

Lab 10 Part 3

Styling the website

Step 1: Bootstrap

We will style the website using a combination of bootstrap and custom css.

Copy the following line of code and paste it into the **base.html** file at the location shown below:

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

```
shop > templates > <> base.html > ...
1  {% load static %}
2  <!DOCTYPE html>
3  <html lang="en">
4  <head>
5  |   <meta charset="utf-8">
6  |   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

This line of code is used to make our website mobile responsive.

Copy the following line of code and paste it into the **base.html** file at the location shown below:

```
<link rel="stylesheet" href="{% static 'css/custom.css' %}">
```

```
6  |   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7  |   <link rel="stylesheet" href="{% static 'css/custom.css' %}">
```

This is the link to the custom css that we will provide ourselves.

Copy the following code containing the link to the bootstrap style sheet into the **base.html** file just after the link to the **custom.css** file:

```
<link
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
  rel="stylesheet"
  integrity="sha384-9ndCyUaIbzAi2FUVXJi0CjmCapSm07SnpJef0486qhLnuZ2cdeRh002iuK6FUUVM"
  crossorigin="anonymous"
/>
```

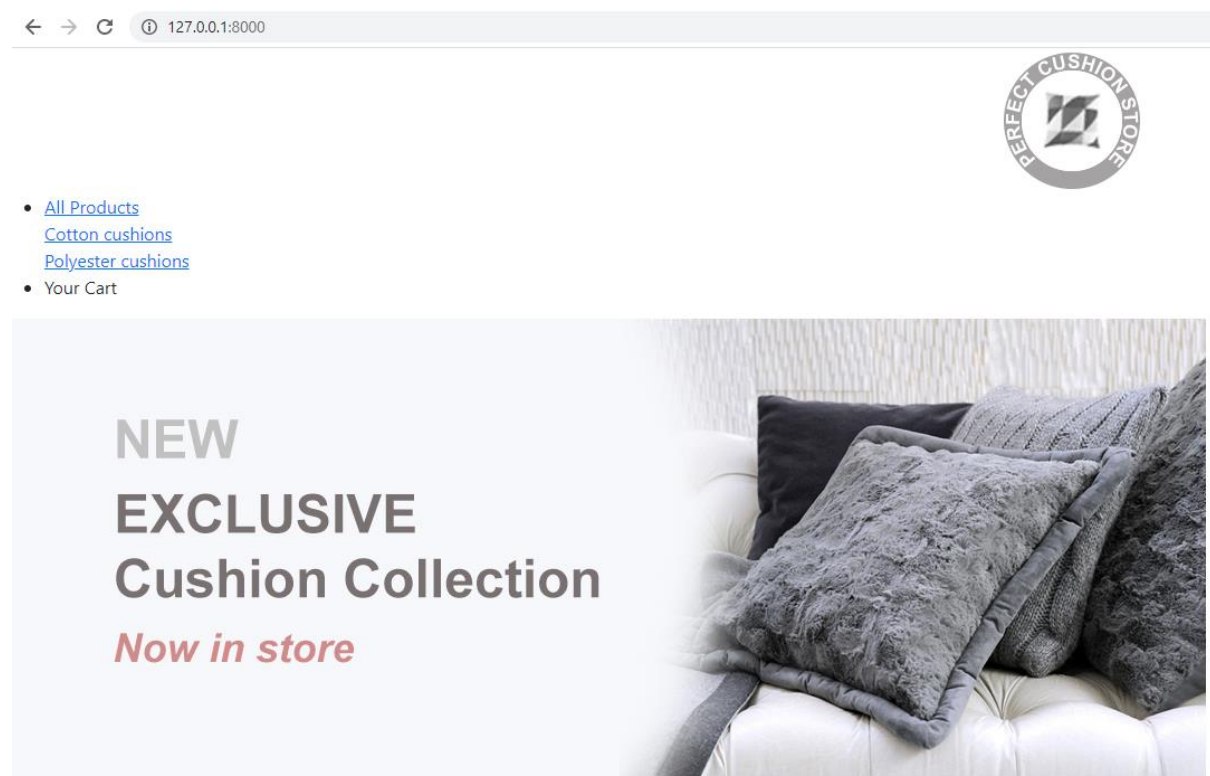
Copy the script tags provided below and paste them into the location of the **base.html** file as shown.

