



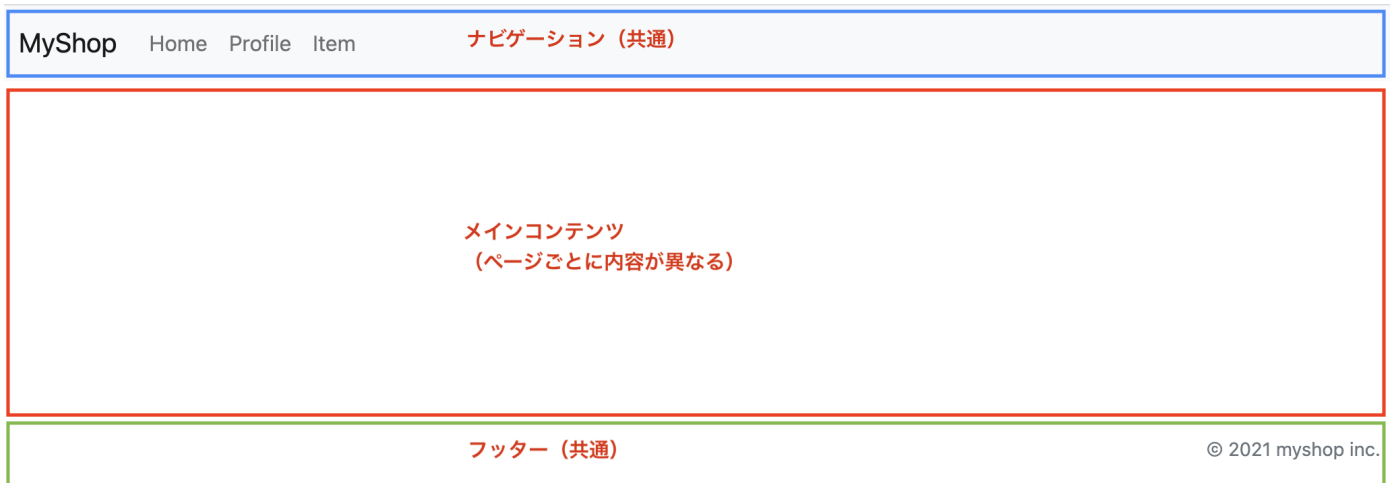
14. EJS のレイアウト

レイアウト

`html` , `head` タグやナビゲーション、フッターといった領域は、基本的にHTMLで共通な部分なので、すべてページで HTML を記述するのは非効率です。HTMLの外枠部分を共通化したファイルをレイアウトといい、`body` で表示するメインコンテンツはページごとに分離できます。

レイアウトファイル

レイアウトファイルにすると、共通部分は以下のようになります。



```
<!DOCTYPE html>
<html lang="ja">

<!-- head は共通 -->
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>Document</title>
</head>

<body>
  <!-- ナビゲーション は共通 -->
```

```
<div class="container">
  <!-- メインコンテンツはページごと -->
</div>

<!-- フッターは共通 -->
</body>

</html>
```

EJS のレイアウト

EJS のレイアウトは `express-ejs-layouts` モジュールを利用すると便利です。

インストール

`express-ejs-layouts` をインストールします。

```
% npm i express-ejs-layouts
```

モジュール読み込み

`express-ejs-layouts` をモジュールを読み込み、ミドルウェアで有効にします。

```
const layouts = require('express-ejs-layouts');
app.use(layouts);
```

レイアウトの任意指定

レイアウトファイルはデフォルトで `views/layout.ejs` が読み込まれます。レイアウトを任意指定する場合は `app.set()` で設定します。

```
app.set('layout', レイアウトファイルパス)
```

例えば、`layouts/default.ejs` をレイアウトとすると以下のようになります。

```
const layouts = require('express-ejs-layouts');
app.set('layout', 'layouts/default');
app.use(layouts);
```

レイアウトファイルの作成

レイアウトファイル `views/layouts/default.ejs` を作成し、HTML の共通部分を記述します。

body 部分の読み込み

レイアウトファイルで `body` の部分だけ読み込むには、`<%= body %>` を利用します。

```
<%= body %>
```

views/layouts/default.ejs

```
<!DOCTYPE html>
<html lang="ja">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>MyPage: <%= title %></title>
</head>

<body>
  <div class="container">
    <h1 class="h1 fs-2 pt-3"><%=title %></h1>

    <%= body %>
  </div>
</body>

</html>
```

render() でレイアウトを指定

`render()` でレイアウトを任意指定することもできます。例えば、`login.ejs` で `layouts/user.ejs` をレイアウトとすると以下ようになります。

```
res.render('login.ejs', { layout: 'user' });
```

