**Account.html**

{% extends "app/layout.html" %}

{% block content %}

<h3 class="text-center">Личный кабинет пользователя **{{** request.user }}</h3>

<hr />

<h3 class="text-center">Ваши заказы</h3>

<table class="table">

<tr style="background:lightblue;">

<td class="text-center">Номер заказа</td>

<td class="text-center">Дата заказа</td>

<td class="text-center">Статус заказа</td>

<td class="text-center">Стоимость заказа</td>

<td class="text-center">Детали заказа</td>

</tr>

{% for item in order %}

<tr>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.id }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.date }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.status }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.total }} руб.</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;"><button type="button"

class="btn btn-primary btn-md" data-toggle="modal" data-target="#myModal-**{{** item.id }}">

Детали заказа</button>

</td>

</tr>

{% endfor %}

</table>

<!-- Button trigger modal -->

<!-- Modal -->

{% for item in order %}

<div class="modal fade" id="myModal-**{{** item.id }}" tabindex="-1" role="dialog" aria-labelledby="myModalLabel">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>

<h4 class="modal-title" id="myModalLabel">Заказ №**{{** item.id }}</h4>

</div>

<div class="modal-body">

<h3>Товар</h3>

<table class="table">

{% for order\_item in item.items.all %}

{% for ord\_item in order\_item.items.all %}

<tr>

<td>**{{** ord\_item.product.title}}</td>

<td>**{{** ord\_item.qty }} шт.</td>

<td>**{{** ord\_item.item\_total}} руб.</td>

</tr>

{% endfor %}

<tr>

<td></td>

<td>Итого</td>

<td><strong>**{{** order\_item.cart\_total}} руб.</strong></td>

</tr>

{% endfor %}

</table>

<h3>Заказчик</h3>

<p>Имя:**{{** item.first\_name }}</p>

<p>Фамилия:**{{** item.last\_name }}</p>

<p>Телефон:**{{** item.phone }}</p>

<h3>Тип доставки</h3>

<p>{% if item.buying\_type == 'self' %}Самовывоз{% else %}Доставка{% endif %}</p>

<p>{% if item.buying\_type == 'delivery' %}**{{** item.address }}{% endif %}</p>

<h3>Комментарий к заказу</h3>

<p>**{{** item.comments }}</p>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-default" data-dismiss="modal">Закрыть</button>

</div>

</div>

</div>

</div>

{% endfor %}

{% endblock %}

**Base.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="container">

<div class="row">

<div id="testCarousel" class="carousel slide" data-ride="carousel">

<!-- Слайды карусели -->

<div class="carousel-inner">

{% for product in products %}

{% if forloop.first %}

<!-- Слайд 1 -->

<div class="item active">

<img class="slide-image" src="**{{** product.image.url}}" alt="Alternate Text"

style="padding-left:260px;height:200px;" />

<p class="text-center"><a href="**{{** product.image.url}}">

<button class="btn btn-primary">

**{{** product.title}} </button></a></p>

</div>

{% else %}

<!-- Слайд 2 -->

<div class="item">

<img class="slide-image" src="**{{** product.image.url}}" alt="Alternate Text"

style="padding-left:260px;height:200px;" />

<p class="text-center"><a href="**{{** product.image.url}}">

<button class="btn btn-primary">

**{{** product.title}} </button></a></p>

</div>

{% endif %}

{% endfor %}

</div>

<!-- Навигация карусели (следующий или предыдущий слайд) -->

<!-- Кнопка, переход на предыдущий слайд с помощью атрибута data-slide="prev" -->

<a class="left carousel-control" href="#testCarousel"

role="button" data-slide="prev" style="background:transparent;" >

<span class="glyphicon glyphicon-chevron-left" style="color:black;" ></span>

</a>

<!-- Кнопка, переход на следующий слайд с помощью атрибута data-slide="next" -->

<a class="right carousel-control" href="#testCarousel"

role="button" data-slide="next" style="background:transparent;" >

<span class="glyphicon glyphicon-chevron-right" style="color:black;" ></span>

</a>

</div>

<div class="col-md-2">

<h3 >Категории</h3>

<div class="list-group">

{% for category in categories %}

<a href="**{{** category.get\_absolute\_url }}" class="list-group-item">

**{{** category.name }}</a>

{% endfor %}

</div>

</div>

</div>

<br />

<br />

<br />

<br />

<br />

<br />

<br />

<div class="row">

{% for product in products %}

<div class="col-sm-4 col-lg-4 col-md-4">

<div class="thumbnail">

<img src="**{{** product.image.url }}" alt="" style='width:70%;

height:200px;'/>

<div class="caption">

<h5 class="text-center"style="font-size:20px"><strong><a href="**{{** product.get\_absolute\_url }}">

**{{** product.title }}</a></strong></h5>

<h4 class="text-center" style="font-size:25px">

<strong>**{{** product.price }} руб.</strong></h4>

{% if request.user.is\_authenticated %}

<p class="text-center"><a href="#" class="add\_to\_cart"

data-slug="**{{**product.slug}}" ><button class="btn btn-danger">Добавить в корзину</button></a></p>

{% else %}

<small class="text-center">Заказывать товар могут только авторизированные пользователи.

Пожалуйста,<a href="{% url 'login' %}">зарегистрируйтесь</a>.

</small>

{% endif %}

</div>

</div>

</div>

{% endfor %}

</div>

</div>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script>

$(document).ready(function()

{

$('.add\_to\_cart').on('click', function (e)

{

e.preventDefault()

product\_slug = $(this).attr('data-slug')

data =

{

product\_slug:product\_slug

}

$.ajax({

type: "GET",

url: '{% url "add\_to\_cart" %}',

data: data,

success: function (data)

{

$("#cart\_count").html(data.cart\_total)

}

})

})

})

</script>

{% endblock %}

**Blog.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="row">

<h1>Новости интернет магазина " CONNECT electronics "</h1>

<br />

<br /><br /><br /><br /><br /><br />

{% for post in posts %}

{% if post.image and post.image.url != '/media/112.jpg' %}

<img src="**{{** post.image.url }}" alt="" />

{% endif %}

<h2><a href="{% url 'blogpost' parametr=post.id %}"> **{{**post.title}}</a> </h2>

<h4>**{{** post.description }} </h4>

<p> **{{**post.posted}} </p>

<hr/>

{% endfor %}

</div>

{% endblock %}

**Blogpost.html**

{% extends "app/layout.html" %}

{% load static %}

{% load bootstrap %}

{% block content %}

{% if user.is\_authenticated %}

{% if post\_1.image and post\_1.image.url != '/media/112.jpg' %}

<img src="**{{** post\_1.image.url }}" alt="" />

{% endif %}

<h2> **{{**post\_1.title}} </h2>

<p> **{{**post\_1.content|linebreaksbr}} </p> {# вывод текста по абзацам#}

<h3>Добавить комментарий</h3>

<form method="POST" >{% csrf\_token %} {# защита от атак CSRF#}

<div>

**{{**form|bootstrap}}

<button type="submit" class="save btn btn-danger" > Отправить

комментарий </button>

</div>

</form>

{% else %}

{% if post\_1.image and post\_1.image.url != '/media/112.jpg' %}

<img src="**{{** post\_1.image.url }}" alt="" />

{% endif %}

<h2> **{{**post\_1.title}} </h2>

<p> **{{**post\_1.content|linebreaksbr}} </p> {# вывод текста по абзацам#}

<h3> Для добавления своего комментария, пожалуйста, авторизуйтесь

</h3>

{% endif %}

{% for comment in comments %}

<hr/>

<h5> **{{**comment.date}} </h5>

<h5> **{{**comment.author}} </h5>

<p> **{{**comment.text|linebreaksbr}} </p>

{% endfor %}

{% endblock %}

**Contact.html**

<!--♕-->

{% extends "app/layout.html" %}

{% block content %}

<h2>**{{** title }}.</h2>

<h3>**{{** message }}</h3>

<address>

Город: Псков.<br />

Улица: пл. Ленина, дом 2. <br />

Индекс: 180000. <br />

<abbr title="Phone">Тел:</abbr>

+7(8112) 29-70-01,

<abbr title="Fax">Факс:</abbr>

+7(8112) 29-70-02

</address>

<address>

<strong>Поддержка:</strong> <a href="mailto:connect@outlook.com">connect@outlook.com</a><br />

<strong>Маркетинг:</strong> <a href="mailto:connect@outlook.com">connect@outlook.com</a>

</address>

<br />

<h3>**{{** Developer }}</h3>

<address>

Город: Псков.<br />

Улица: пл. Техническая, дом 14. <br />

Индекс: 180020. <br />

<abbr title="Phone">Тел:</abbr>

+7(953) 23-72-699,

</address>

<address>

<strong>Поддержка:</strong> <a href="mailto:kovalevskiy2017@outlook.com">kovalevskiy2017@outlook.com</a><br />

<strong>Маркетинг:</strong> <a href="mailto:kovalevskiy2017@outlook.com">kovalevskiy2017@outlook.com</a>

</address>

{% endblock %}

**Cart.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="row">

<h3 class="text-center">Ваша корзина</h3>

<br />

{% if cart.items.count > 0 %}

<div class="my-cart">

<table class="table">

<tr>

<td class="text-center"><strong>Товар</strong></td>

<td class="text-center" ><strong>Количество</strong></td>

<td class="text-center"><strong>Цена</strong></td>

<td class="text-center"></td>

</tr>

{% for item in cart.items.all %}

<tr class="cart-item-**{{** item.product.id}}">

<td class="text-center">**{{** item.product.title }}</td>

<td class="text-center">

<form action="/" method="get">

<input type="number" name="qty" class="cart-item-qty"

data-id="**{{** item.id }}" value="**{{** item.qty }}" min="1" style="width:60px;"/>

</form>

</td>

<td class="text-center" id="cart-item-total-**{{** item.id }}">**{{** item.item\_total }} руб.</td>