```
<!-- JavaScript Bundle with Popper -->
<script
  src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-
geWF76RCwLtnZ8qwWowPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68Sly3Te4Bkz"
  crossorigin="anonymous"
></script>
```

base.html

```
63 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js"
64 integrity="sha384-
65 OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJua0e923+mo//f6V8Qbsw3"
66 crossorigin="anonymous"></script>
67 </body>
68 </html>
```

Run the server and you will see that bootstrap is up and running as the text on the website now have nicer looking fonts applied:

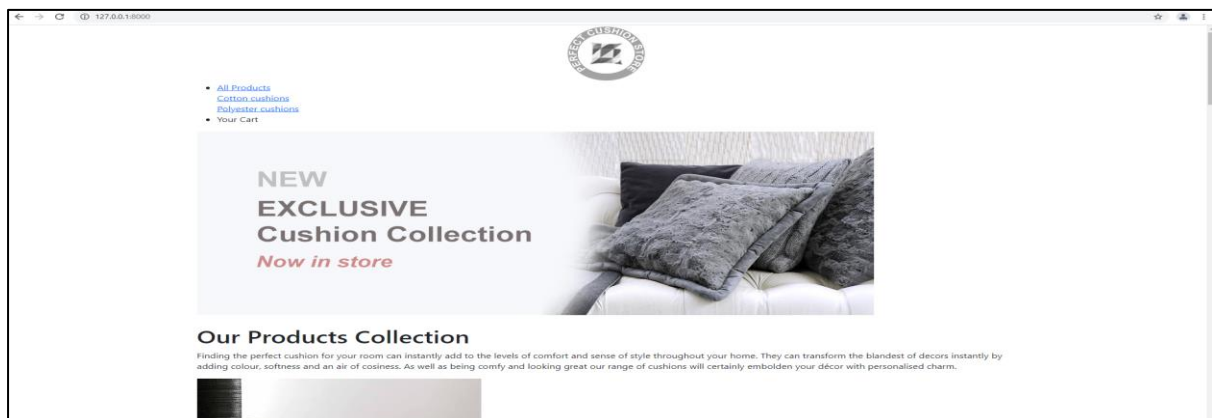


Step 2: Styling the header, navbar, and footer with bootstrap & css

In **base.html** add a div container as shown below and define a class inside it called container:

```
15 <div class="container">
16     {% include 'header.html' %}
17     {% include 'navbar.html' %}
18     {% block content %}
19     {% endblock %}
20
21     {% include 'footer.html' %}
22 </div>
```

Restart the server and access the website again and you will see that the container has been applied as the content has now moved away from the right and left edges as shown by the screenshot below.



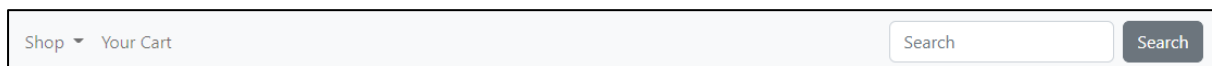
Open **header.html** and add the anchor href tag shown below so that when a user clicks on the logo, all the products will be displayed. Try the link to make sure it works.

