

# Homework 14

SI 507 Winter 2018

This homework is designed to be completed in discussion section. It really shouldn't take more time than that. Don't let it drag on!

You will take the guest book application that was developed during lecture 25 and add the ability to delete posts. The following instructions will lead you through the process.

## Part 1 (5 points)

Create a new template named "admin.html." It should look almost exactly like "index.html" except that each guestbook entry should show the post ID and also contain a <form> element that contains only the following:

- a submit button (<input type="submit">) that displays an "X"
- a hidden field that contains the ID of the post to be deleted (<input type="hidden" name="id" value="{{ the id of the post }}">)

**Welcome to my guestbook!**

[Add a message](#)

*Good bye*

by Newman on 2018-04-12 12:20:54.251230

id: 1

X

*Hello*

by Mark on 2018-04-12 12:20:44.274475

id: 0

X

The form's action should point to a new route called "/delete". This will be implemented in Part 3.

## Part 2 (5 points)

Add a global variable `next_id` to the model. It should default to 0. If there are previous entries that are loaded from `entries.json` at startup, `next_id` should end up being set to one greater than the maximum ID found in the previous entry data. Whenever a new entry is added (in `add_entry()`) it should include the id as a string in the dictionary (`"id": str(next_id)`) and then increment `next_id`.

## Part 3 (5 points)

Implement the `/delete` route so that it extracts the ID from the post data, passes the ID to a `delete_entry()` function in the model (just a stub for now), and then redirect the user back to the `/admin` page.

## Part 4 (5 points)

Implement `delete_entry()` in the model so that the post with the specified ID is deleted. Make sure that these changes are persisted to `entries.json`.

## Extra Credit

### Extra Credit 1 (2 points)

*Note: EC is only available if all of Parts 1-4 are implemented correctly.*

Deploy your modified app to Heroku

### Extra Credit 2 (2 points)

Change the app so that the admin user can use `/admin` to change the text of the message as well, but not the name of the original poster. The timestamp should be changed to the time when the update was made.

## What to turn in

Parts 1-4 and EC 2: A link to your github repo.

EC 1: Also include a link to your app on Heroku.