

# Handin - manual

v0.0.6

(build 47)

Andreas Strauman

April 3, 2018

If you found any bugs or want new functionality, to contribute, view the commented source, get latest version of this package or get in touch with me, you can do all of that at <https://github.com/Strauman/exerciseBank/>

## Contents

<b>1</b>	<b>Motivation</b>	<b>2</b>
<b>2</b>	<b>Flow/Moderate start</b>	<b>2</b>
2.1	Select . . . . .	3
2.2	Exclude . . . . .	3
2.3	Multiple . . . . .	4
2.4	Mixnmatch . . . . .	4
2.5	Solutions . . . . .	4
2.6	Using <code>\At</code> . . . . .	5
2.6.1	Making your own front page for a set . . . . .	5
2.6.2	Printing the file name beneath the problem header . . . . .	5
<b>3</b>	<b>Reference</b>	<b>6</b>
3.1	Environments . . . . .	6
3.2	Internationalization . . . . .	6
3.3	Triggers . . . . .	7
3.4	General reference . . . . .	7
3.5	Making sets . . . . .	8
3.6	Counters . . . . .	9
<b>4</b>	<b>Changelog</b>	<b>10</b>

# 1 Motivation

Exercises are saved as separate files containing part problems. These files can be used to make sets, and you can cherry-pick or exclude certain part problems as you see fit. This makes it easier to maintain and keep your exercises flexible as the syllabus changes.

## 2 Flow/Moderate start

I suspect that working with this package will break your current flow. So let's go through it.

This package assumes you put all of your exercises within the folder named **exercises** (you can change the default folder using `\setExercisesDirP.7`)

exercises/myexercise.tex

```
\begin{intro}
  This introduces our problem
\end{intro}
\begin{problem}
  This is a partproblem 1,
  and will be hidden (just wait, you'll see)
\end{problem}
\begin{problem}
  This is a partproblem 2.
  This will not be hidden, but become part problem a!
\end{problem}
```

Let's build all of them first. In the main file, (the one where you include this package):

main.tex

```
\documentclass{article}
\usepackage{exercisebank}
\makeset{myExerciseSet}{myexercise}
\begin{document}
  \buildset{myExerciseSet}
\end{document}
```

This builds the entire set, and adds Problem header and partproblem counters ( (1a) and (1b) ) by default.

## 2.1 Select

Now, let's build only the second problem.

main.tex

```
\documentclass{article}
\usepackage{exercisebank}
\makeset{myExerciseSet}{\select{myexercise}{2}}
\begin{document}
  \buildset{myExerciseSet}
\end{document}
```

This should only build the intro and the one exercise you `\select`ed!

Now, say you want to hide the intro. Well all you have to do in this case is make the package treat the intro as a problem in regards to what is `\select`ed. Just add the optional argument `[<intro>]` to `\make`. That is switch

```
\makeset{myExerciseSet}{\select{myexercise}{2}}
```

with

```
\makeset[intro]{myExerciseSet}{\select{myexercise}{3}}
```

Notice that there are 3 ‘partproblems’ now since we have to count the intro!

## 2.2 Exclude

But what if you have an exercise with 12 partproblems, and you only want to exclude the 7th partproblem? Well, then `\Exclude` is here to rescue the day for you.

```
\makeset{myExerciseSet}{\exclude{soManyExercises}{7}}
```

Here it's important to note that the `[intro]` argument would not make the intros disappear. If we wanted to only exclude the intro from our previous example file `exercises/myexercise.tex` we would do

```
\makeset[intro]{myExerciseSet}{\exclude{myexercise}{1}}
```

So we're excluding the partproblem 1. But that's the intro when we send the `[intro]` optional argument

## 2.3 Multiple

In `\makeset` you can just separate exercises with commas! Here is an example: Let's say you have two files with exercises. One located in `exercises/circuits/RLC.tex` and one in `exercises/ohm/ohmsGeneralLaw.tex`, and you want to include part-problem 1 through 5 from `RLC.tex` and all of the exercises from `ohmsGeneralLaw.tex`.

```
\makeset{\select{circuits/RLC}{1,...,5}, ohmsGeneralLaw}
```

This will divide it up with problem headers. So that what is in the `RLC.tex`-file will be Problem 1, and `ohmsGeneralLaw.tex` Problem 2.