<td class="text-center"> <a href="#"

class="remove\_from\_cart" data-slug="**{{** item.product.slug }}"

data-id="**{{** item.product.id }}">

<button class="btn btn-default">Удалить из корзины</button></a></td>

</tr>

{% endfor %}

<tr>

<td></td>

<td class="text-center"><strong>Итого :</strong></td>

<td class="text-center" id="cart-total-price"><strong>**{{** cart.cart\_total}} руб.</strong></td>

<td class="text-center"></td>

</tr>

</table>

<br />

<a href="{% url 'checkout' %}" class="pull-right"><button class="btn btn-danger">Предварительный заказ</button></a>

</div>

<h3 class="cart-empty"></h3>

{% else %}

<h3 class="text-center">Ваша корзина пуста</h3>

{% endif %}

</div>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script>

$(document).ready(function()

{

$('.cart-empty').css('display','none')

$('.remove\_from\_cart').on('click', function (e)

{

e.preventDefault()

product\_slug = $(this).attr('data-slug')

item\_product\_id=$(this).attr('data-id')

data =

{

product\_slug:product\_slug

}

$.ajax({

type: "GET",

url: '{% url "remove\_from\_cart" %}',

data: data,

success: function (data)

{

$("#cart\_count").html(data.cart\_total)

$('.cart-item-'+item\_product\_id).css('display','none')

$('#cart-total-price').html('<strong>'+parseFloat(data.cart\_total\_price).toFixed(2)+'руб.</strong>')

if(parseInt(data.cart\_total) == 0)

{

$('.my-cart').css('display','none')

$('.cart-empty').css('display','block')

$('.cart-empty').html('<h3 class="text-center">Ваша корзина пуста</h3>')

}

}

})

})

})

$(document).ready(function ()

{

$('.cart-item-qty').on('click', function ()

{

qty=$(this).val()

item\_id=$(this).attr('data-id')

data =

{

qty:qty,

item\_id:item\_id

}

$.ajax({

type: "GET",

url: '{% url "change\_item\_qty" %}',

data: data,

success: function(data)

{

$('#cart-item-total-'+ item\_id).html(parseFloat(data.item\_total).toFixed(2)+'руб.')

$('#cart-total-price').html('<strong>'+ parseFloat(data.cart\_total\_price).toFixed(2)+'руб.</strong>')

}

})

})

})

</script>

{% endblock %}

**Category.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="container">

<div class="row">

<div class="col-sm-12">

<h2>**{{** category.name}}</h2>

<hr />

{% for product in products\_of\_category%}

<div class="col-sm-4">

<img src="**{{** product.image.url }}" alt="" style='width:70%; height:250px;'/>

<h4 class="text-center">**{{** product.title }}</h4>

<p class="text-center" style="font-size:25px">Цена :<strong >**{{** product.price }}</strong></p>

{% if request.user.is\_authenticated %}

<p class="text-center"><a href="#" class="add\_to\_cart" data-slug="**{{** product.slug}}">

<button class="btn btn-danger">Добавить в корзину</button></a></p>

{% else %}

<p class="text-center">

Заказывать товар могут только авторизированные пользователи.

Пожалуйста, <a href="{% url 'login' %}">зарегистрируйтесь</a>.

</p>

{% endif %}

</div>

{% endfor %}

</div>

</div>

</div>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script>

$(document).ready(function()

{

$('.add\_to\_cart').on('click', function (e)

{

e.preventDefault()

product\_slug = $(this).attr('data-slug')

data =

{

product\_slug:product\_slug

}

$.ajax({

type: "GET",

url: '{% url "add\_to\_cart" %}',

data: data,

success: function (data)

{

$("#cart\_count").html(data.cart\_total)

}

})

})

})

</script>

{% endblock %}

**Checkout.html**

{% extends "app/layout.html" %}

{% block content %}

<h3 class="text-center">Ваш предварительный заказ</h3>

<br />

<div class="row">

<table class="table">

<tr>

<td class="text-center"><strong>Товар</strong></td>

<td class="text-center" ><strong>Количество</strong></td>

<td class="text-center"><strong>Цена</strong></td>

<td class="text-center"></td>

</tr>

{% for item in cart.items.all %}

<tr>

<td class="text-center">**{{** item.product.title }}</td>

<td class="text-center">**{{** item.qty }}</td>

<td class="text-center">**{{** item.item\_total }} руб.</td>

<td > </td>

</tr>

{% endfor %}

<tr>

<td></td>

<td class="text-center"><strong>Итого :</strong></td>

<td class="text-center"><strong>**{{** cart.cart\_total}} руб.</strong></td>

<td class="text-center"></td>

</tr>

</table>

<a href="{% url "order\_create" %}" class="pull-right"><button class="btn btn-success">Оформить заказ</button></a>

</div>

{% endblock %}

**Gallery.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="row">

<div class="gallery">

<figure>

<img src="../../static/app/photogallery/magazin1.png" alt="" />

<figcaption>Daytona Beach <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/magazin2.png" alt="" />

<figcaption>Териберка, gorod Severomorsk <small>Russia</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/magazin3.png" alt="" />

<figcaption> Bad Pyrmont <small>Deutschland</small> </figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/magazin4.png" alt="" />

<figcaption>Yellowstone National Park <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/magazin5.png" alt="" />

<figcaption>Quiraing, Portree <small>United Kingdom</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/magazin6.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad1.png" alt="" />

<figcaption>Daytona Beach <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad2.png" alt="" />

<figcaption>Териберка, gorod Severomorsk <small>Russia</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad3.png" alt="" />

<figcaption> Bad Pyrmont <small>Deutschland</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad4.png" alt="" />

<figcaption>Yellowstone National Park <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad5.png" alt="" />

<figcaption>Quiraing, Portree <small>United Kingdom</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad6.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad7.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad8.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad9.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad10.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad11.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad12.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

<figure>

<img src="../../static/app/photogallery/sclad13.png" alt="" />

<figcaption>Highlands <small>United States</small></figcaption>

</figure>

</div>

</div>

<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" style="display:none;">

<symbol id="close" viewBox="0 0 18 18">

<path fill-rule="evenodd" clip-rule="evenodd" fill="#FFFFFF" d="M9,0.493C4.302,0.493,0.493,4.302,0.493,9S4.302,17.507,9,17.507

S17.507,13.698,17.507,9S13.698,0.493,9,0.493z M12.491,11.491c0.292,0.296,0.292,0.773,0,1.068c-0.293,0.295-0.767,0.295-1.059,0

l-2.435-2.457L6.564,12.56c-0.292,0.295-0.766,0.295-1.058,0c-0.292-0.295-0.292-0.772,0-1.068L7.94,9.035L5.435,6.507

c-0.292-0.295-0.292-0.773,0-1.068c0.293-0.295,0.766-0.295,1.059,0l2.504,2.528l2.505-2.528c0.292-0.295,0.767-0.295,1.059,0

s0.292,0.773,0,1.068l-2.505,2.528L12.491,11.491z"/>

</symbol>

</svg>

<script>

popup = {

init: function(){

$('figure').click(function(){

popup.open($(this));

});

$(document).on('click', '.popup img', function(){

return false;

}).on('click', '.popup', function(){

popup.close();

})

},

open: function($figure) {

$('.gallery').addClass('pop');

$popup = $('<div class="popup" />').appendTo($('body'));

$fig = $figure.clone().appendTo($('.popup'));

$bg = $('<div class="bg" />').appendTo($('.popup'));

$close = $('<div class="close"><svg><use xlink:href="#close"></use></svg></div>').appendTo($fig);

$shadow = $('<div class="shadow" />').appendTo($fig);

src = $('img', $fig).attr('src');

$shadow.css({backgroundImage: 'url(' + src + ')'});

$bg.css({backgroundImage: 'url(' + src + ')'});

setTimeout(function(){

$('.popup').addClass('pop');

}, 10);

},

close: function(){

$('.gallery, .popup').removeClass('pop');

setTimeout(function(){

$('.popup').remove()

}, 100);

}

}

popup.init()

</script>

{% endblock %}

**Index.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<div class="row">

<!-- The carousel -->

<div id="transition-timer-carousel" class="carousel slide transition-timer-carousel" data-ride="carousel">

<!-- Indicators -->

<ol class="carousel-indicators">

<li data-target="#transition-timer-carousel" data-slide-to="0" class="active"></li>

<li data-target="#transition-timer-carousel" data-slide-to="1"></li>

<li data-target="#transition-timer-carousel" data-slide-to="2"></li>

<li data-target="#transition-timer-carousel" data-slide-to="3"></li>

<li data-target="#transition-timer-carousel" data-slide-to="4"></li>

</ol>

<!-- Wrapper for slides -->

<div class="carousel-inner">

{% for carusel in carusels %}

{% if forloop.first %}

<div class="item active">

<img src="**{{** carusel.image.url}}" />

<div class="carousel-caption">

<h1 class="carousel-caption-header"></h1>

<p class="carousel-caption-text hidden-sm hidden-xs"> </p>

</div>

</div>

{% else %}

<div class="item">

<img src="**{{** carusel.image.url}}" />

<div class="carousel-caption">

<h1 class="carousel-caption-header"></h1>

<p class="carousel-caption-text hidden-sm hidden-xs"></p>

</div>

</div>

{% endif %}

{% endfor %}

</div>

<!-- Controls -->

<a class="left carousel-control" href="#transition-timer-carousel" data-slide="prev">

<span class="glyphicon glyphicon-chevron-left"></span>

</a>

<a class="right carousel-control" href="#transition-timer-carousel" data-slide="next">

<span class="glyphicon glyphicon-chevron-right"></span>

</a>

<!-- Timer "progress bar" -->

<hr class="transition-timer-carousel-progress-bar animate" />

</div>

</div>

<div class="row">

<div class="col-md-4">

<h2>Getting started</h2>

<p>

Django gives you a powerful, patterns-based way to build dynamic websites that

enables a clean separation of concerns and gives you full control over markup

for enjoyable, agile development.

