum@pts = 0pt}}{Option checkpts is true, but \string\totalsheet\space is missing in the preamble. \MessageBreak See documentation}
  \PackageWarningNoLine{exesheet}{Option checkpts is true, but \string\totalsheet\space is missing in the preamble. \MessageBreak See documentation}
  \gsetlength{\sum@pts}{0pt}
  \gsetlength{\exe@total}{0pt}
  \AtEndDocument{
    \ifthenelse{\equal{\exe@label}{none}}{
      \PackageWarningNoLine{exesheet}{Sum of points
\begin{verbatim}
663                   is valid: \the\sheet@total
664                   ){
665                       \PackageWarningNoLine{exesheet}{Inconsistent
666                       sum of points:
667                       \the\sum@pts\space instead of \the\sheet@total
668                       \setboolean{scale@valid}{false}
669                   }
670                   }
671                       \exe@check{end}
672                   \ifthenelse{\lengthtest{\sheet@total = \sum@exe}}{
673                       \PackageWarningNoLine{exesheet}{Sum of points
674                       is valid: \the\sheet@total}
675                   }
676                       \PackageWarningNoLine{exesheet}{Inconsistent
677                       sum of points:
678                       \the\sum@exe\space instead of \the\sheet@total}
679                       \setboolean{scale@valid}{false}
680                   }
681                   }
682                   \ifthenelse{\boolean{scale@valid}}{
683                       \PackageWarningNoLine{exesheet}{Scale is valid}
684                   }
685                       \PackageWarningNoLine{exesheet}{INVALID SCALE!
686                       Refer to above}
687                   
688                   \fi
689               \end{verbatim}

7.7 Marginal notes commands

\points
\definecolor{pointscolor}{named}{red}
\newcommand{\pointsstyle}{\small\mdseries\sffamily\color{pointscolor}\fbox}
\newcommand*{\exesheet@points}[1]{\hfill\pointsstyle{#1~\ifthenelse{\lengthtest{#1pt < 2pt}}{\pointname}{\pointsname}}\ifexesheet@checkpts\gaddtolength{\sum@exe}{#1pt}\fi%}
\newcommand*{\points}[1]{\ifthenelse{\boolean{exesheet@questions}}{\exesheet@points{#1}}{}}

To prevent spaces between the \fbox and its inner text, percent symbols are necessary. The test \#1 < 2 doesn’t work with decimal numbers without \lengthtest, but it works with lengths.

\pts
\definecolor{ptscolor}{named}{red}
\newcommand{\ptsstyle}{\footnotesize\centering\sffamily\color{ptscolor} (#1)}
\newcommand*{\ptsmark}[1]{\ifexesheet@checkpts\gaddtolength{\sum@exe}{#1pt}\fi%}
\newcommand*{\points}[1]{\ifthenelse{\boolean{exesheet@questions}}{\exesheet@points{#1}}{}}

32
In the subsequent macros that utilize \marginpar, the presence of percent symbols and \ignorespaces is essential to prevent the occurrence of expanded blank spaces in the text (or the margin), where these macros are incorporated.

\definecolor{markingcolor}{named}{red}
\newcommand*{\markingstyle}[1]{\footnotesize\sffamily\textbf{#1}}
\newlength{\ptsboxlength}
\setlength{\ptsboxlength}{3.1em}
\cornersize{1}
\newcommand*{\totalexe}[1]{\ifexesheet@pts\mbox{}\marginpar{\hspace{0pt}\markingstyle{\ovalbox{\makebox[\ptsboxlength]{\ptsmark{#1}}}}}%
\fi\if\gaddtolength{\sum@pts}{#1pt}\fi\ignorespaces}
\totalsheet
\newcommand*{\totalsheet}[1]{\gsetlength{\sheet@total}{#1pt}}
\note
\note*
The booleans \exesheet@pts and \exesheet@notes control the display of marginal notes. If \exesheet@pts is set to false, \exesheet@notes will be ignored. \noindent is required when using \justifying from the \ragged2e package. Within the \note@marginpar macro, enclosing \markingstyle in double braces helps prevent unintended formatting within the mandatory argument of \note. A vicious error occurs when using an \if \fi structure instead of \ifthenelse inside \note@marginpar (but only if @twoside is true).
The correct option and other (deprecated) commands

\ifs@process@correct

\def\s@process@correct{%
  \ifthenelse{\equal{\exesheet@correct}{false}}{%
    \do nothing
  }{% else
    \ifpackageloaded{schooldocs}{
      \ifthenelse{\equal{\exesheet@correct}{true}}{%
        \correct
      }{% else
        \ifthenelse{\equal{\exesheet@correct}{conditional}}{%
          \ifexesheetanswers \correct \fi
        }{% else
          \fi
        }
      }
    }
  }
}
For the time being, the following macros are kept for compatibility reasons.

\newcommand{\questionsonly}{\PackageWarning{exesheet}{The command \string\questionsonly\space is deprecated; \MessageBreak use the package option ‘output=questions’ instead}\renewcommand\exesheet@output{questions}\exs@process@output}

\newcommand{\answersonly}{\PackageWarning{exesheet}{The command \string\answersonly\space is deprecated; \MessageBreak use the package option ‘output=answers’ instead}\renewcommand\exesheet@output{answers}\exs@process@output}

\newcommand{\displaypts}{\PackageWarning{exesheet}{The command \string\displaypts\space is deprecated; \MessageBreak use the package option ‘display=pts’ instead}\renewcommand\exesheet@display{pts}\exs@process@display}

\newcommand{\displaypoints}{\PackageWarning{exesheet}{The command \string\displaypoints\space is deprecated; \MessageBreak use the package option ‘display=pts’ instead}\renewcommand\exesheet@display{pts}\exs@process@display}

\newcommand*{\displaynotes}{[1]\RaggedLeft}{\% % \renewcommand\noteragged{(#1)} no effect now!}
The environment `tablenum` is deprecated and has been replaced by `tablenum1`. The options `notoc` and `nosetlist` are no longer supported. % \@gobble suppresses the line number here

\PackageInfo{exesheet}{The environment 'tablenum' is deprecated \MessageBreak and has been replaced by 'tablenum1'. \MessageBreak The options 'notoc' and 'nosetlist' \MessageBreak are no longer supported\@gobble} % \@gobble suppresses the line number here

\PackageInfo{exesheet}{The environment 'tablenum' is deprecated \MessageBreak and has been replaced by 'tablenum1'. \MessageBreak The options 'notoc' and 'nosetlist' \MessageBreak are no longer supported\@gobble} % \@gobble suppresses the line number here