HTML, CSS and Tornado

Due date: 11:59pm Friday 2013-06-06 (Week 13)

This assignment is worth 15% of your final assessment.

This third assignment will complete our major project of developing a useful tool for scientific research. In the setup for Assignment 2, we created a series of database tables from data about birds, and sightings of those birds; and then we wrote SQL queries to access that data in the database. In this assignment we will add a Tornado based user interface to allow access to these resources across the web.

This assignment consists of three main questions and an extension question. Each question is worth 25 points. The three main questions will lead you through developing the Tornado interface from scratch. If you complete these successfully you can achieve up to 75% of the available marks for this assignment. The extension question is much more open to see what you can do. The extension will be judged on the sophistication of the extension, the complexity of the code required, and the design/style of that code.

Academic Plagiarism declaration

By submitting this assignment you declare the following:

I declare that I have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure, and except where specifically acknowledged, the work contained in this assignment/project is my own work, and has not been copied from other sources or been previously submitted for award or assessment.

I understand that understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to severe penalties as outlined under Chapter 8 of the University of Sydney By-Law 1999 (as amended). These penalties may be imposed in cases where any significant portion of my submitted work has been copied without proper acknowledgement from other sources, including published works, the internet, existing programs, the work of other students, or work previously submitted for other awards or assessments.

I realise that I may be asked to identify those portions of the work contributed by me and required to demonstrate my knowledge of the relevant material by answering oral questions or by undertaking supplementary work, either written or in the laboratory, in order to arrive at the final assessment mark.

I acknowledge that the School of Information Technologies, in assessing this assignment, may reproduce it entirely, may provide a copy to another member of faculty, and/or communicate a copy of this assignment to a plagiarism checking service or in-house computer program, and that a copy of the assignment may be maintained by the service or the School of IT for the purpose of future plagiarism checking.
Retrieving and sending your files

For this assignment there are no files to download, since the data has already been loaded into the database directly in your work for assignment 2.

To submit your work you should produce a tar file that contains all the various files you wrote for Tornado (such as question1.html question2.html presnt.css etc. You can then submit the tar file via the Challenge site (however, no auto grading is done; the markers will run Tornado and visit your pages by hand, as well as looking at the source). Please keep a copy of your submitted assignment until you receive your marks back.

Accessing your MySQL account

Accessing your database account from within Python (using the MySQLdb module), and from within Tornado, is needed for this assignment; the techniques for this are given in the advanced Tornado notes for Lab 17.
Question 1: Developing a mockup response webpage (25%)

For this question you must develop a static webpage to display data (that might come as the response from a query). Note, this question is just asking for HTML that delivers information in an appropriate way; you do not have to contact the database at runtime (that’s question 3). The page should be accessed as question1.html.

The title and the top level heading of the webpage should indicate that it is the result of searching for birds with body mass in a range. For this question, we will use the particular data that results from question 1 of assignment2, when that is run against the data files you were given for assignment 2.

Your webpage should clearly explain what data is being presented, it must show the lower and upper limits of body mass that determine this search (namely, 2g and 3.3g), and it should show a table of data. The first row of the table should describe the following rows: the first column is a row number, and the second column is the common name. Then each subsequent row should show one bird returned by the query, preceded with a row number that starts at 1 (for the first bird), and increases in each following row.

Thus the table should have the following content and arrangement:

<table>
<thead>
<tr>
<th>rownum</th>
<th>common name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Allen’s Hummingbird</td>
</tr>
<tr>
<td>2</td>
<td>Bee Hummingbird</td>
</tr>
<tr>
<td>3</td>
<td>Black-chinned Hummingbird</td>
</tr>
<tr>
<td>4</td>
<td>Broad-tailed Hummingbird</td>
</tr>
<tr>
<td>5</td>
<td>Calliope Hummingbird</td>
</tr>
<tr>
<td>6</td>
<td>Costa’s Hummingbird</td>
</tr>
<tr>
<td>7</td>
<td>Flammulated Owl</td>
</tr>
<tr>
<td>8</td>
<td>Ruby-throated Hummingbird</td>
</tr>
<tr>
<td>9</td>
<td>Rufous Hummingbird</td>
</tr>
</tbody>
</table>

The assessment of this question includes both the clarity of information presentation, and the correct use of HTML in presenting it.
Question 2: Developing a Cascading Style Sheet (25%)

For this question, you need to revise your static page so that it has a clear separation between the information content and the presentation, and so the presentation is controlled in ways we indicate below. You should offer a new page accessed at question2.html that looks similar to your answer to question 1, but the HTML needs to be written appropriately (e.g. adding extra class or id attributes), and also it should link to a cascading style sheet present.css that you have written.

This style sheet should do the following things:

- set all of the text to helvetica or san serif;
- set the top level headings to 20 point bold and dark grey;
- set tables so that they have a background:
  - no background for the header row
  - shaded in light grey for even rows;
  - shaded in light blue for odd rows;
- set table header text to italic, while the other contents of the table are in normal (roman).
- set the lower and upper limits of the range to bold and bright blue;
- align the tables in the horizontal centre of the page;
Question 3: Performing a dynamic query from the database (25%)

Now it is time to connect to the MySQL database and actually deliver results from a user-controlled request. Use Tornado to provide a form at question3.html, where the user can enter the lower and upper limits for the body mass range they wish to search. There should also be a button to submit the query. When the user presses the submit button, the appropriate SQL query should be run on the database, and then a page should be generated that looks like your mockup from question1 (or its CSS embellishment from question 2). That is, the system needs to display the common names of birds whose body mass is in the given range; each bird should be accompanied by a row number. For full marks in this part, the form should do some basic input-checks, for example, to catch cases where the lower limit is above the upper limit, and also whenever there are no results for the query, then the output should have a statement saying there were no results, instead of showing an empty table. **Warning: if your design is vulnerable to a SQL injection attack, you will lose points.**
Question 4: Extension (25%)

The final stage of the assignment is to implement an extension to the basic system you produced in Questions 3. The extension should add to the site’s informativeness and/or usefulness. You should choose one of the four extension seed ideas listed below. This will form the basis of your extension, but you can take it as far as you like. (If you want to do something quite different, ask the lecturer first.)

The extension will be judged on its usefulness and sophistication, and on the design and style of the code you provide, and also on the sophistication of technical features that you used (but only if they are needed; you don’t get points for doing things in a complex way if they can be done as well by a simpler approach.) In particular, to score well for usefulness, you should make sure that the pages themselves have enough information about the meaning of any data, and also guidance in how to use the system.

The four seed ideas are:

1. Support (generalised variants of) more of the queries from assignment 2. You should arrange that there is a landing page called question4.html from which it is easy to reach the answers to different queries.

2. Use cookies to store session information for users of the website. This would then be used to add extra functionality, e.g. to provide the user with a record of their most recent queries that can be quickly rerun. You should arrange that there is a landing page called question4.html from which a session begins.

3. Deliver a limited amount of data at a time; so for example, if there are 35 birds whose body mass is in the given range, you might initially display the first 10, with a button that leads to displays of the next 10, and then the next 10, and a final page shows the last 5 remaining birds. As the user moves through these pages, the row numbers need to be kept accurate (e.g. the second page should have row numbers 11 to 20). It would be good to have navigation control, so the user can jump to any page of the output at any time. You should arrange that there is a landing page called question4.html which displays the first page of the results.

4. Provide a way to browse the data in the database, and display it in an integrated way. For example, maybe the user can indicate which attributes are of interest, and then select a bird and see all the selected attributes for that bird. You should arrange that there is a landing page called question4.html from which a browsing activity begins.