```
shop > templates > <> header.html > ...
1  {% load static %}
2  <header>
3      <div style="text-align: center;">
4          <a href="{% url 'shop:all_products' %}" >
5              </a>
7      </div>
8  </header>
```

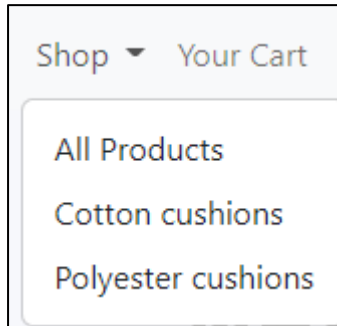
Open **navbar.html** and delete all the code in the file and copy-paste the following code into the file. This is sample navbar code taken from the bootstrap website and edited slightly:

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-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 me-auto mb-2 mb-lg-0">
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-
expanded="false">
            Shop
          </a>
          <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
            <li><a class="dropdown-item" href="{% url 'shop:all_products' %}">All Products</a></li>
            {% for cat in links %}
            <li><a class="dropdown-item" href="{{cat.get_absolute_url}}">{{cat.name}}</a></li>
            {% endfor %}
          </ul>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Your Cart</a>
        </li>
      </ul>
      <form class="d-flex" action="" method="get">
        <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search" name="q">
        <button class="btn btn-secondary my-2 my-sm-0" type="submit">Search</button>
      </form>
    </div>
  </div>
</nav>
```

Refresh the home page and you will see that the navbar is applied:



If you click on the drop-down menu, you will see the two product categories as shown below:



Now we will add some custom css.

In VS Code create a new folder called **css** inside the **static** folder and inside the **css** folder create a new file called **custom.css** and copy-paste the following code:

```
/*Navbar*/
.nav-item {
    letter-spacing: .2em;
    font-size: 14px;
    text-transform: uppercase;
}
.dropdown-item {
    font-size: 14px;
    letter-spacing: .2em;
    text-transform: uppercase;
}
```

Refresh the browser and you will see that the items in the nav bar are now in uppercase and nicely spaced out as are the items in the drop-down menu.



Open **footer.html** & overwrite the code with the following:

```
<div class = "footer navbar-fixed-bottom my-footer">
    <p class = "text-center">&copy; Online Cushions. All Rights Reserved.</p>
</div>
```

We have just added some bootstrap classes and one custom class. The bootstrap class **navbar-fixed bottom** will pin the footer to the bottom of the screen. The bootstrap **text-center** class will center the paragraph on the page

```
shop > templates > <> footer.html > ...  
1 | <div class = "footer navbar-fixed-bottom my-footer">  
2 |   <p class = "text-center">copy; Online Cushions. All Rights Reserved.</p>  
3 | </div>
```

The **my-footer** class is a custom class that we will style using css

Open **custom.css** and add the following code:

```
/*Footer*/  
.my_footer {  
    background-color: #f8f9fa;  
    height: 60px;  
}  
.my_footer p {  
    padding-top: 20px;  
    font-size: 14px;  
}
```

Refresh the browser and you will see that the footer is now centered on the page.



Step 3: Styling the Category page with bootstrap & custom css

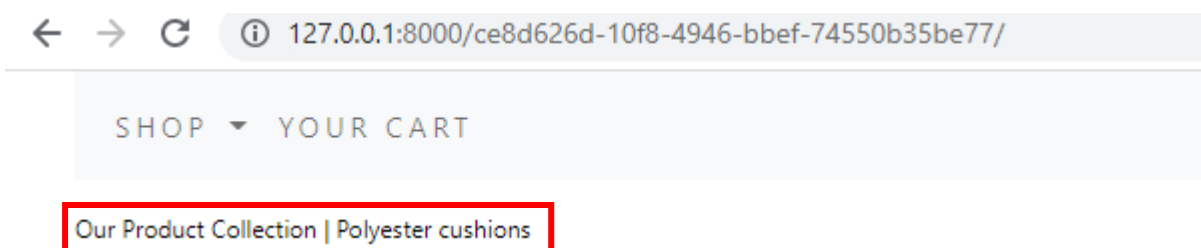
The first part of the category page that we will style is the breadcrumb navigation. Open the **category.html** template and create the following three classes inside the existing divs called **row**, **my_row_class** and **mx-auto**.

```
18 <!--Breadcrumb navigation-->
19 {% if category %}
20 <div class="row my_row_class">
21 <div class="mx-auto">
22 <p><a href="{% url 'shop:all_products' %}">Our Product Collection</a> | {{category.name}}</p>
23 </div>
24 </div>
25 {% endif %}
```