</p>

<p><a class="btn btn-default" href="http://www.djangobook.com/en/2.0/index.html">Learn more &raquo;</a></p>

</div>

<div class="col-md-4">

<h2>Get more libraries</h2>

<p>The Python Package Index is a repository of software for the Python programming language.</p>

<p><a class="btn btn-default" href="https://pypi.org/">Learn more &raquo;</a></p>

</div>

<div class="col-md-4">

<h2>Microsoft Azure</h2>

<p>You can easily publish to Microsoft Azure using Visual Studio. Find out how you can host your application using a free trial today.</p>

<p><a class="btn btn-default" href="http://azure.microsoft.com">Learn more &raquo;</a></p>

</div>

</div>

{% endblock %}

**Layout.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>**{{** title }} - My Django Application</title>

{% load staticfiles %}

<link rel="stylesheet" type="text/css" href="{% static 'app/content/bootstrap.min.css' %}" />

<link rel="stylesheet" type="text/css" href="{% static 'app/content/site.css' %}" />

<link rel="stylesheet" type="text/css" href="{% static 'app/content/bootstrap.css' %}" />

<script src="{% static 'app/scripts/bootstrap.js' %}"></script>

<script src="{% static 'app/scripts/modernizr-2.6.2.js' %}"></script>

<!--Для карусели index.html-->

<link href="//netdna.bootstrapcdn.com/bootstrap/3.1.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//netdna.bootstrapcdn.com/bootstrap/3.1.0/js/bootstrap.min.js"></script>

<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>

<!--Для галереи gallary.html-->

<link href="//netdna.bootstrapcdn.com/bootstrap/3.0.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//netdna.bootstrapcdn.com/bootstrap/3.0.0/js/bootstrap.min.js"></script>

<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>

<!--Для под меню -->

<link href="//netdna.bootstrapcdn.com/font-awesome/3.2.1/css/font-awesome.css" rel="stylesheet">

</head>

<body>

{% if user.is\_authenticated %}

<div id="wrapper">

<!-- Sidebar -->

<div id="sidebar-wrapper">

<ul class="sidebar-nav" style="margin-left:0;">

<li class="sidebar-brand">

<a href="#menu-toggle" id="menu-toggle" style="margin-top:20px;float:right;" > <i class="fa fa-bars " style="font-size:20px !Important;" aria-hidden="true" ></i> </a>

</li>

{% if not user.is\_staff %}{% if not user.is\_superuser %}

<li>

<p style="background-color:gold;" > <span style="margin-left:60px;">Мой профиль</span> </p>

</li>

<li>

<a href="{% url 'account' %}"> <span style="margin-left:60px;">Мои заказы</span> </a>

</li>

<li>

<a href="#"> <span style="margin-left:60px;">Спец.предложения</span> </a>

</li>

{% endif %}{% endif %}

{% if user.is\_staff %}{% if not user.is\_superuser %}

<li>

<p style="background-color:gold;" > <span style="margin-left:60px;">Менеджер</span> </p>

</li>

<li>

<a href="{% url 'my\_order' %}"> <span style="margin-left:60px;">Заказы</span> </a>

</li>

{% endif %}{% endif %}

{% if user.is\_superuser %}

<li>

<p style="background-color:gold;" > <span style="margin-left:40px;">Администратор</span> </p>

</li>

<li>

<a href="{% url 'my\_order' %}"> <span style="margin-left:40px;">Заказы</span> </a>

</li>

<li>

<a href="{% url 'newpost' %}"> <span style="margin-left:40px;">Добавить статью</span> </a>

</li>

{% endif %}

</ul>

</div>

{% endif %}

<!-- /#sidebar-wrapper -->

<!-- Page Content -->

<div id="page-content-wrapper">

<div class="container-fluid">

<div class="row">

<div class="col-lg-12">

<!-- /#page-content-wrapper -->

<!-- /#wrapper -->

<script>

$("#menu-toggle").click(function(e) {

e.preventDefault();

$("#wrapper").toggleClass("toggled");

});

</script>

</div>

</div>

</div>

</div>

</div>

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<!-- <a href="/" class="navbar-brand">name</a>-->

<a href="/" class="navbar-left"> <img src=" static/app/content/logo.ico" width="150" height="70" alt="logo" /> </a>

</div>

<div class="navbar-collapse collapse">

<ul class="nav navbar-nav">

<li><a href="{% url 'home' %}">Главная</a></li>

<li><a href="{% url 'about' %}">О компании</a></li>

<li><a href="{% url 'blog' %}">Новости</a></li>

<li><a href="{% url 'gallery' %}">Галерея</a></li>

<li><a href="{% url 'contact' %}">Контакты</a></li>

{% if user.is\_authenticated %} {% if not user.is\_staff %}{% if not user.is\_superuser %}

<li> <a href="{% url 'account' %}">Мои заказы</a></li>

{% endif %}{% endif %}{% endif %}

<li><a href="{% url 'base\_view' %}">Каталог товаров</a></li>

{% if not user.is\_authenticated %}

<li><a href="{% url 'registration' %}">Регистрация</a></li>

{% endif %}

</ul>

{% include 'app/loginpartial.html' %}

</div>

</div>

</div>

<div class="container body-content">

{% block content %}{% endblock %}

<hr/>

<footer>

<p>&copy; **{{** year }} - My Django Application</p>

</footer>

</div>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script src="{% static 'app/scripts/bootstrap.js' %}"></script>

<script src="{% static 'app/scripts/respond.js' %}"></script>

{% block scripts %}{% endblock %}

</body>

</html>

**Login.html**

<!--♕-->

{% extends "app/layout.html" %}

{% block content %}

<h2>**{{** title }}</h2>

<div class="row">

<div class="col-md-8">

<section id="loginForm">

<form action="." method="post" class="form-horizontal">

{% csrf\_token %}

<h4>Используйте локальную учетную запись для входа.</h4>

<hr />

<div class="form-group">

<label for="id\_username" class="col-md-2 control-label">Имя пользователя</label>

<div class="col-md-10">

**{{** form.username }}

</div>

</div>

<div class="form-group">

<label for="id\_password" class="col-md-2 control-label">Пароль</label>

<div class="col-md-10">

**{{** form.password }}

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="hidden" name="next" value="/" />

<input type="submit" value="Авторизоваться" class="btn btn-primary" />

</div>

</div>

{% if form.errors %}

<p class="validation-summary-errors">Пожалуйста, введите правильное имя пользователя и пароль.</p>

{% endif %}

</form>

</section>

</div>

<div class="col-md-4">

<section id="socialLoginForm"></section>

</div>

</div>

{% endblock %}

{% block scripts %}

{% load staticfiles %}

<script src="{% static 'app/scripts/jquery.validate.min.js' %}"></script>

{% endblock %}

**Loginpartial.html**

<!--♕-->

{% if user.is\_authenticated %}

<form id="logoutForm" action="/logout" method="post" class="navbar-right">

{% csrf\_token %}

{% if user.is\_authenticated %} {% if not user.is\_staff %}{% if not user.is\_superuser %}

<ul class="nav navbar-nav navbar-right">

<li><a href="{% url 'cart' %}">Корзина

<span class="label label-default"> <span class="bange" id="cart\_count">

{% if not request.session.total %}

0 {% else %}**{{** request.session.total }} {% endif %}</span></span>

<span class="label label-default"></span></a></li>

</ul>

{% endif %} {% endif %} {% endif %}

<ul class="nav navbar-nav navbar-right">

<li><span class="navbar-brand">Hello **{{** user.username }}!</span></li>

<li><a href="javascript:document.getElementById('logoutForm').submit()">Выйти</a></li>

</ul>

</form>

{% else %}

<ul class="nav navbar-nav navbar-right">

<li><a href="{% url 'login' %}">Вход на сайт</a></li>

</ul>

{% endif %}

**My\_order.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<h3 class="text-center">Личный кабинет пользователя **{{** request.user }}</h3>

<hr/>

<h3 class="text-center">Заказы</h3>

<table class="table">

<tr style="background:lightblue;">

<td class="text-center">Заказчики</td>

<td class="text-center">Номер заказа</td>

<td class="text-center">Дата заказа</td>

<td class="text-center">Статус заказа</td>

<td class="text-center">Стоимость заказа</td>

<td class="text-center">Детали заказа</td>

</tr>

{% for item in order %}

<tr>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.user }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.id }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.date }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.status }}</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;">**{{** item.total }} руб.</td>

<td class="text-center" style="padding-top:10px;padding-bottom:20px;"><button type="button"

class="btn btn-primary btn-md" data-toggle="modal" data-target="#myModal-**{{** item.id }}">

Детали заказа</button> </td>

</tr>

{% endfor %}

</table>

<!-- Button trigger modal -->

<!-- Modal -->

{% for item in order %}

<div class="modal fade" id="myModal-**{{** item.id }}" tabindex="-1" role="dialog" aria-labelledby="myModalLabel">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>

<h4 class="modal-title" id="myModalLabel">Заказ №**{{** item.id }}</h4>

</div>

<div class="modal-body">

<h3>Товар</h3>

<table class="table">

{% for order\_item in item.items.all %}

{% for ord\_item in order\_item.items.all %}

<tr>

<td>**{{** ord\_item.product.title}}</td>

<td>**{{** ord\_item.qty }} шт.</td>

<td>**{{** ord\_item.item\_total}} руб.</td>

</tr>

{% endfor %}

<tr>

<td></td>

<td>Итого</td>

<td><strong>**{{** order\_item.cart\_total}} руб.</strong></td>

</tr>

{% endfor %}

</table>

<h3>Заказчик</h3>

<p>Имя:**{{** item.first\_name }}</p>

<p>Фамилия:**{{** item.last\_name }}</p>

<p>Телефон:**{{** item.phone }}</p>

<h3>Тип доставки</h3>

<p>{% if item.buying\_type == 'self' %}Самовывоз{% else %}Доставка{% endif %}</p>

<p>{% if item.buying\_type == 'delivery' %}**{{** item.address }}{% endif %}</p>

<h3>Комментарий к заказу</h3>

<p>**{{** item.comments }}</p>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-default" data-dismiss="modal">Закрыть</button>

</div>

</div>

</div>

</div>

{% endfor %}

{% endblock %}

**Newpost.html**

{% extends "app/layout.html" %}

{% load static %}

{% load bootstrap %}

{% block content %}

<form method="POST" enctype="multipart/form-data">{% csrf\_token %} {# защита от атак CSRF#}

