sitepoint premium

Handout

Your First Meteor 1.2 Application

Lesson 5 - Step 1

David Turnbull

Templates

In this chapter, we're going to start building the user interface for the Leaderboard application. This involves creating our first **templates**.

Templates are used to create a connection between the interface and our JavaScript code. When we place interface elements inside a template, we're able to reference and manipulate those elements using application logic. Before we get started, add a basic page structure to the project's HTML file. This includes head tags, `title` tags, body tags:

```html
<head>
    <title>Leaderboard</title>
</head>
<body>
</body>
```

A few things do appear to be missing though. We haven't included the `html` tags, we haven't included any JavaScript files, and we haven't included any CSS files. This is because Meteor takes care of them for us - it automatically adds those tags and includes any resources that are inside the project's folder. This isn't the most remarkable feature in the world, but one of Meteor's core philosophies is developer happiness. So there's plenty of time saving features like this throughout the framework.

Creating Our First Template

To create our first template, add the following code to the bottom of the project's HTML:

```html
<template name="Leaderboard"></template>
```

Within this template, include some static text such as "Hello world". Here, we're using a template tag to create a template and a name attribute that will allow us to distinguish between the templates we create. It doesn't matter what we name the templates but the names will be easier to remember if they are relevant to the content inside the template.

If you save the file in it's current state though, the template won't appear inside the web browser. This is because, by default, templates don't appear inside the interface. This might sound weird, but consider that in some cases you might want a template to appear at certain times.

As such, we need to manually include templates inside the interface. This does require an extra step, but it ultimately provides a lot of flexibility. Place the following tag in the HTML file between the body tags:

```
{{> leaderboard}}
```

Obviously this isn't an HTML tag. Instead, the use of double curly braces means this is the Spacebar's syntax, which is Meteor Syntax for building dynamic interfaces. It's the syntax that bridges the gap between our HTML and our JavaScript. We'll learn more about Spacebars as we progress through this course.