



## TUTORIAL

# How To Use EJS to Template Your Node Application

Node.js JavaScript

By [Chris On Code](#)

Posted September 19, 2020 25.2k

English



## Introduction

When creating quick on-the-fly Node applications, an easy and fast way to template our application is sometimes necessary.

Jade comes as the view engine for Express by default but Jade syntax can be overly complex for many use cases. EJS is one alternative does that job well and is very easy to set up. Let's take a look at how we can create a simple application and use EJS to **include repeatable parts of our site (partials)** and **pass data to our views**.

---

## Setting up the Demo App

We will be making two pages for our application with one page with **full width** and the other with a **sidebar**.

**Get the code:** You can find a git repo of the complete demo code on GitHub here

## File Structure

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Enter your email address

Sign Up

```
----- footer.ejs
----- head.ejs
----- header.ejs
----- pages
----- index.ejs
----- about.ejs
- package.json
- server.js
```

`package.json` will hold our Node application information and the dependencies we need (express and EJS). `server.js` will hold our Express server setup, configuration. We'll define our routes to our pages here.

## Node Setup

Let's go into our `package.json` file and set up our project there.

`package.json`

```
{
  "name": "node-ejs",
  "main": "server.js",
  "dependencies": {
    "ejs": "^3.1.5",
    "express": "^4.17.1"
  }
}
```

All we will need is Express and EJS. Now we have to install the dependencies we just defined. Go ahead and run:

```
$ npm install
```

With all of our dependencies installed, let's configure our application to use EJS and set up our routes for the two pages we need: the index page (full width) and the about page (sidebar). We will do all of this inside our `server.js` file.

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Sign Up

```
// set the view engine to ejs
app.set('view engine', 'ejs');

// use res.render to load up an ejs view file

// index page
app.get('/', function(req, res) {
  res.render('pages/index');
});

// about page
app.get('/about', function(req, res) {
  res.render('pages/about');
});

app.listen(8080);
console.log('8080 is the magic port');
```

Here we define our application and set it to show on port 8080. We also have to set EJS as the view engine for our Express application using `app.set('view engine', 'ejs');`. Notice how we send a view to the user by using `res.render()`. It is important to note that **res.render() will look in a views folder** for the view. So we only have to define `pages/index` since the full path is `views/pages/index`.

## Start Up our Server

Go ahead and start the server using:

```
$ node server.js
```

Now we can see our application in the browser at `http://localhost:8080` and `http://localhost:8080/about`. Our application is set up and we have to define our view files and see how EJS works there.

---

## Create the EJS Partial

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ×

Sign Up

---

```
<meta charset="UTF-8">
<title>EJS Is Fun</title>

<!-- CSS (load bootstrap from a CDN) -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.5
<style>
  body { padding-top:50px; }
</style>
```

views/partials/header.ejs

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="/">EJS Is Fun</a>
  <ul class="navbar-nav mr-auto">
    <li class="nav-item">
      <a class="nav-link" href="/">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="/about">About</a>
    </li>
  </ul>
</nav>
```

views/partials/footer.ejs

```
<p class="text-center text-muted">© Copyright 2020 The Awesome People</p>
```

---

## Add the EJS Partials to Views

We have our partials defined now. All we have to do is include them in our views. Let's go into `index.ejs` and `about.ejs` and use the `include` syntax to add the partials.

## Syntax for including an EJS Partial

Use `<%- include( 'RELATIVE/PATH/TO/FILE' ) %>` to embed an EJS partial in another file.

---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ×

Enter your email address

Sign Up

---

```

<!DOCTYPE html>
<html lang="en">
<head>
  <%- include('../partials/head'); %>
</head>
<body class="container">