<div>

**{{**form|bootstrap}}

<button type="submit" class="save btn btn-primary" > Разместить статью </button>

</div>

</form>

{% endblock %}

**Order.html**

{% extends "app/layout.html" %}

{% load static %}

{% load bootstrap %}

{% block content %}

<form action="{% url "make\_order" %}" method="post"enctype="multipart/form-data">

{% csrf\_token %}

<div class="col-sm-4">**{{** form.name|bootstrap}}</div>

<div class="col-sm-4">**{{** form.last\_name|bootstrap}}</div>

<div class="col-sm-4">**{{** form.phone|bootstrap}}</div>

**{{** form.buying\_type|bootstrap}}

**{{** form.address|bootstrap}}

**{{** form.date|bootstrap}}

**{{** form.comments|bootstrap}}

<input type="submit" class="btn btn-success" value="Оформить заказ" />

</form>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script>

$(document).ready(function () {

$('#id\_address').css('display','none')

$('#id\_buying\_type').on('click', function () {

buying\_type = $(this).val()

if (buying\_type == 'delivery') {

$('#id\_address').css('display', 'block')

}

else {

$('#id\_address').css('display', 'none')

}

})

$('#id\_date\_month').css('width','100px')

$('#id\_date\_month').css('display','inline')

$('#id\_date\_year').css('width', '100px')

$('#id\_date\_year').css('display','inline')

$('#id\_date\_day').css('width', '70px')

$('#id\_date\_day').css('display','inline')

})

</script>

{% endblock %}

**Product.html**

{% extends "app/layout.html" %}

{% load static %}

{% block content %}

<br />

<div class="container">

<div class="row">

<div class="col-sm-4">

<img src="**{{** product.image.url }}" alt="" style='width:70%; height:200px;'/>

</div>

<div class="col-sm-8">

<h3 class="text-center">**{{** product.title }} </h3>

<p>**{{** product.description}}</p>

<hr />

<p style="font-size:25px">Цена :<strong >**{{** product.price }} руб.</strong></p>

{% if request.user.is\_authenticated %}

<a href="#"

class="add\_to\_cart" data-slug="**{{** product.slug}}">

<button class="btn btn-danger">Добавить в корзину</button></a>

{% else %}

<p class="text-center">

Заказывать товар могут только авторизированные пользователи.

Пожалуйста, <a href="{% url 'login' %}">зарегистрируйтесь</a>.

</p>

{% endif %}

</div>

</div>

</div>

<script src="{% static 'app/scripts/jquery-1.10.2.js' %}"></script>

<script>

$(document).ready(function()

{

$('.add\_to\_cart').on('click', function (e)

{

e.preventDefault()

product\_slug = $(this).attr('data-slug')

data =

{

product\_slug:product\_slug

}

$.ajax({

type: "GET",

url: '{% url "add\_to\_cart" %}',

data: data,

success: function (data)

{

$("#cart\_count").html(data.cart\_total)

}

})

})

})

</script>

{% endblock %}

**Registration.html**

<!--♕-->

{% extends "app/layout.html" %}

{% load bootstrap %}

{% block content %}

<div class="row">

<h2 class="text-center">Регистрация пользователя</h2>

<form method="POST" enctype="multipart/form-data">

{% csrf\_token %}

<div class="col-sm-8 col-sm-offset-2">

**{{** regform|bootstrap }}

<p class="text-center"><button type="submit" class="save btn btn-primary" > Зарегистрировать</button></p>

</div>

</form>

</div>

{% endblock %}

**Thank\_you.html**

{% extends "app/layout.html" %}

{% block content %}

<div class="row">

<div class="col-sm-8 col-sm-offset-2">

<h3 class="text-center">Спасибо за Ваш заказ!

Он принят в обработку.В ближайшее время менеджер с Вами свяжется.</h3>

<p class="text-center"><a href="{% url 'base\_view'%}">

<button class="btn btn-default">Вернуться к покупкам</button></a>

</p>

</div>

</div>

{% endblock %}

**Form.py**

"""

Definition of forms.

"""

from .models import Blog

from.models import Comment

from django.contrib.auth.models import User

from django.utils import timezone

from django import forms

from django.contrib.auth.forms import AuthenticationForm

from django.utils.translation import ugettext\_lazy as \_

class BootstrapAuthenticationForm(AuthenticationForm):

"""Authentication form which uses boostrap CSS."""

username = forms.CharField(max\_length=254,

widget=forms.TextInput({

'class': 'form-control',

'placeholder': 'User name'}))

password = forms.CharField(label=\_("Password"),

widget=forms.PasswordInput({

'class': 'form-control',

'placeholder':'Password'}))

class OrderForm(forms.Form):

name=forms.CharField()

last\_name=forms.CharField(required=False)

phone=forms.CharField()

buying\_type=forms.ChoiceField(widget=forms.Select(),

choices=([("self","Самовывоз"),("delivery","Доставка")]))

date=forms.DateField(widget=forms.SelectDateWidget(),initial=timezone.now())

address=forms.CharField(required=False)

comments=forms.CharField(widget=forms.Textarea,required=False)

def \_\_init\_\_(self,\*args,\*\*kwargs):

super(OrderForm,self).\_\_init\_\_(\*args,\*\*kwargs)

self.fields['name'].label='Имя'

self.fields['last\_name'].label='Фамилия'

self.fields['phone'].label='Контактный телефон'

self.fields['phone'].help\_text='Пожалуйста, указывайте реальный номер телефона, по которому с Вами можно связаться'

self.fields['buying\_type'].label='Способ получения'

self.fields['address'].label='Адрес доставки'

self.fields['address'].help\_text='\*Обязательно указывайте город!'

self.fields['comments'].label='Комментарий к заказу'

self.fields['date'].label='Дата доставки'

self.fields['date'].help\_text='Доставка производится на следующий день после оформления заказа. Менеджер с Вами предварительно свяжется.'

class CommentForm (forms.ModelForm):

class Meta:

model = Comment # используемая модель

fields = ('text',) # требуется заполнить только поле text

labels = {'text': "Комментарий"} # метка к полю формы text

class BlogForm (forms.ModelForm):

class Meta:

model = Blog # используемая модель

fields = ('image','title','description','content','posted',) # требуется заполнить только поле text

**models.py**

"""

Definition of models.

"""

from datetime import datetime

from django.contrib.auth.models import User #для импорта автора комментариев

from django.core.urlresolvers import reverse

import site

from decimal import Decimal

from transliterate import translit # переводит кириллицу

from django.utils.text import slugify #превращает поле в тип slug

from django.db.models.signals import pre\_save

from django.db import models #В этом модуле объявлен базовый класс модели Model и классы полей, предназначенных для хранения данных различных типов.

from django.contrib import admin #добавили использование административного модуля

from django.conf import settings #это объект.Файл настроек Django содержит полную конфигурацию установленного проект

# Create your models here.

class Category(models.Model):

name=models.CharField(max\_length=120)

slug=models.SlugField(blank=True)

def \_\_str\_\_(self):

return self.name

def get\_absolute\_url(self):

return reverse ('category\_detail',kwargs={'category\_slug':self.slug})

def pre\_save\_category\_slug(sender,instance,\*args,\*\*kwargs):

if not instance.slug:

slug=slugify(translit(instance.name,reversed=True))# заполняет латиницей slug

instance.slug=slug

pre\_save.connect(pre\_save\_category\_slug,sender=Category)

admin.site.register(Category)

class Brand(models.Model):

name=models.CharField(max\_length=120)

def \_\_str\_\_(self):

return self.name

admin.site.register(Brand)

#Чтобы корректно сохранять изображения

def image\_folder(instance,filename):

filename=instance.slug + '.'+ filename.split('.')[1]

return "{0}/{1}".format(instance.slug,filename)

#Чтобы при снятие галочки в admin ->available изчезал продукт с каталога

class ProductManager(models.Manager):

def all(self, \*args, \*\*kwargs):

return super(ProductManager,self).get\_queryset().filter(available=True)

class Product(models.Model):

category=models.ForeignKey(Category)

brand=models.ForeignKey(Brand)

title=models.CharField(max\_length=120)

slug=models.SlugField()

description=models.TextField()

image=models.ImageField(upload\_to=image\_folder)

price=models.DecimalField(max\_digits=9,decimal\_places=2)

available=models.BooleanField(default=True)

objects=ProductManager()

def \_\_str\_\_(self):

return self.title

def get\_absolute\_url(self):

return reverse ('product\_detail',kwargs={'product\_slug':self.slug})

admin.site.register(Product)

# Создаем корзину

#модель продукта

class CartItem(models.Model):

product=models.ForeignKey(Product)#продукт

qty=models.PositiveIntegerField(default=1)#кол-во yне может быть отрицательным поэтому PositiveIntegerField

item\_total=models.DecimalField(max\_digits=9, decimal\_places=2, default=0.0)#для изменения в корзине кол-во

def \_\_str\_\_(self):

return "Cart item for product {0}". format(self.product.title)

admin.site.register(CartItem)

#модель корзины

class Cart(models.Model):

items=models.ManyToManyField(CartItem,blank=True)#Мы можем добавить много в корзину CartItem

cart\_total=models.DecimalField(max\_digits=9, decimal\_places=2, default=0.0)

def \_\_str\_\_(self):

return str(self.id)# возращает id корзины

def add\_to\_cart(self,product\_slug):

cart=self

product=Product.objects.get(slug=product\_slug)

new\_item, \_=CartItem.objects.get\_or\_create(product=product,item\_total=product.price)

if new\_item not in cart.items.all():

cart.items.add(new\_item)

cart.save()

def remove\_from\_cart(self,product\_slug):

cart=self

product=Product.objects.get(slug=product\_slug)

for cart\_item in cart.items.all():

if cart\_item.product == product:

cart.items.remove(cart\_item)

cart.save()

def change\_qty(self,qty,item\_id):

cart=self

cart\_item=CartItem.objects.get(id=int(item\_id))

cart\_item.qty=int(qty)

cart\_item.item\_total = int(qty) \* Decimal(cart\_item.product.price)

cart\_item.save()

new\_cart\_total=0.00

for item in cart.items.all():

new\_cart\_total+=float(item.item\_total)

cart.cart\_total=new\_cart\_total

cart.save()