Open **custom.css** and copy-paste the following css code:

```
/*Category page*/
.my_row_class {
    padding-top: 15px;
}
.my_row_class .mx-auto p {
    color: #000;
    font-size: 12px;
}
.my_row_class .mx-auto p a {
    color: #000;
    font-size: 12px;
    text-decoration: none;
}
```

Refresh the browser and when you click on one of the product categories the breadcrumb navigation appears, and the styling has been applied:



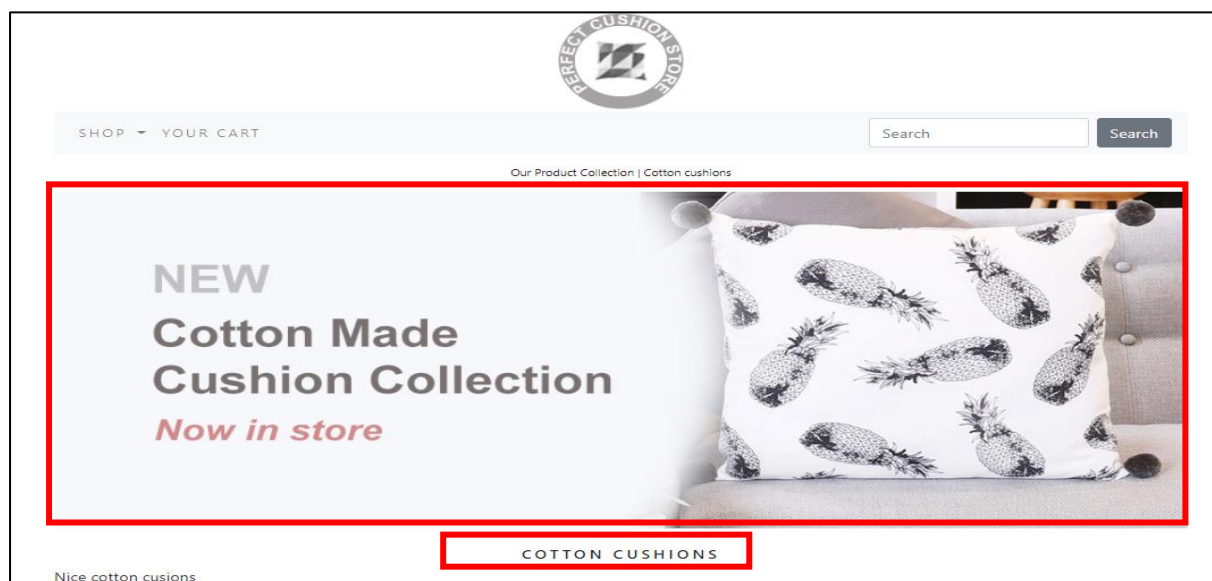
Next, we will focus on the category banner and description. Open **category.html** and add the following five classes to the section of html just after the breadcrumb section:

```
26 | <div class = mx-auto>
27 | {% if category %}
28 |   
29 | </div>
30 | <br>
31 | <div>
32 |   <h1 class = "text-center my_title" {{category.name}}</h1>
33 |   <p class = "text-justify" {{category.description}}</p>
34 | </div>
35 | {% else %}
```

Open **custom.css** and add the following css code for the two custom classes **my_image** and **my_title**:

```
/*Category page*/
.my_image {
  width: 100%;
  height: auto;
}
.my_title {
  font-size: 16px;
  text-transform: uppercase;
  letter-spacing: .2em;
}
```

Refresh the browser and when you click on one of the product categories you will see that the image is centered and the text for the name and description look better.



