# Lab: Templating

The following tasks do not have tests in the Judge system. They are for practice.

**Working with Remote Data**

For the solution of some of the following tasks, you will need to use an up-to-date version of the **local REST service**, provided in the lesson's resources archive. You can [read the documentation here](https://github.com/softuni-practice-server/softuni-practice-server).

## Contacts Book

Create a simple page containing some cards with contacts. Each card should have a **name** and a **button "Details"**. Each time the button is pressed, you have to **toggle a div** containing more detailed information about the corresponding contact.

You must create a **template** for the **contact card** and **render** all of them on the main page inside a <div> with **id "contacts"**. You will be provided with the **HTML, CSS** and a **JavaScript** file with contacts’ data. Import the file **contacts.js** as a module.

See the file example.html for an example of how the rendered HTML should look like.

### Example





## My Cookbook – Refactoring

The resources for this task are available in the following GitHub repository:

<https://github.com/viktorpts/js-apps-workshop>

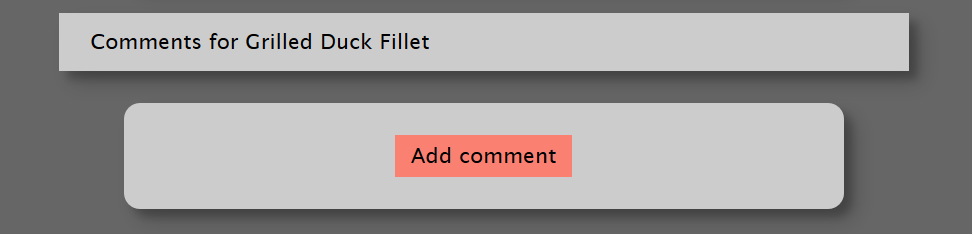
You may check-out the repository or download the files via the green button labeled "Code" in the upper-right corner. Use the files located in **lesson-06/base** to begin the task. Before starting, make sure you have the most recent version of the repository. To see the solution, check the files inside **lesson-06/finished**.

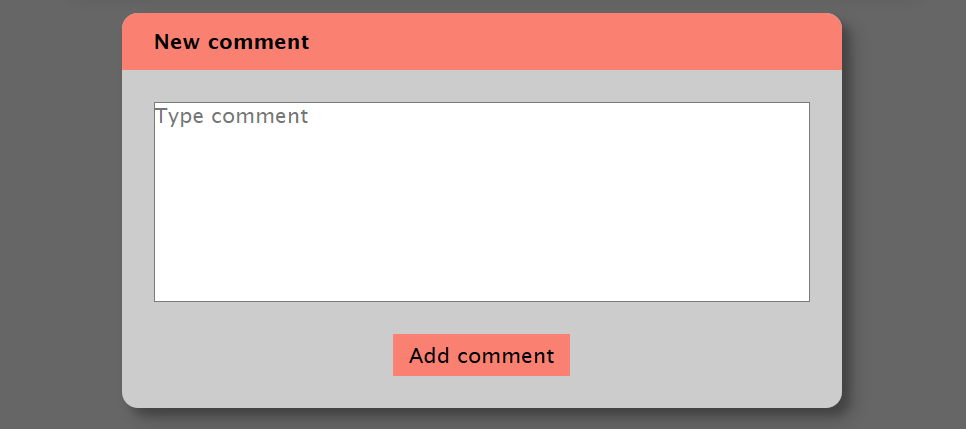
Refactor the existing code so that it uses **templates** to render each section ad its contents, instead of static HTML derived from a file. You may use the **tests** from the **previous lesson** to continuously check if everything is implemented correctly.

## My Cookbook – Part 5

Add **comments** functionality to the application. Unregistered users only see the list of comments:



Logged-in users see a **button**, that **displays a form** for adding a new comment Any registered user can leave a comment under any recipe.  




Use the following endpoints:

* To load all comments for a recipe: **data/comments?where=recipeId%3D%22{recipeId}%22** (GET)
* To create a new comment: **/data/comments** (POST)

Each comment has the following structure:

{

recipeId,

content

}

Where recipeId is the Id of the recipe, which the comment is associated with, and content is the comment's text content.