<header>
  <%- include('../partials/header'); %>
</header>

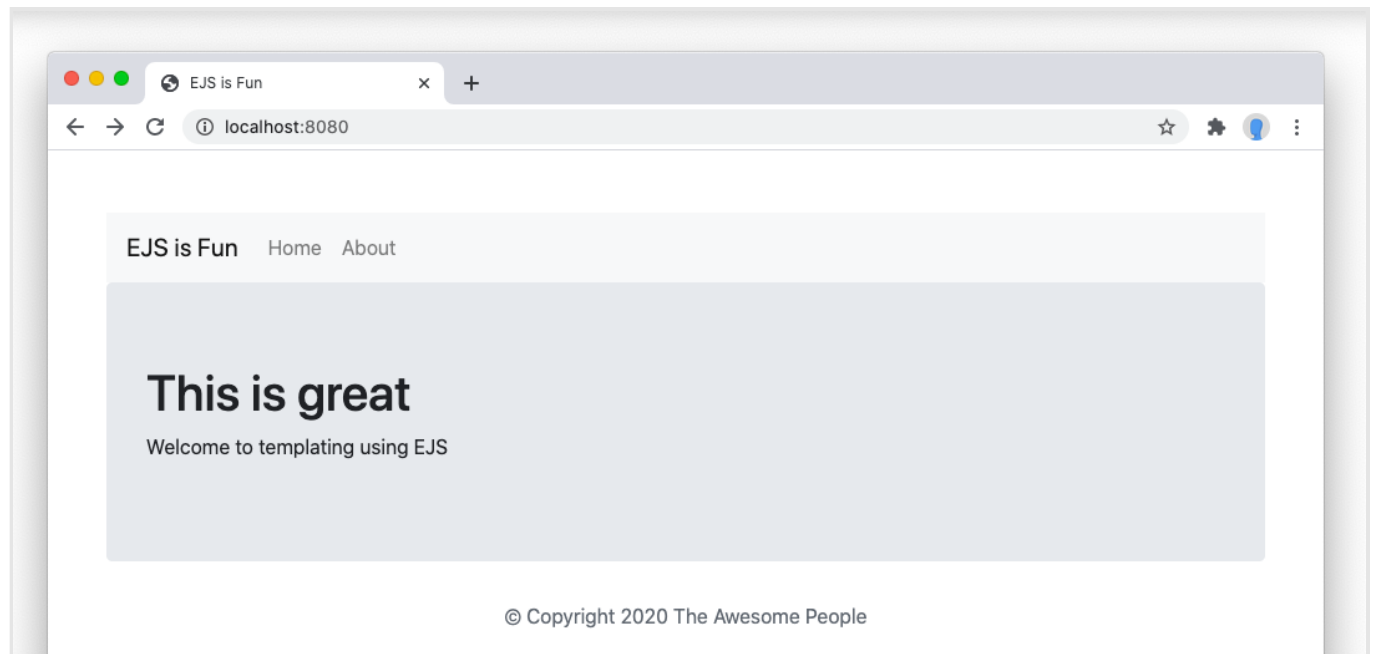
<main>
  <div class="jumbotron">
    <h1>This is great</h1>
    <p>Welcome to templating using EJS</p>
  </div>
</main>

<footer>
  <%- include('../partials/footer'); %>
</footer>

</body>
</html>

```

Now we can see our defined view in the browser at <http://localhost:8080>.



**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address

Sign Up

```
<!DOCTYPE html>
<html lang="en">
<head>
  <%- include('../partials/head'); %>
</head>
<body class="container">

<header>
  <%- include('../partials/header'); %>
</header>

<main>
<div class="row">
  <div class="col-sm-8">
    <div class="jumbotron">
      <h1>This is great</h1>
      <p>Welcome to templating using EJS</p>
    </div>
  </div>

  <div class="col-sm-4">
    <div class="well">
      <h3>Look I'm A Sidebar!</h3>
    </div>
  </div>