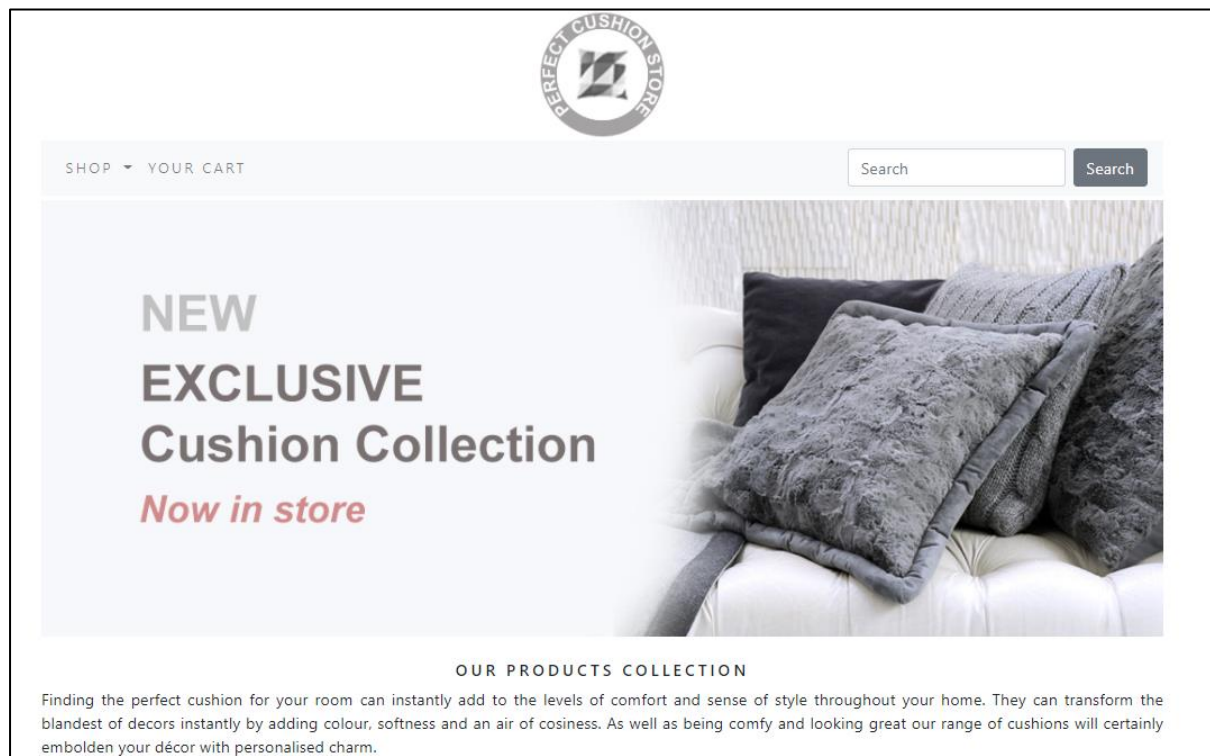
Now we will work on the styling for the banner image and text in the category page. Add the following classes to **category.html** as shown below:

```
35     {% else %}
36         Our Products Collection
41         <p class = "text-justify">Finding the perfect cushion for
42             sense of style throughout your home. They can transfo
43             by adding colour, softness and an air of cosiness. As
44             our range of cushions will certainly embolden your de
45     </div>
46     {% endif %}
```

Add the following css code to **custom.css**:

```
.my_image_padding {
    padding-top: 5px;
}
```

Refresh the browser and when you view the top of the page, no major difference but some padding has been applied and all elements look ok.



Step 4: Change the display format for our products

Next, we want to change the display for our products so that the products are displayed in rows of 3 on the category page using the bootstrap grid system.