ORDER\_STATUS\_CHOICES=(('Принят в обработку','Принят в обработку'),

('Выполняется','Выполняется'),

('Оплачен','Оплачен')

)

admin.site.register(Cart)

class Order(models.Model):

user=models.ForeignKey(settings.AUTH\_USER\_MODEL)

items=models.ManyToManyField(Cart)

total=models.DecimalField(max\_digits=9, decimal\_places=2,default=0.00)

first\_name=models.CharField(max\_length=200)

last\_name=models.CharField(max\_length=200)

phone=models.CharField(max\_length=20)

address=models.CharField(max\_length=255)

buying\_type=models.CharField(max\_length=40, choices=(('Самовывоз','Самовывоз'),('Доставка','Доставка')),default='Самовывоз')

date=models.DateTimeField(auto\_now\_add=True)

comments=models.TextField()

status=models.CharField(max\_length=100,choices=ORDER\_STATUS\_CHOICES,

default=ORDER\_STATUS\_CHOICES[0][0])

def \_\_str\_\_(self):

return "Заказ №{0}".format(str(self.id))

admin.site.register(Order)

class CaruselInfo(models.Model):

title=models.CharField(max\_length=255)

content = models.TextField()

image=models.ImageField(upload\_to=image\_folder)

slug=models.SlugField()#Slug – газетный термин. “Slug” – это короткое название-метка, которое содержит только буквы, числа, подчеркивание или дефис. В основном используются в URL.

available=models.BooleanField(default=True)

objects=ProductManager()

def \_\_str\_\_(self):

return self.title

def get\_absolute\_url(self):

return reverse ('caruselInfo\_detail',kwargs={'caruselInfo\_slug':self.slug})

admin.site.register(CaruselInfo)

# Модель данных Блога

class Blog(models.Model):

title = models.CharField(max\_length = 100, unique\_for\_date = "posted", verbose\_name = "Заголовок")

description = models.TextField (verbose\_name = "Краткое содержание")

content = models.TextField (verbose\_name = "Полное содержание")

posted = models.DateTimeField (default = datetime.now(), db\_index = True,verbose\_name = "Опубликована")

image = models.FileField(default = '112.jpg', verbose\_name = "Путь к картинке")

def get\_absolute\_url(self): # метод возвращает строку с уникальным интернет-адресом записи (для просмотра записи на сайте)

return reverse("blogpost", args=[str(self.id)])

def \_\_str\_\_(self): # метод возвращает название, используемое для представления отдельных записей в административном разделе

return self.title

class Meta: # метаданные — вложенный класс, который задает дополнительные параметры модели:

db\_table = "Posts" # имя таблицы для модели

ordering = ["-posted"] # порядок сортировки данных в модели ("–"означает по убыванию)

verbose\_name = "статья блога" # имя, под которым модель будет отображаться в административном разделе (для одной статьи блога)

verbose\_name\_plural = "статьи блога" # тоже для всех статей блога

admin.site.register(Blog)

# Модель комментариев

class Comment(models.Model):

text = models.TextField(verbose\_name = "Комментарий")

date = models.DateTimeField(default = datetime.now(), db\_index = True, verbose\_name = "Дата")

author = models.ForeignKey(User, on\_delete = models.CASCADE, verbose\_name = "Автор") # из модели User (вторичный ключ), каскадноеудаление записей в обоих таблицах

post = models.ForeignKey(Blog, on\_delete = models.CASCADE, verbose\_name = "Статья") # из модели Blog (вторичный ключ), каскадное удаление записей в обоих таблицах

def \_\_str\_\_(self):

return 'Комментарий %s к %s' % (self.author, self.post)

class Meta: # метаданные — вложенный класс, который задает дополнительные параметры модели:

db\_table = "Comments" # имя таблицы для модели

verbose\_name = "Комментарий"

verbose\_name\_plural = "Комментарии к статьям блога"

ordering = ["-date"]

admin.site.register(Comment)

**views.py**

"""

Definition of views.

"""

from.forms import BlogForm

from.models import Comment # использование модели комментариев

from.forms import CommentForm # использование формы ввода комментария

from .models import Blog

from app.models import CaruselInfo

#from django.urls import reverse

from django.core.urlresolvers import reverse

from django.http import JsonResponse

from django.http import HttpResponse

from django.http import HttpResponseRedirect

from app.models import Category,Product,CartItem

from app.models import Cart

from django.shortcuts import render

from django.http import HttpRequest

from django.template import RequestContext

from datetime import datetime

from decimal import Decimal

from app.forms import OrderForm

from app.models import Order

from django.shortcuts import render, redirect

from django.contrib.auth.forms import UserCreationForm

from django.shortcuts import render

from django.http import HttpRequest

from django.template import RequestContext

from datetime import datetime

def home(request):

"""Главная страница."""

assert isinstance(request, HttpRequest)

carusels=CaruselInfo.objects.all()

return render(

request,

'app/index.html',

{

'carusels':carusels,

'title':'Home Page',

'year':datetime.now().year,

}

)

def contact(request):

"""Контакты."""

assert isinstance(request, HttpRequest)

return render(

request,

'app/contact.html',

{

'title':'Контакты',

'message':'Контакты интернет магазина «CONNECT electronics».',

'Developer':'Контакты разработчика сайта',

'year':datetime.now().year,

}

)

def about(request):

"""О компании"""

assert isinstance(request, HttpRequest)

categories=Category.objects.all()

products=Product.objects.all()

return render(

request,

'app/about.html',

{

'title':'About',

'message':'Your application description page.',

'year':datetime.now().year,

}

)

def gallery(request):

"""Главная страница."""

assert isinstance(request, HttpRequest)

return render(

request,

'app/gallery.html',

{

'title':'gallery',

'message':'Your application description page.',

'year':datetime.now().year,

}

)

def registration(request):

"""Регистрация."""

regform = UserCreationForm (request.POST)

if request.method == "POST": # после отправки формы

regform = UserCreationForm (request.POST)

if regform.is\_valid(): #валидация полей формы

reg\_f = regform.save(commit=False) # не сохраняем данные формы

reg\_f.is\_staff = False # запрещен вход в административный раздел

reg\_f.is\_active = True # активный пользователь

reg\_f.is\_superuser = False # не является суперпользователем

reg\_f.date\_joined = datetime.now() # дата регистрации

reg\_f.last\_login = datetime.now() # дата последней авторизации

reg\_f.save() # сохраняем изменения после добавления данных (добавление пользователя в БД пользователей)

return redirect('home') # переадресация на главную страницу после регистрации

else:

regform = UserCreationForm() # создание объекта формы для ввода данных нового пользователя

assert isinstance(request, HttpRequest)

return render(request,'app/registration.html',{

'regform': regform, # передача формы в шаблон веб-страницы

'year':datetime.now().year,

}

)

def base\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

categories=Category.objects.all()

products=Product.objects.all()

context={

'categories':categories,

'products':products,

}

return render(request,'app/base.html',context)

def product\_view(request,product\_slug):

try:

cart\_id=request.session['cart\_id']

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id

cart=Cart.objects.get(id=cart\_id)

product=Product.objects.get(slug=product\_slug)

categories=Category.objects.all()

context={

'product':product,

'categories':categories,

'cart':cart,

}

return render(request,'app/product.html',context)

def category\_view(request,category\_slug):

category=Category.objects.get(slug=category\_slug)

products\_of\_category=Product.objects.filter(category=category)

context={

'category':category,

'products\_of\_category': products\_of\_category

}

return render(request,'app/category.html',context)

def cart\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

context={

'cart':cart

}

return render(request,'app/cart.html',context)

def add\_to\_cart\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

product\_slug=request.GET.get('product\_slug')

product=Product.objects.get(slug=product\_slug)

cart.add\_to\_cart(product.slug)

new\_cart\_total=0.00

for item in cart.items.all():

new\_cart\_total+=float(item.item\_total)

cart.cart\_total=new\_cart\_total

cart.save()

return JsonResponse({'cart\_total':cart.items.count(),'cart\_total\_price':cart.cart\_total})

def remove\_from\_cart\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

product\_slug=request.GET.get('product\_slug')

product=Product.objects.get(slug=product\_slug)

cart.remove\_from\_cart(product.slug)

new\_cart\_total=0.00

for item in cart.items.all():

new\_cart\_total+=float(item.item\_total)

cart.cart\_total=new\_cart\_total

cart.save()

return JsonResponse({'cart\_total':cart.items.count(),'cart\_total\_price':cart.cart\_total})

def change\_item\_qty(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

qty=request.GET.get('qty')

item\_id=request.GET.get('item\_id')

cart.change\_qty(qty, item\_id)

cart\_item=CartItem.objects.get(id=int(item\_id))

cart\_item.qty=int(qty)

cart\_item.item\_total = int(qty) \* Decimal(cart\_item.product.price)

cart\_item.save()

return JsonResponse({'cart\_total':cart.items.count(),

'item\_total':cart\_item.item\_total,

'cart\_total\_price':cart.cart\_total})

def checkout\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

context={

'cart':cart

}

return render(request,'app/checkout.html',context)

def order\_create\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

form=OrderForm(request.POST or None)

context={

'form':form,

'cart':cart

}

return render(request,'app/order.html',context)

def make\_order\_view(request):

try:

cart\_id=request.session['cart\_id']# Получение значения сессии при помощи ключа(то есть, 'cart\_id').

cart=Cart.objects.get(id=cart\_id)

request.session['total']=cart.items.count()

except:

cart=Cart()

cart.save()

cart\_id=cart.id

request.session['cart\_id']=cart\_id# Передача значения в сессию

cart=Cart.objects.get(id=cart\_id)

form=OrderForm(request.POST or None)

if form.is\_valid():

name=form.cleaned\_data['name']

last\_name=form.cleaned\_data['last\_name']

phone=form.cleaned\_data['phone']

buying\_type=form.cleaned\_data['buying\_type']

address=form.cleaned\_data['address']

comments=form.cleaned\_data['comments']

new\_order=Order()

new\_order.user=request.user

new\_order.save()

new\_order.items.add(cart)

new\_order.first\_name=name

new\_order.last\_name=last\_name

new\_order.phone=phone

new\_order.address=address

new\_order.buying\_type=buying\_type

new\_order.comments=comments

new\_order.total=cart.cart\_total

new\_order.save()

del request.session['cart\_id']

del request.session['total']

return HttpResponseRedirect(reverse('thank\_you'))

def account\_view(request):

order=Order.objects.filter(user=request.user).order\_by('-id')#.order\_by('-id') более новый заказ на верхних строчках

context={

'order':order,

}

return render(request,'app/account.html',context)

def blog(request):

"""Renders the blog page."""

posts = Blog.objects.order\_by('-posted') # запрос на выбор всех статей из модели, отсортированных по убыванию даты опубликования

assert isinstance(request, HttpRequest)

return render(

request,

'app/blog.html',

{ # параметр в {} — данные для использования в шаблоне.

