

Using Bootstrap

Open-source web browser framework တစ်ခုဖြစ်တဲ့ Bootstrap ကို သုံးပြီး၊ မိမိ webpage ရဲ့ user components တွေကို ပိုမိုပြီး ရှင်းရှင်းလင်းလင်းနဲ့ ဆွဲဆောင်မှုရှိအောင် ပြင်ပါမယ်။

Bootstrap က client-side framework ဖြစ်တဲ့အတွက် server နဲ့တော့ တိုက်ရိုက်ပါဝင်ပတ်သတ်မှု မရှိပါဘူး။ Bootstrap ရဲ့ CSS နဲ့ JavaScript files တွေကို reference လုပ်ထားတဲ့ HTML responses တွေကိုပဲ ထောက်ပံ့ပေးထားတာပါ။ HTML, CSS နဲ့ JavaScript code တွေကမှတဆင့် မိမိလိုချင်တဲ့ user interface elements တွေကို instantiate လုပ်ပေးရပါမယ်။ အဲဒါတွေအားလုံး လုပ်ဖို့ အကောင်းဆုံး နေရာကတော့ templates ပါ။

Bootstrap ကို သုံးဖို့ အရိုးဆုံးနဲ့ အလွယ်ကူဆုံးနည်းလမ်းကတော့ Bootstrap ကနေ recommend ပေးထားတဲ့ document တွေကို HTML templates တွေထဲမှာ လိုအပ်သလို ပြောင်းလဲပြီး ထည့်သွင်း အသုံးပြုတာပါပဲ။

ပထမဆုံး CSS နဲ့ JavaScript file တွေကို base template file ထဲကို reference လုပ်ပါမယ်။

Bootstrap website ကို သွားပါ။ bootstrap.min.css file ရဲ့ link ကို head အပိုင်းမှာ copy ကူးထည့်ပါ။ script files တွေကိုတော့ body tab ရဲ့ အောက်ခြေမှာ copy ကူးထည့်ပေးပါ။

templates/layout.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <!-- Required meta tags -->
```

```
    <meta charset="utf-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
    <!-- Bootstrap CSS -->
```

```

<link rel="stylesheet" ref="{{ url_for('static', filename=' bootstrap.min.css ') }}">
....
</head>
<body>
    <div class="container">
        {% block content %}{% endblock %}
    </div>

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="jquery-3.2.1.slim.min.js"></script>
    <script src="popper.min.js "></script>
    <script src="bootstrap.min.js"></script>
</body>
</html>

```

file ကို run ကြည့်တဲ့အခါ Bootstrap effect ပါလာပေမဲ့ လိုချင်တဲ့ပုံစံ မဖြစ်လာသေးပါဘူး။ ဒါကြောင့် လိုအပ်တာကို ထပ်ထည့်ပါမယ်။ <body> အပိုင်းထဲမှာ nav bar ထည့်ပါမယ်။ nav bar ရဲ့ ဘယ်အခြမ်းမှာ Flask Blog, Home နဲ့ About ပါပြီး၊ ညာဘက်မှာ Login နဲ့ Register ပါမယ်။ navbar ရဲ့ အောက် main အပိုင်းကို Blog post ပြမဲ့ main section နဲ့ Sidebar ထည့်ပါမယ်။

လိုအပ်တဲ့ codes တွေကို code snippet file တွေထဲမှ copy ကူးထည့်ပါမယ်။

templates/layout.html

```

<!DOCTYPE html>
<html>
<head>
...
</head>
<body>

```

```

<header class="site-header">
  <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">
    <div class="container">
      <a class="navbar-brand mr-4" href="/">Flask Blog</a>
      <button class="navbar-toggler" type="button" data-toggle="collapse"
        data-target="#navbarToggle" aria-controls="navbarToggle" aria-expanded="false"
        aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarToggle">
        <div class="navbar-nav mr-auto">
          <a class="nav-item nav-link" href="/">Home</a>
          <a class="nav-item nav-link" href="/about">About</a>
        </div>

        <!-- Navbar Right Side -->
        <div class="navbar-nav">
          <a class="nav-item nav-link" href="#">Login</a>
          <a class="nav-item nav-link" href="#">Register</a>
        </div>
      </div>
    </nav>
  </header>

```

```

<main role="main" class="container">
  <div class="row">
    <div class="col-md-8">
      {% block content %}{% endblock %}
    </div>

```

```

<div class="col-md-4">
  <div class="content-section">
    <h3>Our Sidebar</h3>
    <p class='text-muted'>You can put any information here you'd like.
    <ul class="list-group">
      <li class="list-group-item list-group-item-light">Latest Posts</li>
      <li class="list-group-item list-group-item-light">Announcements</li>
      <li class="list-group-item list-group-item-light">Calendars</li>
      <li class="list-group-item list-group-item-light">etc</li>
    </ul>
  </p>
</div>
</div>
</div>
</main>
<!-- Optional JavaScript -->
...
</body>
</html>

```

ဒီနေရာမှာ ဘာထပ်ထည့်ပေးရမလဲဆိုရင် main.css file ကို reference လုပ်ပေးရပါမယ်။ main.css file ကို static folder ဆောက်ပြီး ထည့်ထားရမှာပါ။ Jinja2 ရဲ့ helper function ဖြစ်တဲ့ url_for ကိုသုံးပါမယ်။

နောက် Blog post တွေ ဖော်ပြတဲ့အခါ မိမိလိုသလိုပုံဖော်ဖို့ home.html ကိုလည်းပြင်ပါမယ်။

templates/home.html

```

{% extends "layout.html" %}
{% block content %}

```

```
{% for post in posts %}
  <article class="media content-section">
    <div class="media-body">
      <div class="article-metadata">
        <a class="mr-2" href="#">{{ post.author }}</a>
        <small class="text-muted">{{ post.date_posted }}</small>
      </div>
      <h2><a class="article-title" href="#">{{ post.title }}</a></h2>
      <p class="article-content">{{ post.content }}</p>
    </div>
  </article>
{% endfor %}
{% endblock content %}
```

လိုအပ်တာတွေ ပြင်ဆင်ပြီးရင်တော့ file ကို ပြန် run ကြည့်ပါမယ်။