</div>
</main>

<footer>
  <%- include('../partials/footer'); %>
</footer>

</body>
</html>
```

If we visit <http://localhost:8080/about>, we can see our about page with a sidebar!

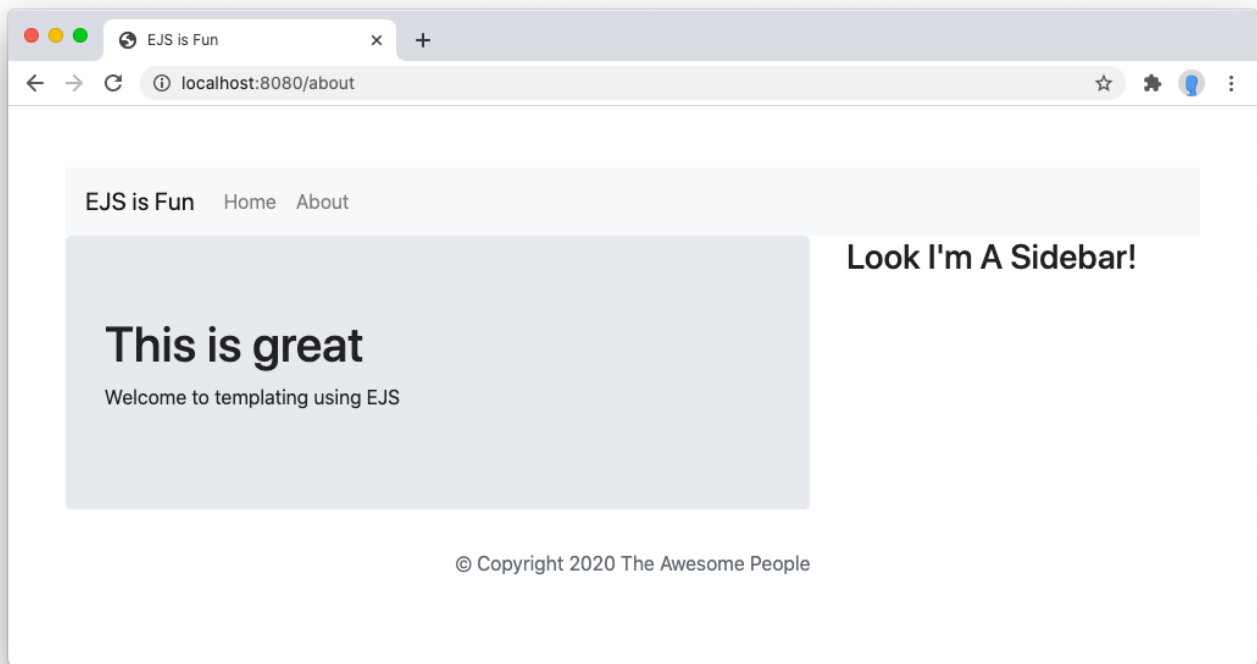
---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address

Sign Up

---



Now we can start using EJS for passing data from our Node application to our views.

---

## Pass Data to Views and Partial

Let's define some basic variables and a list to pass to our home page. Go back into your `server.js` file and add the following inside your `app.get('/')` route.

`server.js`

```
// index page
app.get('/', function(req, res) {
  var mascots = [
    { name: 'Sammy', organization: "DigitalOcean", birth_year: 2012},
    { name: 'Tux', organization: "Linux", birth_year: 1996},
    { name: 'Moby Dock', organization: "Docker", birth_year: 2013}
  ];
  var tagline = "No programming concept is complete without a cute animal mascot.";

  res.render('pages/index', {
```

---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address

Sign Up

---

We have created a list called `mascots` and a simple string called `tagline`. Let's go into our `index.ejs` file and use them.

## Render a Single Variable in EJS

To echo a single variable, we just use `<%= tagline %>`. Let's add this to our `index.ejs` file:

views/pages/index.ejs

```
...
<h2>Variable</h2>
<p><%= tagline %></p>
...
```

## Loop Over Data in EJS

To loop over our data, we will use `.forEach`. Let's add this to our view file:

views/pages/index.ejs