## 2.4 Mixnmatch

What if you want to make both of them the same exercise? Well, then you pass the `[nohead]` argument to `\makeset`:

```
\makeset[nohead]{\phead, \select{circuits/RLC}{1,...,5}, ohmsGeneralLaw}
```

The `\phead` command makes a problem header. You can pass them as much as you want:

```
\makeset[nohead]{\phead, \select{circuits/RLC}{1,...,5},  
ohmsGeneralLaw, \phead, someOtherExercise, moreExercises}
```

## 2.5 Solutions

The last thing to cover then is solutions. In your exercise files you just use the solution environment

```
\begin{solution}  
Solution goes here  
\end{solution}
```

They are hidden by default, so you would have to use `\DisplaySolutions` in your main file to display them.

## 2.6 Using \At

### 2.6.1 Making your own front page for a set

To create your own front page for each set, you'd use `\At\StartBuildset{\FrontPageContents}`. Anything you send in to `\At\StartBuildset` will be executed before the first problem starts:

```
\At\StartBuildset{
  \input{myFrontpage}
}
```

### 2.6.2 Printing the file name beneath the problem header

Maybe you want to see what files are what problems. One way to do that is hook into the `\At\VeryBeginProblem` and use `\exerciseFile`<sup>→P.8</sup>. This is the command that the headers call, so anything you put in there will be executed right after the headers.

```
\At\VeryBeginProblem{
  \exerciseFile\
}
```

Note that the `\AtBeginProblem` is executed *before* the headers. That covers the basics. Enjoy

! • `\begin{problem}, \end{problem},`  
`\begin{solution}, \end{solution},`  
`\begin{intro}` and `\end{intro}` has to be on their own line without any spaces!

## 3 Reference

### 3.1 Environments

```
\begin{problem}  
  <environment content>  
\end{problem}
```

Inside the exercises folder, you keep your exercises. Inside there you'd use a problem environment to write your partproblems. It might be a little confusing that you're using `\begin{problem}` instead of `\begin{partproblem}` when you're writing a partproblem, but it's less typing.

```
\begin{solution}  
  <environment content>  
\end{solution}
```

Things inside here is only visible if `\DisplaySolutions` are given before `\begin{document}`

! `\end{solution}` has to be on it's own line without any leading spaces!

```
\DisplaySolutions
```

Turns on the solutions, so they are shown.

```
\begin{intro}  
  <environment content>  
\end{intro}
```

Sometimes you'd want to introduce your exercises and tell a little bit about it. Maybe have a figure there also. Those things should go inside this environment. This can be treated as a problem in terms of counting. See `\makeset`<sup>→P.8</sup> for more info.

### 3.2 Internationalization

```
\translateExBank{<Translation key/vals>}
```

This is to translate the text inside the package. As of now the available key/values are

- Problem
- Solution

The Norwegian translation would then be done with

```
\translateExBank{Problem=Oppgave, Solution=Løsning}
```

### 3.3 Triggers

`\Trigger{⟨Any Macro⟩}`

See `\At`

Available triggers:

`\Trigger\BeginPartproblem:`

Triggers before a `partproblem` is inserted

`\Trigger\VeryBeginPartproblem:`

Triggers right after `\BeginPartproblem`. This is so that the user can do stuff before the actual headers start. The `partproblem` headers are invoked by `\At\VeryBeginPartproblem \Trigger\InputExercise:`

Triggers before a file is included

`\Trigger\BeginProblem:`

Triggers before a file is included, but only if problem headers are to be written (no `[nohead]` given)

`\Trigger\EndProblem:`

Triggers right after problem is included if `[nohead]` `\emph{not}` given

`\Trigger\BeginBuildset:`

Triggers right before a set has begun building (not if `\sprite` is used). You might want to put your set-header here

`\Trigger\EndBuildset:`

Triggers when a set has stopped building (not if `\sprite` is used)

### 3.4 General reference

`\ownLineNoSpacesGotIt`

This is to annoy the user enough to get his attention about the requirements of the `problem`<sup>→P.6</sup>, `solution`<sup>→P.6</sup> and `intro`<sup>→P.6</sup> environments.

