

## Unit 14.1 Highlights

(Source: <https://github.com/eric/express-handlebars>)

**handlebars** - JavaScript based templating/viewing engine, based on the mustache template language `{{mustache}}`, which enables developers to use static template files in node/express application. At runtime, the template engine replaces variables in a template file with actual values, and transforms the template into an HTML file sent to the client.

**response.render()** - Renders a view and sends the rendered HTML string to the client.

Handlebars Directory Structure

```
.
├── app.js
├── views
│   ├── home.handlebars
│   └── layouts
│       └── main.handlebars
```

### Ex. 1

```
var express = require("express");
var exphbs = require("express-handlebars");

var app = express();

var PORT = process.env.PORT || 8080;

app.engine("handlebars", exphbs({ defaultLayout: "main" }));
app.set("view engine", "handlebars");

app.get('/', function (req, res) {
  var data = {
    lunch: "PB & J",
    dinner: "Steak & potatoes",
    daysOfWorkWeek: [{day: 'Monday'}, {day: 'Tuesday'}, {day: 'Wednesday'}, {day: 'Thursday'}, {day: 'Friday'}]
  }
  res.render('home', data);
});
```

**{{ variable name or javascript expression }}** - handlebars returns HTML-escaped values evaluated within double curly braces

**{{{ body }}}** - handlebars returns non-HTML-escaped values evaluated within double curly braces, for different evaluation

**{{#if variable\_name}} {{/if}}** - handlebars built-in helper to evaluate true/false values for conditional logic

**{{#each item}} {{/each}}** - handlebars built-in helper which runs iterative code for every item in an array / list

**views/layouts/main.handlebars**

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Example App</title>
</head>
<body>

  {{{body}}}

</body>
</html>
```

**views/home.handlebars:**

```
<h1>For lunch, I'll eat a {{lunch}}</h1>
<h1>For dinner, I'll eat {{dinner}}</h1>
I will eat {{lunch}} & {{dinner}} on..
<ul>
  {{#each daysOfWorkWeek }}
    <li>{{day}}</li>
  {{/each}}
</ul>
```