Open the **category.html** page and replace lines 47-53 with the following code (copy-paste available) that contains the new classes:

```
<div class="container">
  <div class="row mx-auto">
    {% for product in prods %}
      <div class="my_bottom_margin col-9 col-lg-4 col-sm-12 col-md-4">
        <div class="card text-center" style="min-width:18rem;">
          <a class="img" href="{{product.get_absolute_url}}"></a>
          <div class="card-body">
```

```
47 <div class="container">
48   <div class="row mx-auto">
49     {% for product in products %}
50       <div class = "my_bottom_margin col-9 col-lg-4 col-sm-12 col-md-4">
51         <div class = "card text-center" style="min-width:18rem;">
52           <a href="{{product.get_absolute_url}}"><img class = "card-img-top my_image" src=
53             <div class="card-body">
54               <h4>{{product.name}}</h4>
55               <p>€{{product.price}}</p>
56             </div>
57           </div>
58         </div>
59       {% endfor %}
60     </div>
61   <br>
62 </div>
```

Line 47: Container moves content away from the edges

Line 48: The row class will place the content in a row and mx-auto will center the content inside the row

Line 50: The my_bottom_margin class puts a bottom margin in for each row. The col-9 means the column width is 9 out of a possible 12. The other classes are to make the website mobile responsive – col-lg is for large desktops, col-sm is for phones, col-md is for medium devices such as tablets

Line 51: We use the bootstrap card class to display each product inside a card

Line 52: card-img-top places an image to the top of the card

Line 53: A custom class which has css code in custom.css

Copy-paste the following css code into **custom.css**:

```
.my_bottom_margin {  
    margin-bottom: 10px;  
}  
  
.card-body h4 {  
    font-size: 14px;  
    text-transform: uppercase;  
    letter-spacing: .2em;  
}
```

Refresh the page and you should see that the products are in rows of 3 and each product is in its own card & the web page is also mobile responsive (reduce the window size to see this).

NEW

EXCLUSIVE Cushion Collection


Now in store

OUR PRODUCTS COLLECTION


Finding the perfect cushion for your room can instantly add to the levels of comfort and sense of style throughout your home. They can transform the blandest of decors instantly by adding colour, softness and an air of cosiness. As well as being comfy and looking great our range of cushions will certainly embolden your décor with personalised charm.



BLACK URBAN
€100.00



DESPICABLE ME
€100.00



ELEPHANT
€50.00

Step 5: Styling the Product page with bootstrap & custom css

Now we will style the product page using bootstrap and custom css. First, we will add some classes to the breadcrumb navigation section of the **product.html** page.

Open the **product.html** page in VS Code and add the following classes to the divs just after block content:

```
9  {% block content %}
10  <div class = "row my_prod_row_class">
11  <div class = "mx-auto">
```

Copy-paste the following code to **custom.css**:

```
/*Product Page*/
.my_prod_row_class {
    padding-top: 15px;
    padding-bottom: 20px;
}
.my_prod_row_class .mx-auto p {
    color: #000;
    font-size: 12px;
}
.my_prod_row_class .mx-auto p a {
    color: #000;
    font-size: 12px;
    text-decoration: none;
}
```

Refresh the browser and you will see that the navigation crumb on the product page has changed slightly and has a bit more padding.

Next, we will work with the product image and product description.

Add the following classes to **product.html**:

```
11  <div class = "mx-auto">
12  <p><a href="{% url 'shop:all_products' %}">Home</a> | <a href="{{ product.c
13  </div>
14  <div class="container">
15  <br>
16  <div class="row">
17  <div class="col-12 col-sm-12 col-md-12 col-lg-6 text-center">
18  <div>
19  
20  </div>
21  </div>
```

