PGML Cheatsheet

LaTeX Math

This is for typesetting equations using LaTeX math format. The Typeset MathObjects context is used. The two modes are

- `[` \frac{\sqrt(3x+2)}{2} `]` - Inline mathematics.
- `[``  \frac{\sqrt(3x+2)}{2} ``]` - Display style mathematics

ASCII Math

This is for typesetting equations using "ascii" math format (or more specifically MathObjects format). The Typeset MathObjects context is used. The two modes are

- `[: sqrt(3x+2)/2 :]` - Inline mathematics.
- `[:: sqrt(3x+2)/2 ::]` - Display style mathematics

The options for this mode are:

- `[: sqrt(3x+2)/2 :]*` - Typeset in current context instead.
- `[: sqrt(3x+2)/2 :]**` - Reduce formula before displaying.
- `[: sqrt(3x+2)/2 :]***` - Reduce and typeset in current context instead.

Variable Substitution

This is for substituting a previously defined variable into PGML, inside or outside an existing expression. The format is `[$variable]`. E.G. `[`f(x) = [$f]`]`. It supports the following options:

- `[$var]*` - Do not escape HTML special characters.
- `[$var]**` - Substitute variable and process the result as PGML.
- `[$var]***` - Substitute the variable and process the result as LaTeX.

Answer Blanks

Answer blanks are created using the notation `[_______]`. The width of the answer blank is proportional to the width of the resulting field. The correct answer can be specified with the blank. E.G. `[__________]{"sqrt(x)"}`

- For matricies and vectors you can provide an array of answer blanks instead of a single answer blank by adding a *

  `[___]*{Matrix([1,2],[3,4])}`

- You can specify the length of the blank numerically via `[____]{"3x"}{20}`.
- Create radio button or popup answers by providing a RadioButtons or PopUp object; i.e. `[__]{$popUp}`
- If you provide a string as the correct answer it is turned into a MathObject. You can also provide MathObjects directly or as variables.

  `[______]{Complex(0,1)}` `[______]{$f}`

- Multianswer comparorators can be used by providing the same multianswer to both blanks.
- Comparators can also be provided directly

  `[______]{$f->cmp}` `[______]{num_cmp(pi)}`

Command Substitution

This is for evaluating a string of Perl code and then including the result into PGML. The format is `[@ 1+2 @]`.

```
[@ image(insertGraph($graph), width=>300, height=>300) @]*
```

This mode recognizes the following options:

- `[@ command @]*` - Do not escape HTML special characters.
- `[@ command @]**` - Evaluate command and process the result as PGML.
- `[@ command @]***` - Evaluate command and process the result as LaTeX.

Miscellaneous Blocks

- `*Bold Text*` - The text is bold.
- `_Italic Text_` - The text is italic.
- `[% comments %]` - The text is a comment and is not rendered.
- `[| Verbatim Text |]` - The text is included verbatim.
- ```` ``` print('hello world') ``` ```` - The text is printed as programming code.
- `>> centered <<` - Creates centered text.

Formatting Markdown

- Paragraphs and line breaks. Normal line breaks are ignored and will not break Miscellaneous Blocks.
  - Leave a blank line or end a line with three spaces to create a paragraph break. This will break most Miscellaneous Blocks.
  - End a line with two spaces to create a line break. This will not break Miscellaneous blocks, headings, or right justified text.
- Use backslashes to escape PGML special characters (including backslash). HTML and LaTeX special characters are dealt with automatically.
- Indenting - Use four spaces to indent a paragraph. End a paragraph by leaving a blank line or ending a line with three spaces.
- Lists - The type of number or bullet controls the format of the list but not the actual number. Indent with four spaces to get sub lists.
  - `* list item` - Use a bullet and a space for each item.
  - `1. list item` - Use a number/letter, a period, and a space for each item.
- `## Heading` - Use `#` for headings. You can use up to six hashes for various sizes.
- `>> Right Justified` - Creates a right justified line or paragraph.
- `:   Pre-formatted` - Use a colon and three spaces to create a pre-formatted line.
- `------` - Use three or more dashes to make a rule. You can also specify the width and size of the rule: `====={200px}{5px}`