

# Assignment: Courses

## Objectives:

- More practice with full-stack creating, reading, and deleting from a database
  - More practice with model validations
  - Learn how to implement one-to-one relationships with Django's ORM
- 

Make a new Django project and application, and complete the functionality of the wireframe below. Require the course name to be more than 5 characters and the description to be more than 15 characters.

### Bonus Features (Optional)

1. Make description a one-to-one relationship with the Course table rather than a column in the course table (*intermediate*).
2. Add an action [comment](#) to each course that takes you to a page where you can add comments about that course and view all comments that have been entered (*advanced*)

← → ✕ 🏠

Add a student to all bootcamp courses

http://localhost/ 🔍

Add a new course

Name:

Description:

Add

Courses

Course Name	Description	Date Added	Actions
How to be a ninja		Dec 1st 2013 5:34PM	<a href="#">remove</a>
How to fly		Nov 1st 2011 11:12AM	<a href="#">remove</a>
How to get more energy in the bootcamp		Oct 1st 2013 11:11AM	<a href="#">remove</a>
How to pair program more effectively		Sep 28th 2011 10:15PM	<a href="#">remove</a>

← → ✕ 🏠

Delete a course

http://localhost/courses/destroy/3 🔍

Are you sure you want to delete the following course?

Name:

How to get more energy in the bootcamp

Description:

Description about this course...

No

Yes! I want to delete this

1. Make a new Django project and app
2. Have the root route render the main wireframe above
3. Create the POST method to add new courses to the database
4. Ensure that the name entered is more than 5 characters, and the description is more than 15 characters
5. Display all the courses in a table beneath the form
6. For each course, have a link to remove that renders a page as shown in the second wireframe
7. If the user selects "No," redirect to the root route. If the user selects "Yes," delete the course and redirect to the root route
8. NINJA BONUS: Make the description a separate class and have the description field of the Course class be a one-to-one relationship with Description
9. NINJA BONUS: For each course, have a link to comment that renders a page with a form to make comments and a list of all comments for that course
10. SENSEI BONUS: Use AJAX for the remove functionality, prompting with a modal and processing the remove accordingly (No--close the modal without deleting, Yes--close the modal and delete the entry)