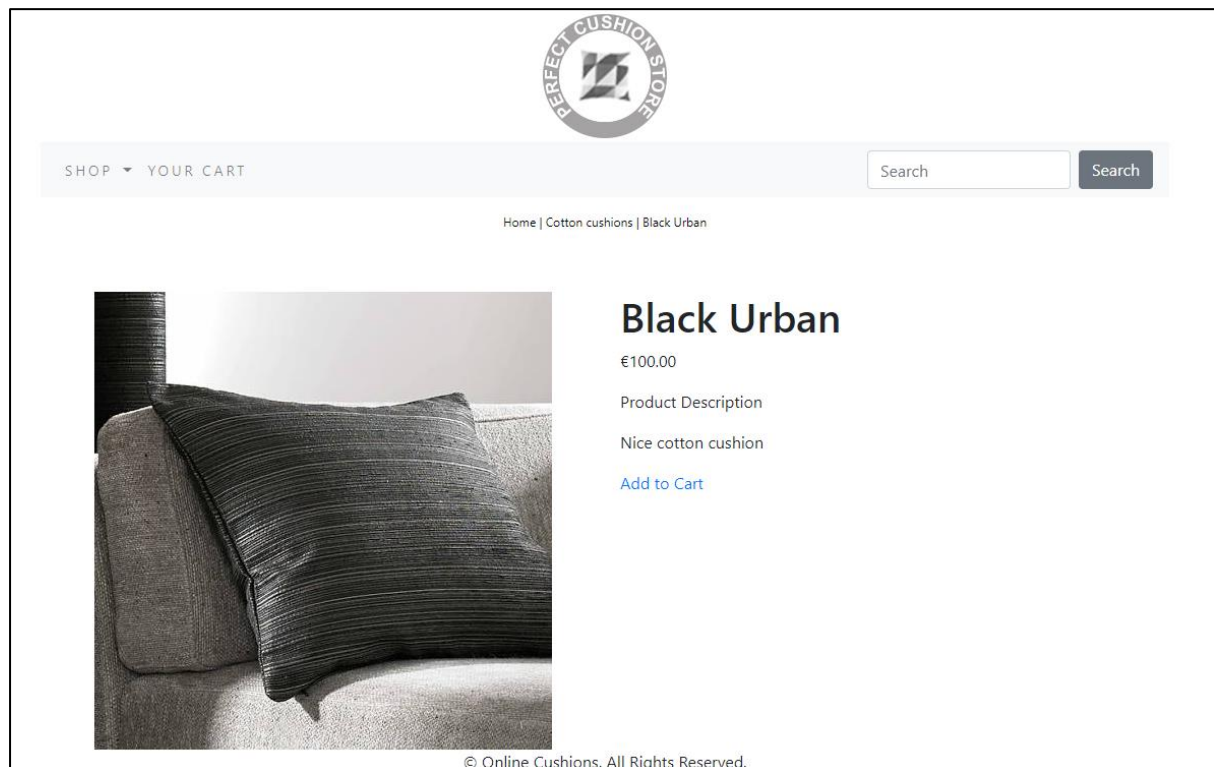
We will now add a new style to the div that contains the product image & we will also add a class for the product description main div

```

17 <div class= "col-12 col-sm-12 col-md-12 col-lg-6 text-center">
18   <div style="min-width: 18rem;">
19     
20   </div>
21 </div>
22 <div class = "col-12 col-sm-12 col-md-12 col-lg-6">
23   <div>
24     <h1>{{product.name}}</h1>
25     <p>€{{product.price}}</p>
26     <p>Product Description</p>
27     <p>{{product.description}}</p>
28     {% if product.stock == 0%}
29     <p><b>Out of Stock</b></p>
30     {% else %}
31     <a href="">Add to Cart</a>
32     {% endif %}
33   </div>
34 </div>
35 </div>
36 </div>
37 </div>
38 {% endblock %}

```

Refresh the browser and you should see that the product description is on the right-hand side of the image on the product page:



We will add new classes to the product details section of the template. Add the code for these new classes highlighted below:

```

22 |         <div class = "col-12 col-sm-12 col-md-12 col-lg-6">
23 |             <div>
24 |                 <h1 class="my_prod_title">{{product.name}}</h1>
25 |                 <p>€{{product.price}}</p>
26 |                 <p class = "my_title">Product Description</p>
27 |                 <p class = "text-justify my_prod_text">{{product.description}}</p>
28 |                 {% if product.stock == 0 %}
29 |                     <p class="text-justify my_prod_text"><b>>Out of Stock</b></p>
30 |                 {% else %}
31 |                     <a class = "btn btn-secondary" href="">Add to Cart</a>
32 |                 {% endif %}
33 |             </div>
34 |         </div>
35 |     </div>
36 | </div>
37 | </div>
38 | {% endblock %}

```

The text-justify for product description may not show any change unless the description is a few lines long. In order to see the Out of Stock text you will need to change the stock quantity of a product to 0 (this can be done in Django admin).