'title':'Блог',

'posts': posts, # передача списка статей в шаблон веб-страницы

'year':datetime.now().year

}

)

def blogpost(request, parametr):

"""Renders the blogpost page."""

post\_1 = Blog.objects.get(id=parametr) # запрос на выбор конкретной статьи по параметру

comments=Comment.objects.filter(post=parametr)

if request.method=="POST":# после отправки данных формы на сервер методом POST

form=CommentForm(request.POST)

if form.is\_valid():

comment\_f=form.save(commit=False)

comment\_f.author=request.user# добавляем (так как этого поля нет в форме) в модель Комментария (Comment) в поле автор авторизованного пользователя

comment\_f.date=datetime.now() # добавляем в модель Комментария (Comment) текущую дату

comment\_f.post=Blog.objects.get(id=parametr)# добавляем в модель Комментария (Comment) статью, для которой данный комментарий

comment\_f.save()# сохраняем изменения после добавления полей

return redirect('blogpost', parametr=post\_1.id)# переадресация на ту же страницу статьи после отправки комментария

else:

form = CommentForm() # создание формы для ввода комментария

assert isinstance(request, HttpRequest)

return render(

request,

'app/blogpost.html',

{

'comments': comments, # передача всех комментариев к данной статье в шаблон веб-страницы

'form': form, # передача формы в шаблон веб-страницы

'post\_1': post\_1, # передача конкретной статьи в шаблон веб-страницы

'year':datetime.now().year,

}

)

def newpost(request):

if request.method=="POST"and request.FILES:

form = BlogForm(request.POST, request.FILES)

if form.is\_valid():

cd = form.cleaned\_data

file = cd['image']

title = cd['title']

description = cd['description']

content = cd['content']

form.save()

return redirect('blog')

else:

form = BlogForm()

assert isinstance(request, HttpRequest)

return render(

request,

'app/newpost.html',

{

'form': form

}

)

def my\_order(request):

order=Order.objects.all().order\_by('-id')#.order\_by('-id') более новый заказ на верхних строчках

context={

'order':order,

}

return render(request,'app/my\_order.html',context)

**setting.py**

"""

Django settings for firstDjangosite project.

Generated by 'django-admin startproject' using Django 1.9.1.

For more information on this file, see

https://docs.djangoproject.com/en/1.9/topics/settings/

For the full list of settings and their values, see

https://docs.djangoproject.com/en/1.9/ref/settings/

"""

import os

import posixpath

#LOGIN\_URL="login"

#LOGOUT\_URL="logout"

#LOGIN\_REDIRECT\_URL="/"

# Build paths inside the project like this: os.path.join(BASE\_DIR, ...)

BASE\_DIR = os.path.dirname(os.path.dirname(os.path.abspath(\_\_file\_\_)))

# Quick-start development settings - unsuitable for production

# See https://docs.djangoproject.com/en/1.9/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!

SECRET\_KEY = '0478c8d5-f69d-4449-af29-b2a8d7ef29e8'

# SECURITY WARNING: don't run with debug turned on in production!

DEBUG = True

ALLOWED\_HOSTS = []

# Application definition

INSTALLED\_APPS = [

'app',

# Add your apps here to enable them

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

'crispy\_forms',#Пакет django-cripsy-forms для стилизации форм.( https://sharelink.ru/blog/ustanovka-nastrojka-i-primenenie-django-cripsy-forms/)

'bootstrapform',#Пакет django-bootstrap-form( https://github.com/tzangms/django-bootstrap-form)

]

MIDDLEWARE\_CLASSES = [

'django.middleware.security.SecurityMiddleware',

'django.contrib.sessions.middleware.SessionMiddleware',

'django.middleware.common.CommonMiddleware',

'django.middleware.csrf.CsrfViewMiddleware',

'django.contrib.auth.middleware.AuthenticationMiddleware',

'django.contrib.auth.middleware.SessionAuthenticationMiddleware',

'django.contrib.messages.middleware.MessageMiddleware',

'django.middleware.clickjacking.XFrameOptionsMiddleware',

]

ROOT\_URLCONF = 'firstDjangosite.urls'

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [],

'APP\_DIRS': True,

'OPTIONS': {

'context\_processors': [

'django.template.context\_processors.debug',

'django.template.context\_processors.request',

'django.contrib.auth.context\_processors.auth',

'django.contrib.messages.context\_processors.messages',

'django.template.context\_processors.media',#Если включить этот процессор, в RequestContext будет добавлена переменная MEDIA\_URL, которая содержит значение MEDIA\_URL.

],

},

},

]

WSGI\_APPLICATION = 'firstDjangosite.wsgi.application'

# Database

# https://docs.djangoproject.com/en/1.9/ref/settings/#databases

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.sqlite3',

'NAME': os.path.join(BASE\_DIR, 'db.sqlite3'),

}

}

# Password validation

# https://docs.djangoproject.com/en/1.9/ref/settings/#auth-password-validators

AUTH\_PASSWORD\_VALIDATORS = [

{

'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',

},

]

# Internationalization

# https://docs.djangoproject.com/en/1.9/topics/i18n/

LANGUAGE\_CODE = 'ru-ru'

TIME\_ZONE = 'Europe/Moscow'

USE\_I18N = True

USE\_L10N = True

USE\_TZ = True

# Static files (CSS, JavaScript, Images)

# https://docs.djangoproject.com/en/1.9/howto/static-files/

STATIC\_URL = '/static/'

STATIC\_ROOT = posixpath.join(\*(BASE\_DIR.split(os.path.sep) + ['static']))

MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')

MEDIA\_URL = '/media/'

#Pillow, форк от PIL (Python Imaging Library) - библиотека для манипуляций с картинками для python.

#pip install Pillow

# Pillow(https://pythonru.com/biblioteki/osnovnye-vozmozhnosti-biblioteki-python-imaging-library-pillow-pil)

CRISPY\_TAMPLATE\_PACK='bootstrap3'

**Urls.py**

"""

Definition of urls for firstDjangosite.

"""

from app.views import my\_order

from app.views import account\_view

from django.views.generic import TemplateView

from app.views import make\_order\_view

from app.views import order\_create\_view

from app.views import checkout\_view

from app.views import change\_item\_qty

from app.views import remove\_from\_cart\_view

from app.views import add\_to\_cart\_view

from app import views

from app.views import cart\_view

from app.views import category\_view,product\_view

from django.conf import settings#это объект.Файл настроек Django содержит полную конфигурацию установленного проект

from django.conf.urls.static import static#Работа со статическими файлами (CSS, изображения)

from django.contrib.staticfiles.urls import staticfiles\_urlpatterns#собирает статические файлы со всех ваших приложений

from datetime import datetime

from django.conf.urls import url

import django.contrib.auth.views

import app.forms

import app.views

# Uncomment the next lines to enable the admin:

from django.conf.urls import include

from django.contrib import admin

admin.autodiscover() # инициализирует административный раздел

urlpatterns = [

# Examples:

url(r'^$', app.views.home, name='home'),

url(r'^contact$', app.views.contact, name='contact'),

url(r'^about$', app.views.about, name='about'),

url(r'^base\_view$', app.views.base\_view, name='base\_view'),

url(r'^gallery$', app.views.gallery, name='gallery'),

url(r'^blog$', app.views.blog, name='blog'),

url(r'^(?P<parametr>\d+)/$', app.views.blogpost, name='blogpost'),

url(r'^newpost$', app.views.newpost, name='newpost'),

url(r'^login/$',

django.contrib.auth.views.login,

{

'template\_name': 'app/login.html',

'authentication\_form': app.forms.BootstrapAuthenticationForm,

'extra\_context':