`\setExercisesDir{⟨directory⟩}`

This is the directory, relative to the file you included the package, where the package should be looking for exercises. Default is `exercises`

This package also includes some extra stuff. For example the `\At` and `\Trigger`

`\At{⟨AnyMacro⟩}`

Here you can send any macro because it isn't evaluated! For example `\At\BeginSomething` is fine and even if `\BeginSomething` is not defined. Also and when using `\Trigger` it just ignores it if it didn't exist. It's pretty similar in function as to `\AtBeginDocument`.

```
\At\BeginSomething{DoSomething}
Which is triggered with
\Trigger\BeginSomething, this evaluates to DoSomething
```

`\isin{⟨haystack⟩}{⟨needle⟩}{⟨True⟩}{⟨False⟩}`

⟨haystack⟩ is a comma separated list of integers

⟨needle⟩ is an integer

Executes ⟨True⟩ if ⟨needle⟩ is found in ⟨haystack⟩ else executes ⟨False⟩

`\exerciseFile`

This is a ‘read-only’ macro that contains the name of the current exercise-File

### 3.5 Making sets

`\makeset[⟨intro,nohead⟩]{⟨filable⟩}`

This command is the one you use to make a set! Later you use `\buildset` to build the sets you make. The ⟨filable⟩ argument is either the name of the file relative to the `\exerciseDir`-path (default its in the root called exercises), or you could use the `\select` or `\exclude` to respectively cherry pick or exclude exercises. (See their docs).

`[⟨intro⟩]` this counts the intro environment as a part problem, so that you can `\select` or `\exclude` the intro

`[⟨nohead⟩]` prevents the builder from adding a problem header. This is handy if you want to create an exercise that is composed of multiple parts. You can use the `\phead` to insert the problem header where you want it

```
\makeset[nohead]{\phead, \select{myexercise}{1,2,3}}
```

`\about{⟨text⟩}`

This contains information about an exercise set. It is intended to be on the top of an exercise, explaining short what the exercise is about. It’s only visible when using `\sprite`

`\sprite[⟨PiP⟩]`

This is a way to visualize all exercises. It takes one optional argument which is how many pages inside one page. Defaults to 4



If `\sprite` is used, it should be the only command in `\begin{document}\end{document}`

`\exclude{⟨exerciseFileName⟩}{⟨Comma separated numbers⟩}`

As you can see in the intro section of the documentation, this is for excluding partproblems To be used in `\makeset`

`\select{⟨exerciseFileName⟩}{⟨Comma separated numbers⟩}`

As you can see in the intro section of the documentation, this is for cherry picking partproblems To be used in `\makeset`



`\setName`

This variable prints out the name of your set that you sent to `\buildset`. The following example prints "Exercise set number 1" and "Exercise set number 2" on the top of each set

```
\At\StartBuildset{
  Exercise set number \setName
}
%... \makesets here ...%
\begin{document}
  \buildset{1}{myexercise}
  \buildset{2}{myexercise}
\end{document}
```

`\pplabel{\langle label \rangle}`

Labels a partproblem. You can reference to it later using `\ppref{\langle label \rangle}`

`\ppref{\langle label \rangle}`

Reference a partproblem created by `\pplabel{\langle label \rangle}`. This prints e.g. 1c)

`\ppref{\langle label \rangle}`

Reference a partproblem created by `\pplabel{\langle label \rangle}`. This prints e.g. 1

### 3.6 Counters

`problemcounter`-counter holds the current problem number and `partproblemcounter`-counter holds the current partproblem *number*.

## 4 Changelog

Version	Date	Change
v0.0.2b11	2018/04/02	Updated documentation syntax.
v0.0.3b38	2018/04/03	<ul style="list-style-type: none"><li>• Updated triggers doc</li><li>• added trigger <code>\VeryBeginPartproblem</code></li></ul>
v0.0.3b40	2018/04/03	<ul style="list-style-type: none"><li>• Added examples that uses the <code>\At</code> command. Making front page and other snacks</li></ul>
v0.0.4b44	2018/04/03	<ul style="list-style-type: none"><li>• Changed design of part problems. (Looks much better now!)</li></ul>
v0.0.5b46	2018/04/03	<ul style="list-style-type: none"><li>• Fixed partproblems and solutions to fit on pages using <code>\filbreak</code>.</li></ul>