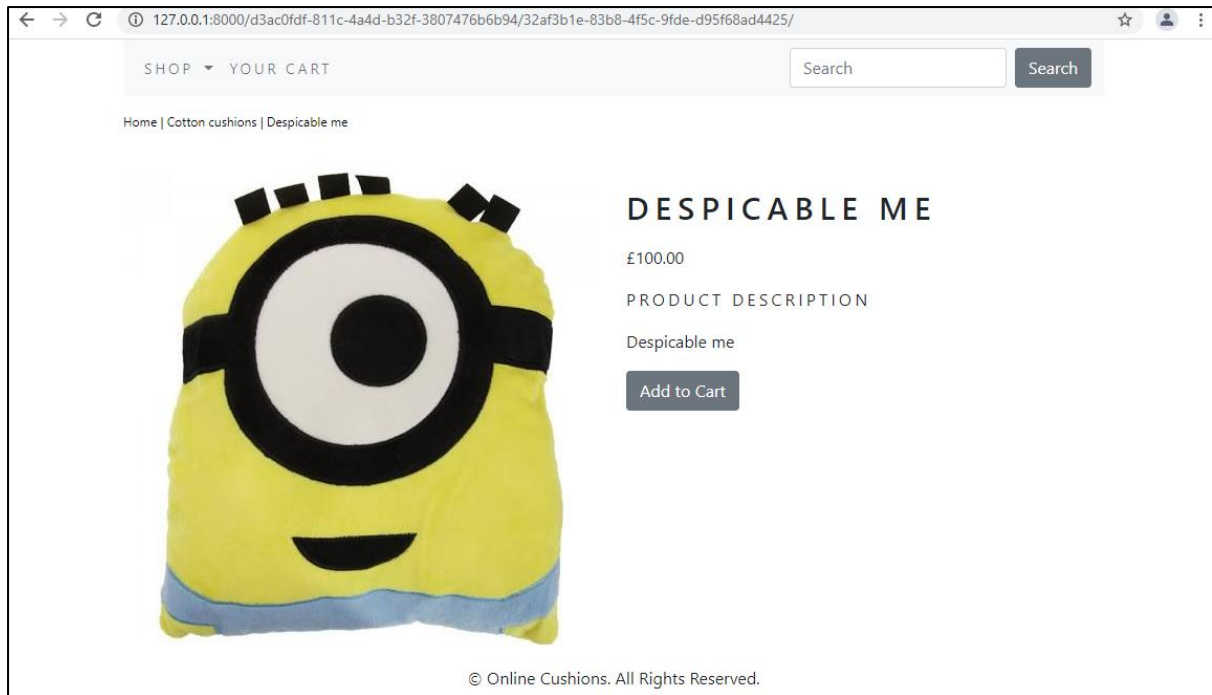
Copy-paste the following css to the **custom.css** file for the two custom classes my_prod_title and my_prod_text

```

.my_prod_title {
    font-size: 30px;
    text-transform: uppercase;
    letter-spacing: .2em;
    padding-top: 15px;
    padding-bottom: 10px;
}
.my_prod_text {
    padding-right: 20px;
}

```

Refresh the browser & you should see that the fonts have been applied and the bootstrap button styling has also been applied.



Step 6: Commit the changes and push them to the lab10 repo

Stop the server and run the following git commands to update the local and remote repositories:

git add -A

git commit -m "lab 10 part 3 commit"

git push -u origin main