{

'title': 'Авторизоваться',

'year': datetime.now().year,

}

},

name='login'),

url(r'^logout$',

django.contrib.auth.views.logout,

{

'next\_page': '/',

},

name='logout'),

# Uncomment the admin/doc line below to enable admin documentation:

url(r'^admin/doc/', include('django.contrib.admindocs.urls')),

# Uncomment the next line to enable the admin:

url(r'^admin/', include(admin.site.urls)),

url(r'^registration$', app.views.registration, name='registration'),

url(r'^category/(?P<category\_slug>[-\w]+)/$',category\_view, name='category\_detail'),

url(r'^product/(?P<product\_slug>[-\w]+)/$', product\_view, name='product\_detail'),

url(r'^add\_to\_cart/$', add\_to\_cart\_view, name='add\_to\_cart'),

url(r'^remove\_from\_cart/$', remove\_from\_cart\_view, name='remove\_from\_cart'),

url(r'^change\_item\_qty/$', change\_item\_qty, name='change\_item\_qty'),

url(r'^cart/$',cart\_view,name='cart'),

url(r'^checkout/$',checkout\_view,name='checkout'),

url(r'^order/$',order\_create\_view,name='order\_create'),

url(r'^make\_order/$',make\_order\_view,name='make\_order'),

url(r'^thank\_you/$',TemplateView.as\_view(template\_name='app/thank\_you.html'),name='thank\_you'),

