

MPS115/MPS116 PRESENTATION LAB 9

Today we will put mathematical content on to webpages using *MathJax*.

Note: Using MathJax works more smoothly in some browser than others. If you have problems, try Firefox or Chrome.

1. BUILDING A MATHEMATICAL WEBPAGE

1.1. Getting started. Go to the ‘Course materials’ section of the module webpage, under ‘Computer labs’ find `lab_9_template.html` and, using right-click and ‘Save link as’, download it, saving it as `lab_9.html`. Now open it in Notepad++, or your favourite editor. Set the title and the main `<h1>` heading to “Newton’s laws”.

Download `newton.css` from the module webpage, using right-click and ‘Save link as’, and save it as `lab_9.css`. Feel free to edit the CSS file.

Download the `newton.txt` text file from the module webpage. Copy and paste this text into the `<main>` element of your HTML file. You are now going to turn this plain text into HTML. The text consists of two sections, ‘Newton’s Laws of Motion’ and ‘Newton’s Laws of Gravitation’, put these into `<section>` elements and make the two titles into `<h2>` heading elements.

The document should have the following schematic structure. (I’ve ignored closing tags for brevity and clarity.)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Newton’s Laws
  <body>
    <header>
      <h1>Newton’s Laws
    <main>
      <section>
        <h2>Newton’s Laws of Motion
        (content)
      <section>
        <h2>Newton’s Laws of Gravitation
        (content)
```

Now turn the content into HTML. Start by turning the wikipedia acknowledgement sentences at the beginning of both sections into smallprint, using the `<small>` tag. Turn the term ‘Wikipedia page’ in the introductory sentence into a link to the Wikipedia page for ‘Newton’s Laws of Motion’.

As you work down through the text in this section, sort out each paragraph by adding `<p>` at the beginning and `</p>` at the end.

- (1) Turn the three laws into an *ordered list*. To do this, use the tag `` before the list and `` afterwards. Then, add `` before each list item and `` afterwards.
(Think of this like LaTeX’s `enumerate` environment, with `` similar to `\begin{enumerate}` and `` similar to `\item`.)
- (2) Change the `ol` tags to `ul` tags to create an *unordered list*. Change them back.

```

<table>
  <thead>
    <tr>
      <th>Column 1 heading
      <th>Column 2 heading
    </tr>
  <tbody>
    <tr>
      <td>Row 1 column 1 entry
      <td>Row 1 column 2 entry
    </tr>
    <tr>
      <td>Row 2 column 1 entry
      <td>Row 2 column 2 entry
    </tr>
  </tbody>
</table>

```

FIGURE 1. The schematic structure of an HTML table (closing tags omitted for brevity)

1.2. **MathJax.** Now we will improve the mathematics on the page.

- (3) Find the mathematical symbols in the text and put them in dollar-signs, like you would in LaTeX. (At this stage, you should expect to see dollar-signs on your webpage.)
- (4) Go to the module webpage and look for the MathJax code in the ‘HTML and web design’ section under ‘Extras’. Copy and paste this into the head of your HTML document and refresh your webpage. Any change? If not, ask for help!
- (5) Tidy up the maths using your LaTeX skills to format the equations in a pleasing way, using `\frac` commands and subscript notation where appropriate. Display the equation describing Newton’s second law, $F = \frac{d}{dt}(mv)$, by putting it in `\[. .\]`.

You should have found that the MathJax code in the head of your document allowed you to use standard mathematical LaTeX notation to display equations and expressions.

1.3. **Tables.** Add a section at the bottom called ‘Appendix’ and follow it with the paragraph ‘The table below shows the SI units for the quantities involved in the above discussion’.

By looking at the source code for the course webpage, build a table as below, filling in the rows corresponding to the symbols used in the formulas on the webpage. You should be looking for tags such as `<table>`, `<thead>` (table head), `<tbody>` (table body), `<tr>` (table row), `<th>` (table heading cell) and `<td>` (table data cell). Also, see Figure 1.

Symbol	Quantity	SI unit
F	Force	Newton (N)
\vdots	\vdots	\vdots

If you aren’t happy with the format of the table, use CSS to change it by using the `table`, `th` and `td` selectors. (You could borrow ideas from the CSS file on the course webpage).

2. PROGRAMMING CODE ON WEBPAGES

2.1. Displaying code. We can also get computer code to display well in webpages. You should put it within `<pre>` (which stands for ‘pre-formatted’) and `<code>` elements.

Underneath the statement of Newton’s law of Gravitation put an `<h3>` header “A Python program” and a paragraph of “Here is a Python program to plot the gravitational force between two point masses.” Then add the following.

```
<pre>
    <code>
        (Code goes here)
    </code>
</pre>
```

Download `gravitational_force.py` from the module webpage (right-click to save) and paste in the code of the program in the place indicated above.

For inline code, you can just use `<code>...</code>`. If you don’t like the default style of font, you can change it using the font-family property in CSS (Courier New would be a good choice).

Some characters may not display properly (for example, `<` signs will cause problems). To correct this, convert your code to ‘safe’ HTML characters using an online converter, such as the one in the Extras section of the course webpage. Choose the options to convert line breaks and change spaces to ` `. (Can you work out what ` ` does?)

3. MINI-PROJECT REVISITED

Let’s find out how easy it is to turn a LaTeX file into a webpage.

- (6) Start a new HTML file called `miniproject.html`. Take the LaTeX code from the body of your mini-project LaTeX file and paste it into the body of the HTML document.
- (7) Replace all the LaTeX commands for sections, titles, itemized lists, etc. with their HTML equivalents, adding in `<p>` tags and others as required to get your mini-project working as a webpage.
- (8) You can display your Python code on your webpage using `<code>` tags and you can also link to your Python file so that the user can download it.

4. GOING FURTHER WITH HTML AND CSS

That ends the material on HTML and CSS for this course. There are lots of online tutorials on using HTML and CSS. If you find a really good one, why not share it with the rest of the MPS115 and MPS116 students on the course discussion board?

HOMEWORK

Finish the sheet.

The main homework this week is the peer assessment exercise. See the announcement on Blackboard for details. This is due in on Monday 8th December at 2pm.