

Template Inheritance

In this lesson, we will be learning the method of creating template inheritance with Jinja!

WE'LL COVER THE FOLLOWING ^

- Introduction
 - `home.html` + `style.css`
 - `about.html` + `style.css`
 - `base.html`
- Blocks
- Complete implementation

Introduction

Template inheritance is a compelling feature of Jinja that lets us **reuse** templates. Consider an example where we have two templates: `home.html` and `about.html`. Both of these files use the same **CSS** file called `style.css`. You can see the files and their outputs below.

`home.html` + `style.css` #

HTML CSS

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <link rel="stylesheet" href="{{url_for('static', filename='style.css')}}">
5     <title>Home Page - Jinja Demo</title>
6   </head>
7   <body>
8     <div id="header"> JINJA DEMO </div>
9     <div id="content">
10      <h1>Home Page</h1>
11      <p>Welcome to the Jinja2 Demo.</p>
12    </div>
13    <div id="footer"> Copyright © 2019
14  </div>
15 </body>
16 </html>
```

```
1 #header {
2   padding: 30px;
3   text-align: center;
4   background-color: #140005;
5   color: white;
6   font-size: 40px;
7 }
8 #footer {
9   position: fixed;
10  width: 100%;
11  background-color: #BBC4C2;
12  color: white;
13  text-align: center;
14  left: 0;
15  bottom: 0;
```

JINJA DEMO

Home Page

Welcome to the Jinja2 Demo.

Copyright © 2019 All Rights Reserved



Home Page

about.html + style.css #

HTML CSS

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <link rel="stylesheet" href="{{url_for('style')}}">
5     <title>About Page - Jinja Demo</title>
6   </head>
7   <body>
8     <div id="header"> JINJA DEMO </div>
9     <div id="content">
10      <h1>About Page</h1>
11      <p>In this lesson we are learning Jinja2</p>
12    </div>
13    <div id="footer"> Copyright © 2019 All Rights Reserved</div>
14  </body>
15 </html>
```

```
1 #header {
2   padding: 30px;
3   text-align: center;
4   background-color: #140005;
5   color: white;
6   font-size: 40px;
7 }
8 #footer {
9   position: fixed;
10  width: 100%;
11  background-color: #BBC4C2;
12  color: white;
13  text-align: center;
14  left: 0;
15  bottom: 0;
16 }
```

JINJA DEMO

About Page

In this lesson we are learning about Template Inheritance

Copyright © 2019 All Rights Reserved



About Page

You may have observed that these files contain common components.

💡 Jinja provides us a way to avoid rewriting these components.

Let's create a new template containing **only the shared components** from both templates. We will name this template `base.html`. We will be using placeholders for the elements that differ.

`base.html` #

```
<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" href="{{url_for('static', filename='format.css')}}" />

  <title>{% block title %}<!-- Placeholder for Title -->{% endblock %} - Jinja Demo</title>

  {% block head %}
  <!-- Placeholder for Other Imports -->
  {% endblock %}
</head>
<body>
  <div id="header"> JINJA DEMO </div>
  <div id="content">
    {% block content %}
    <!-- Placeholder for Page Content -->
```

```
{% endblock %}
</div>
<div id="footer"> Copyright © 2019 All Rights Reserved </div>

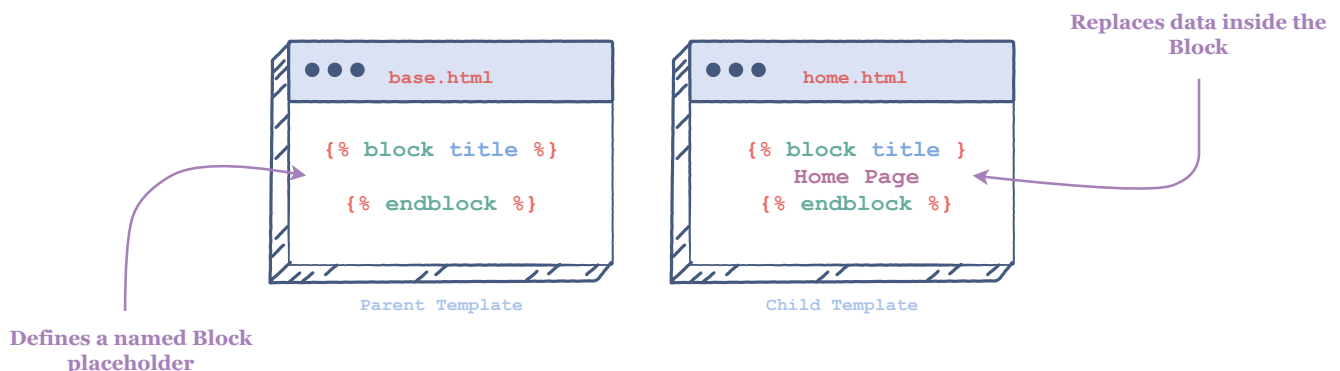
</body>
</html>
```

The placeholders we have used are actually the syntax for **blocks** in jinja!

Blocks

✦ **Note:** In jinja, the **blocks** are used as *placeholders* as well as *replacements*. In the **parent** template, these blocks are used as **placeholders**. Whereas in *child* template, they are used as **replacements**.

In the example shown above, **base.html** is a **parent** template. A **child** template would replace the placeholders as:



✦ **Note:** The blocks can be named whatever we want them to be. However, we can't define multiple blocks with the same name in a template.

Complete implementation

In the complete implementation given below, take a look at the **home.html** and **about.html** templates and observe how the blocks from **base.html** have been replaced in the child templates. Moreover, notice that we used an **extends** statement in **line 1** of the child templates to specify which template they are inherited from.

```
#header {  
  padding: 30px;  
  text-align: center;  
  
  background: #140005;  
  color: white;  
  font-size: 40px;  
}  
#footer {  
  position: fixed;  
  width: 100%;  
  background-color: #BBC4C2;  
  color: white;  
  text-align: center;  
  left: 0;  
  bottom: 0;  
}  
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
}  
  
li {  
  display: inline;  
}
```

Quick Quiz!

Q

Which of these statements do we use in the child templates to inherit them from `base.html`?

In the next lesson, we will work on our course project again!