```
...
<ul>
  <% mascots.forEach(function(mascot) { %>
    <li>
      <strong><%= mascot.name %></strong>
      representing <%= mascot.organization %>, born <%= mascot.birth_year %>
    </li>
  <% }); %>
</ul>
...
```

Now we can see in our browser the new information we have added!

---

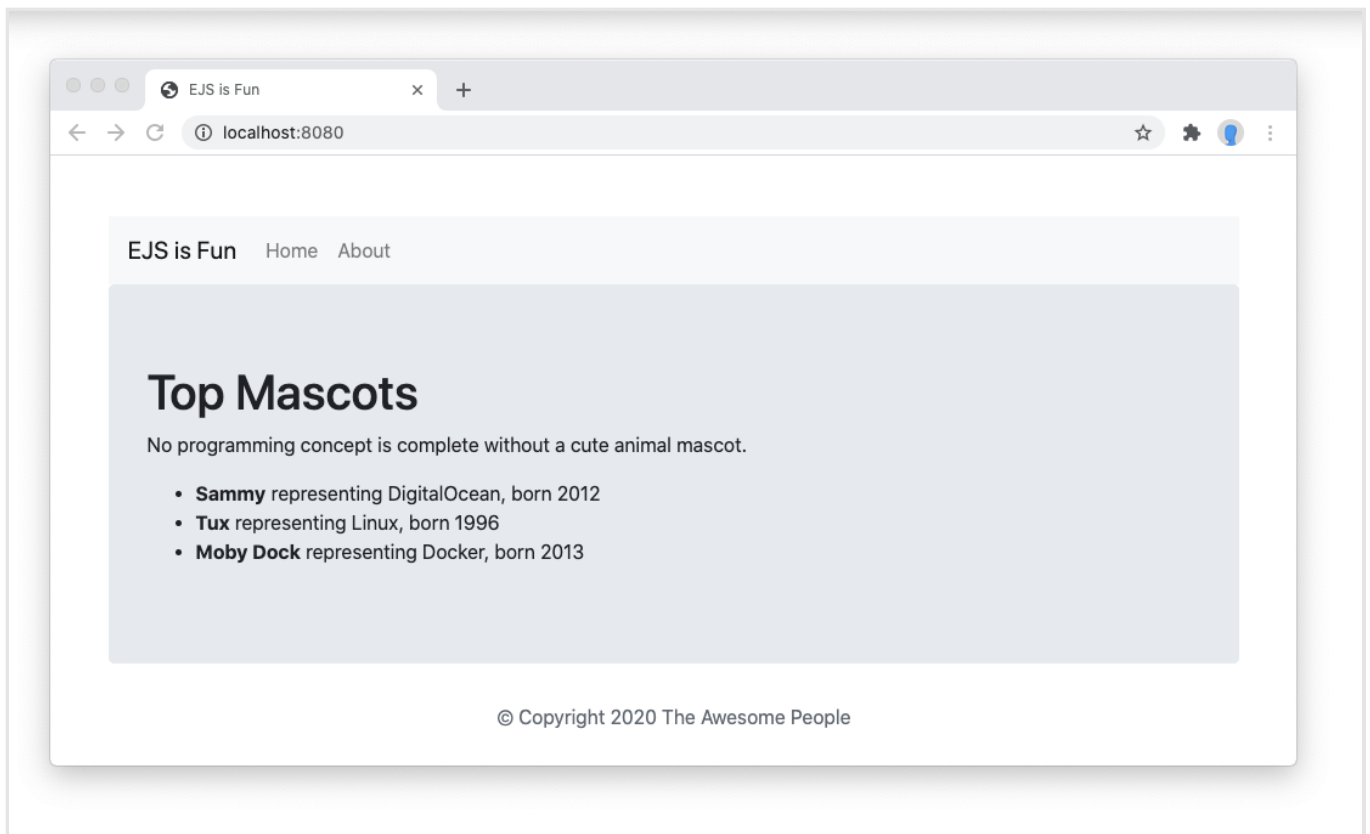
**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Sign Up

---





## Pass Data to a Partial in EJS

The EJS partial has access to all the same data as the parent view. But be careful: If you are referencing a variable in a partial, it needs to be defined in every view that uses the partial or it will throw an error.

You can also define and pass variables to an EJS partial in the include syntax like this:

views/pages/about.ejs

```
...
<header>
  <%- include('../partials/header', {variant:'compact'}) %>
</header>
...
```

But you need to again be careful about assuming a variable has been defined.

---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address

Sign Up

---