メインコンテンツ

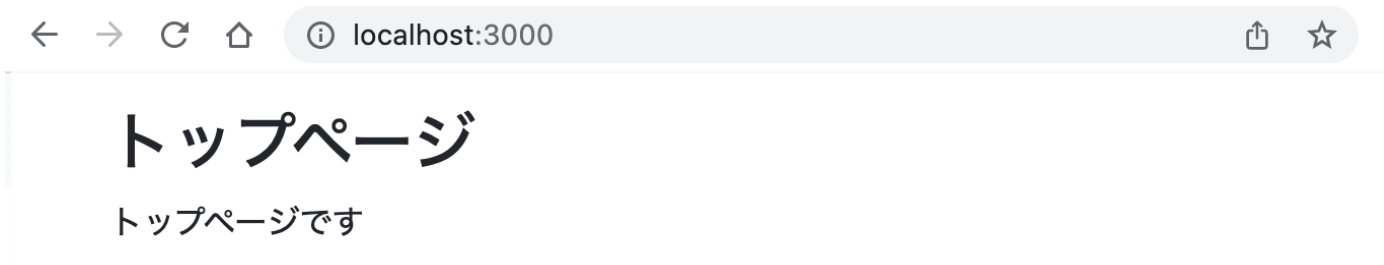
ページごとの **EJS** ファイルから、HTMLの外枠部分を削除します。

トップページ

./index.ejs

```
<p>
  トップページです
</p>
```

レイアウトが利用できるか確認してみましょう。



その他ページ

プロフィールや商品ページも、レイアウトに対応させます。

./profile/index.ejs

```
<div>
  <dl>
    <dt>名前</dt>
    <dd><%= user.name %></dd>
    <dt>出身地</dt>
    <dd><%= user.birthplace %></dd>
    <dt>趣味</dt>
    <dd>
      <% user.hobby.forEach((hobby) => { %>
        <%= hobby %>
      <% }) %>
    </dd>
  </dl>
</div>
```

item/index.ejs

```
<dl>
  <% items.forEach ((item) => { %>
    <dd><a href="/item/<%= item.id %>"><%= item.name %></a></dd>
    <dd><%= item.price %>円</dd>
  <% }) %>
</dl>
```

item/index.ejs

```
<div>
  <% if (item) { %>
    <dl>
      <dt>商品名</dt>
      <dd><%= item.name %></dd>
      <dt>価格</dt>
      <dd><%= item.price %>円</dd>
    </dl>
  <% } else { %>
    <div><%= message %></div>
  <% } %>
</div>
```

コンポーネント

コンポーネントは、同じような処理を部品化したファイルです。コンポーネントを作成すると、色々なページから外部読み込みできます。

include() で外部ファイルを読み込む

`include()` を利用すると外部ファイルを読み込むことができます。

```
<%- include(ファイルパス) %>
```

ナビゲーション

`views/components/nav.ejs` を作成します。

```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="/">MyShop</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <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="/profile">Profile</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/item">Item</a>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

レイアウトからナビゲーションファイルを読み込みます。

```
<%- include('../components/nav') %>
```

フッター

views/components/footer.ejs を作成します。

```

<footer class="footer">
  <div class="container-fluid">
    <p class="text-muted text-end"><small>© 2021 myshop inc.</small></p>
  </div>
</footer>

```

レイアウトからフッターファイルを読み込みます。

```
<%- include('../components/footer') %>
```

レイアウトとコンポーネントの確認

作成したレイアウト、コンポーネントが表示されるか、各ページで確認してみましょう。



ソース

server.js

```
const express = require('express')
const routes = require('./routes')

require('dotenv').config()
const host = process.env.HOST
const port = process.env.PORT

const app = express()

//レイアウト
const layouts = require('express-ejs-layouts')
app.set('layout', 'layouts/default');
app.use(layouts);

app.set('view engine', 'ejs')
app.use(routes)

app.listen(port, host, () => {
  console.log(`Server listen: http://${host}:${port}`)
})
```

views/layouts/default.ejs

```
<!DOCTYPE html>
<html lang="ja">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>MyShop: <%= title %></title>
</head>

<body>
  <%- include('../components/nav') %>

  <div class="container">
    <h1 class="h1 fs-2 pt-3"><%=title %></h1>

    <%- body %>
  </div>
```

Copy

```
<%- include('../components/footer') %>
</body>

</html>
```

当サイトの教材をはじめとするコンテンツ（テキスト、画像等）の無断転載・無断使用を固く禁じます。これらのコンテンツについて権利者の許可なく複製、転用等する事は法律で禁止されています。尚、当ウェブサイトの内容をWeb、雑誌、書籍等へ転載、掲載する場合は「ロジコヤ」までご連絡ください。