```
...
<em>Variant: <%= typeof variant != 'undefined' ? variant : 'default' %></em>
...
```

In the line above, the EJS code is rendering the value of `variant` if it's defined, and `default` if not.

---

## Conclusion

EJS lets us spin up quick applications when we don't need anything too complex. By using partials and having the ability to easily pass variables to our views, we can build some great applications quickly.

For more reference on EJS see the [official docs here](#).

Was this helpful?

Yes

No



[Report an issue](#)

## About the authors



**Chris On Code**

has authored 19 tutorials.



**Andy  
Hattermer**

Editor

---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Enter your email address

Sign Up

---

## Comments

### 1 Comment

Leave a comment...

Sign In to Comment

^ [stadibenker](#) September 29, 2020  
0 Great article! Thank you so much! :)  
[Reply](#) [Report](#)

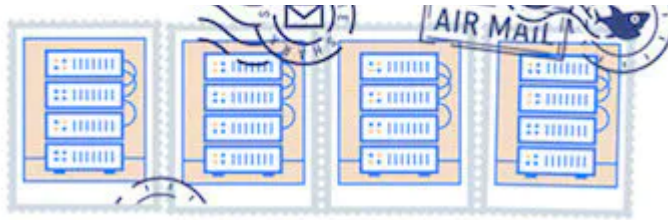


This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics. ×

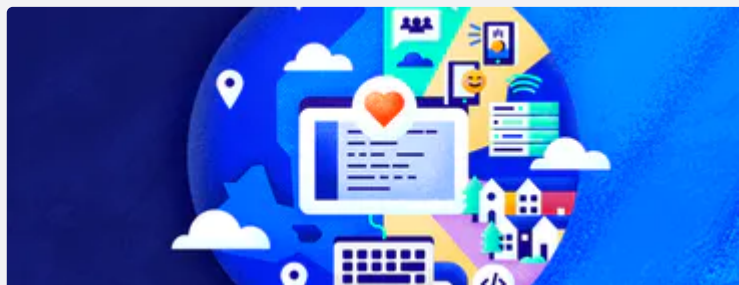
Enter your email address

Sign Up



**GET OUR BIWEEKLY NEWSLETTER**

Sign up for Infrastructure as a  
Newsletter.



**HUB FOR GOOD**

Working on improving health  
and education, reducing  
inequality, and spurring  
economic growth? We'd like to  
help.



**DEPLOY: NOV 10, 2020**

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



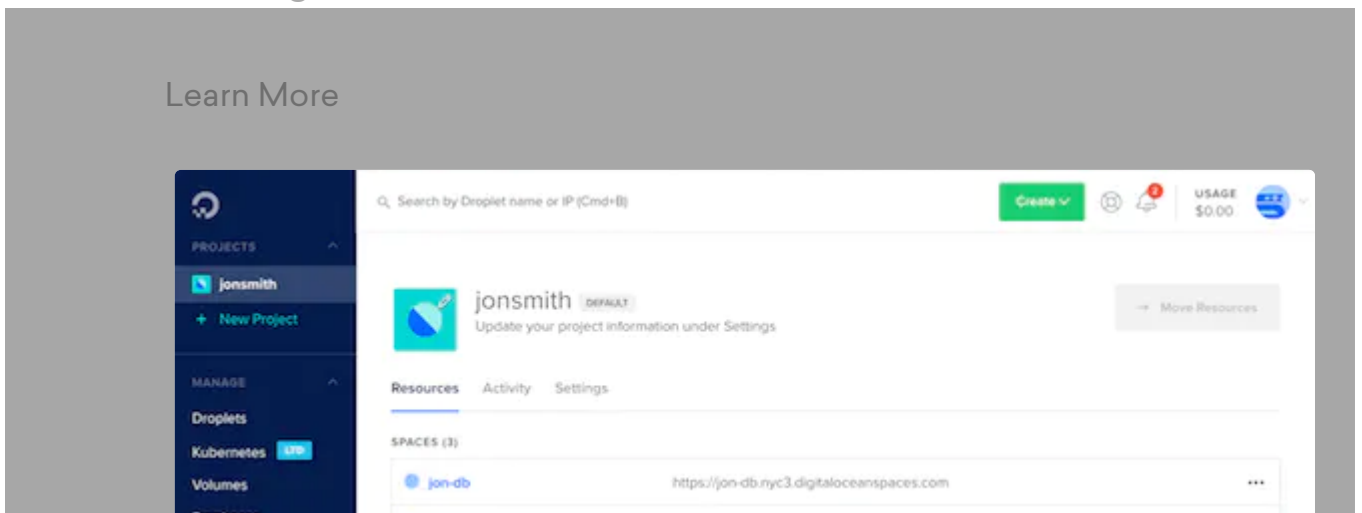
Enter your email address

**Sign Up**

## Welcome to the developer cloud

DigitalOcean makes it simple to launch in the cloud and scale up as you grow – whether you’re running one virtual machine or ten thousand.

Learn More



© 2020 DigitalOcean, LLC. All rights reserved.

Company

About

Leadership

Blog

Careers

Partners

Referral Program

Press

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Enter your email address

Sign Up

[Droplets](#)  
[Kubernetes](#)  
[Managed Databases](#)  
[Spaces](#)  
[Marketplace](#)  
[Load Balancers](#)  
[Block Storage](#)  
[API Documentation](#)  
[Documentation](#)  
[Release Notes](#)

[Tools and Integrations](#)  
[Tags](#)  
[Product Ideas](#)  
[Write for DigitalOcean](#)  
[Presentation Grants](#)  
[Hatch Startup Program](#)  
[Shop Swag](#)  
[Research Program](#)  
[Open Source](#)  
[Code of Conduct](#)

[Trouble Signing In?](#)  
[Sales](#)  
[Report Abuse](#)  
[System Status](#)

---

**Sign up for our newsletter** Get the latest tutorials on SysAdmin and open source topics.



Enter your email address

**Sign Up**

---