url(r'^account/$',account\_view,name='account'),

url(r'^my\_order/$',my\_order,name='my\_order'),

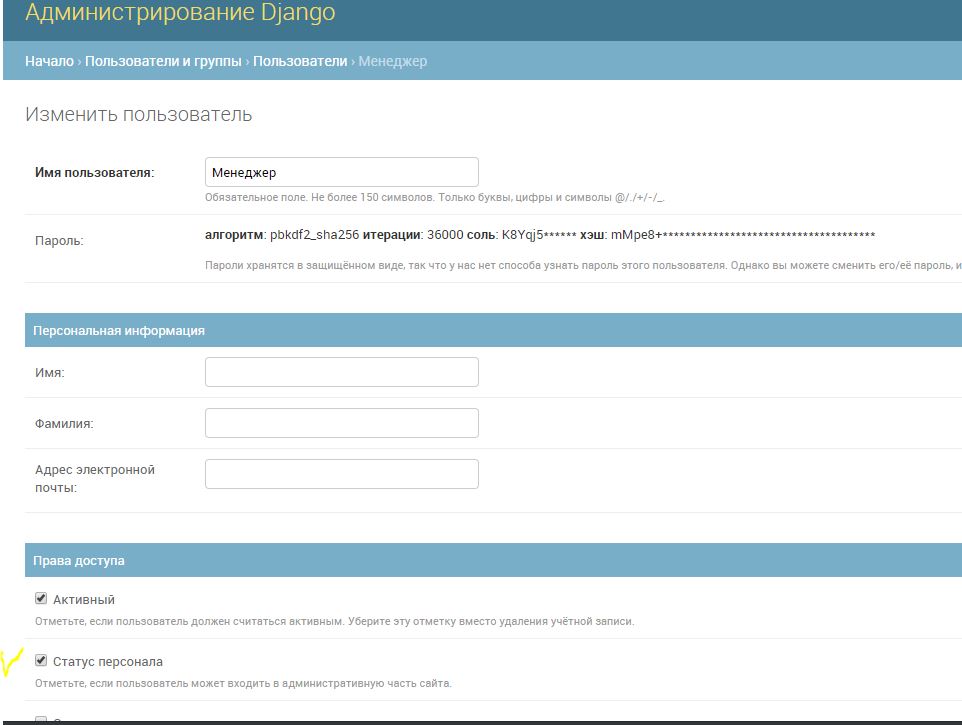
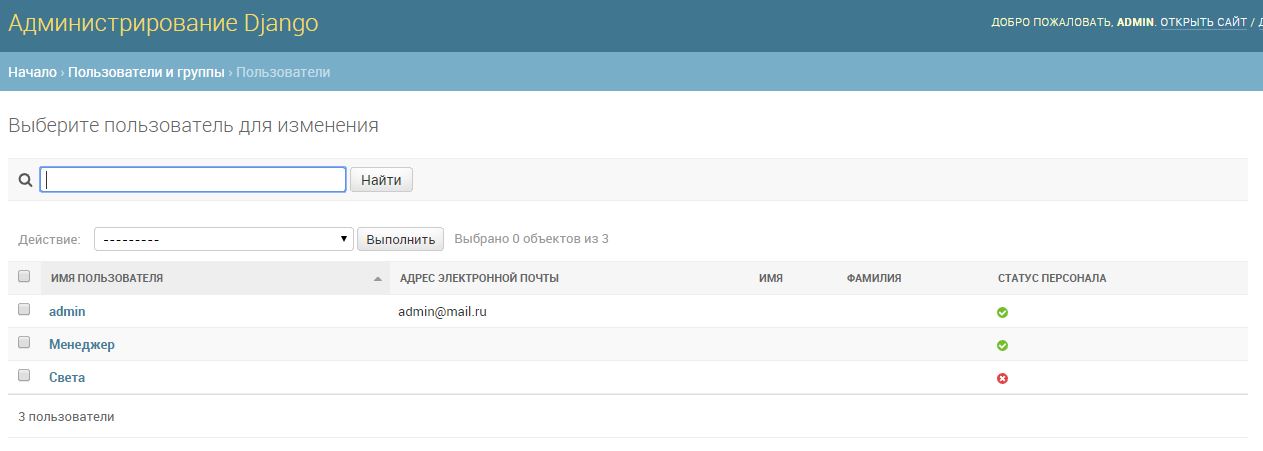
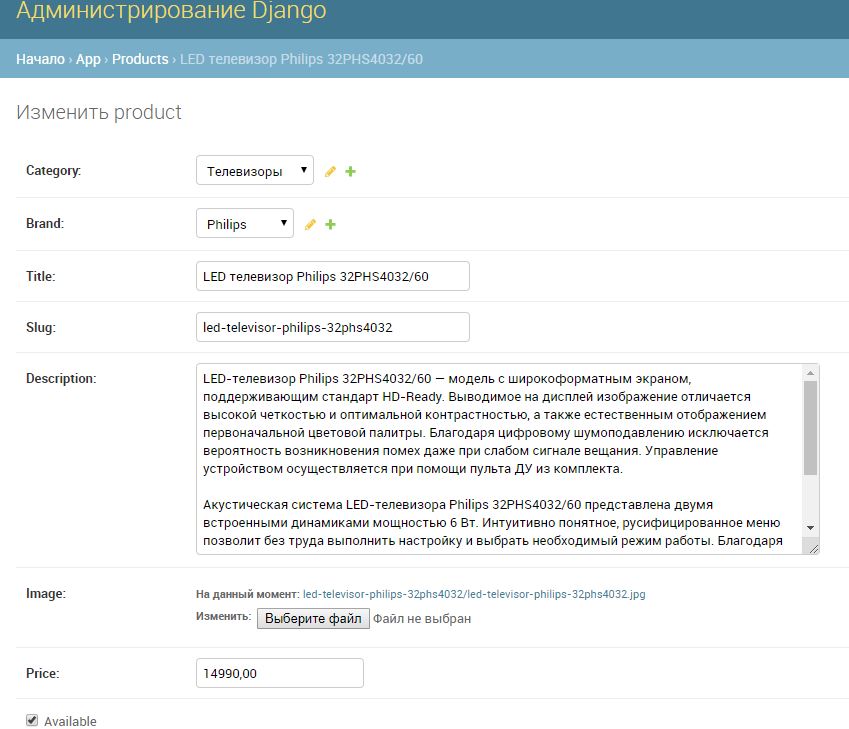
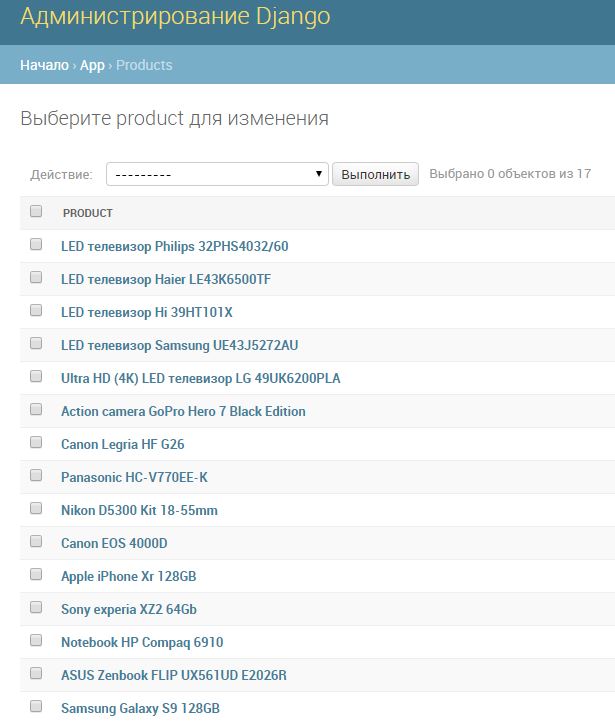
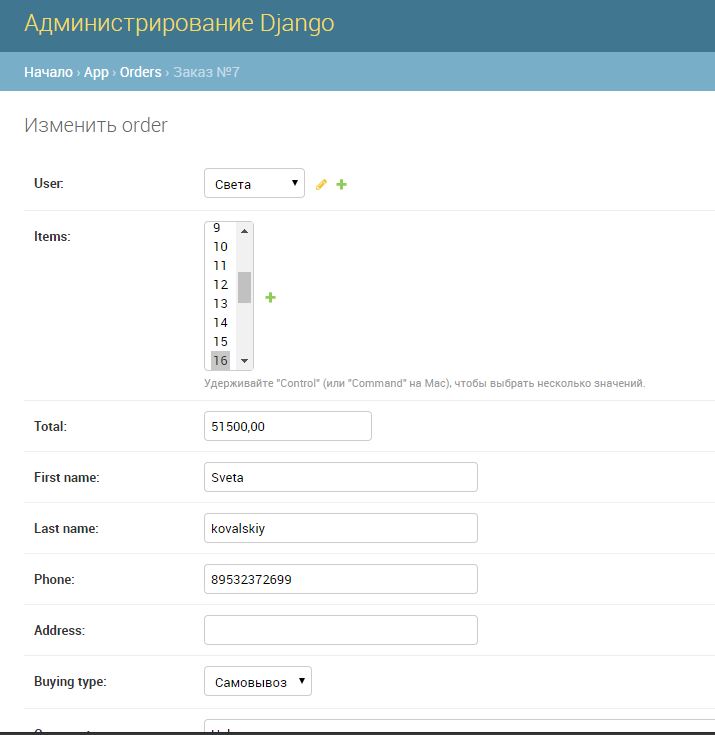
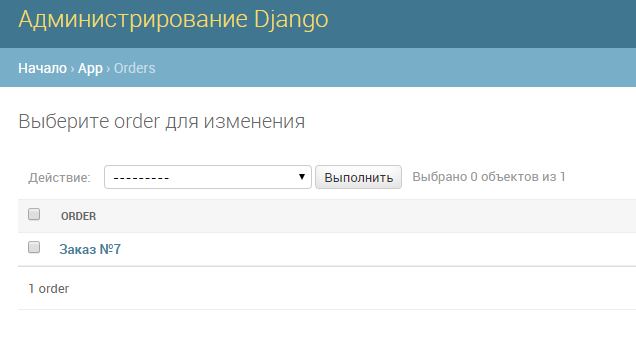
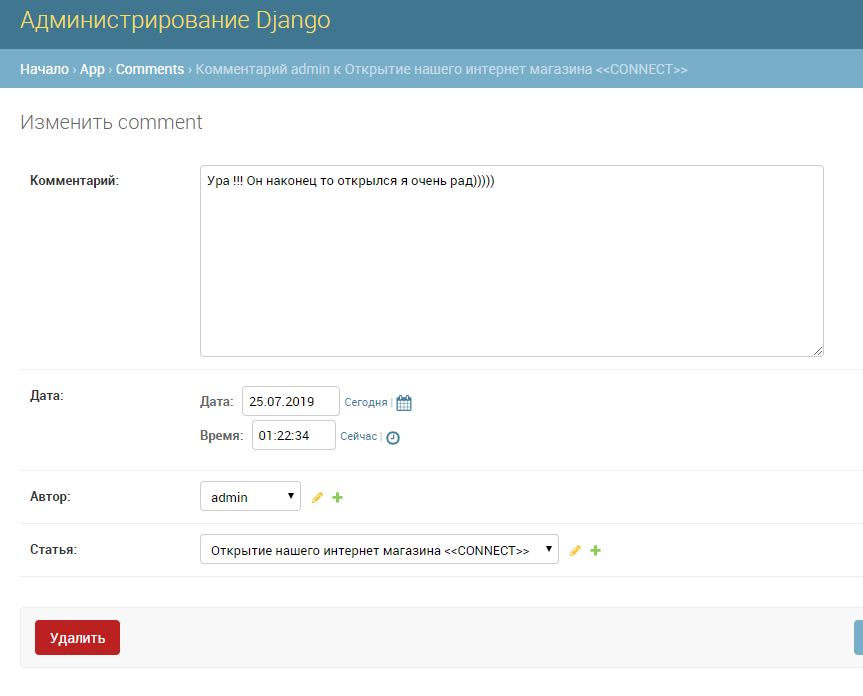
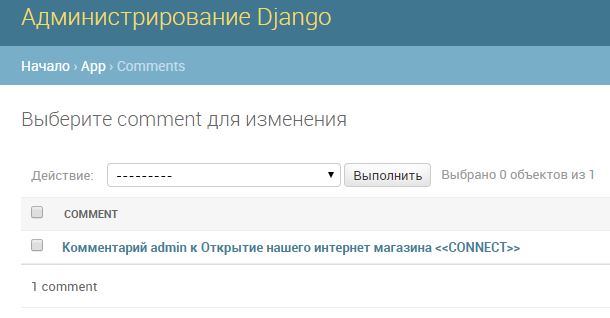
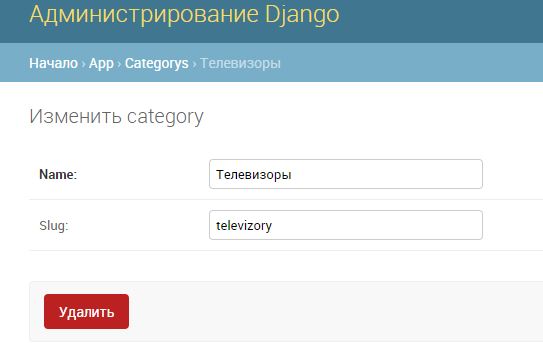
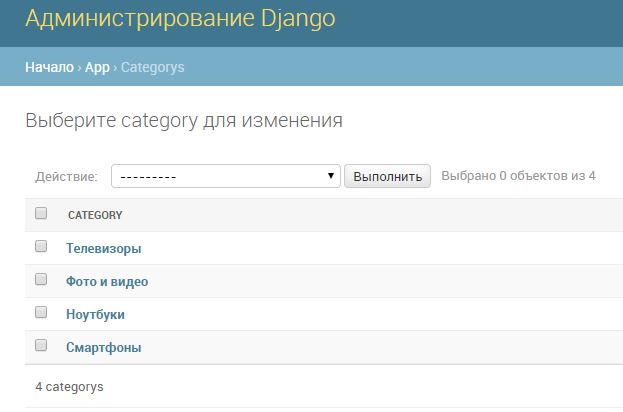
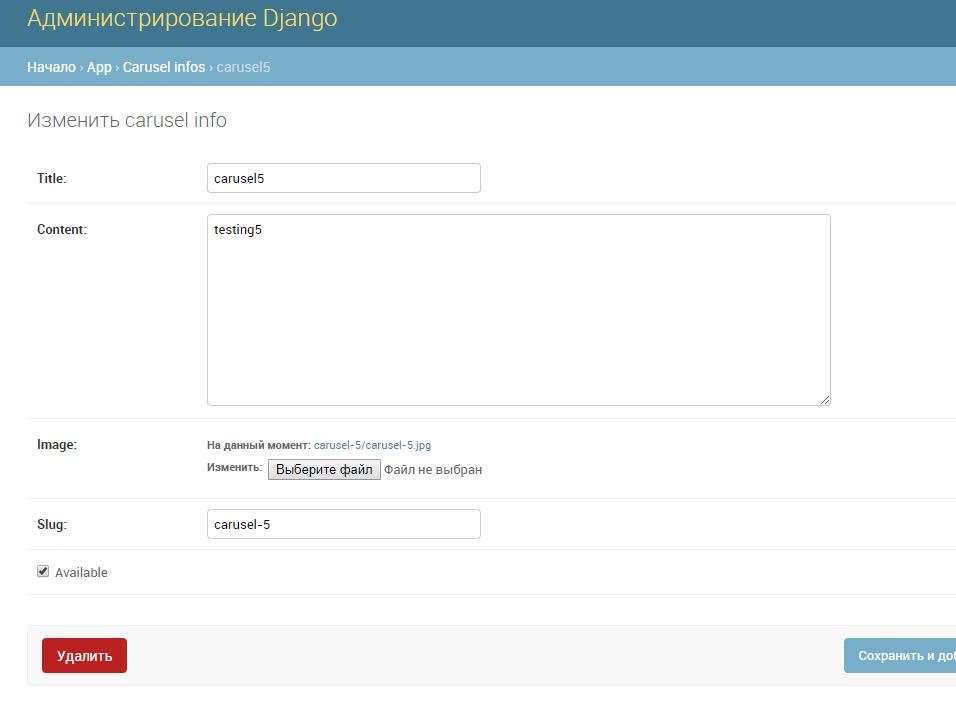
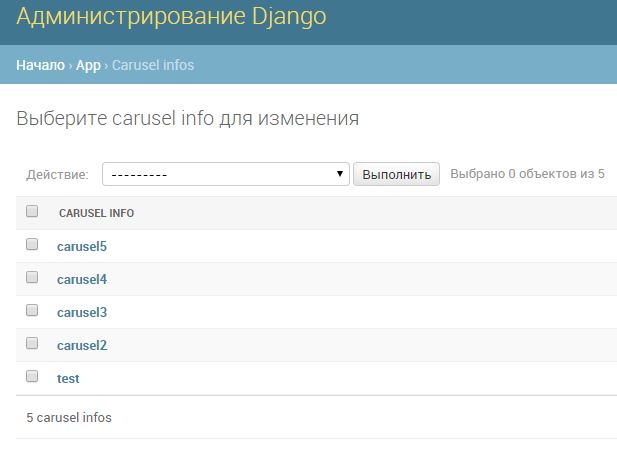
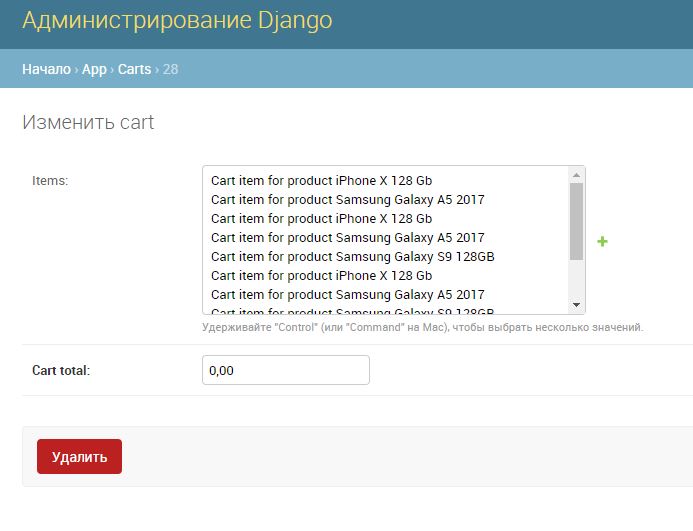
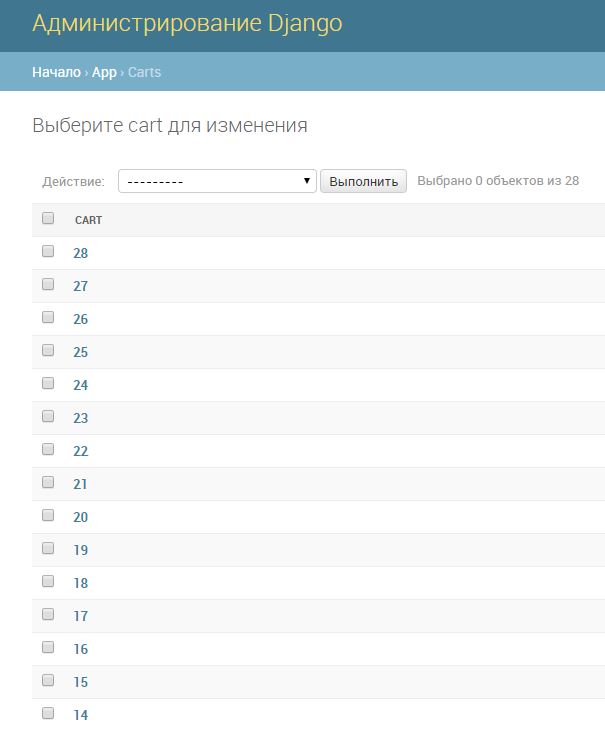
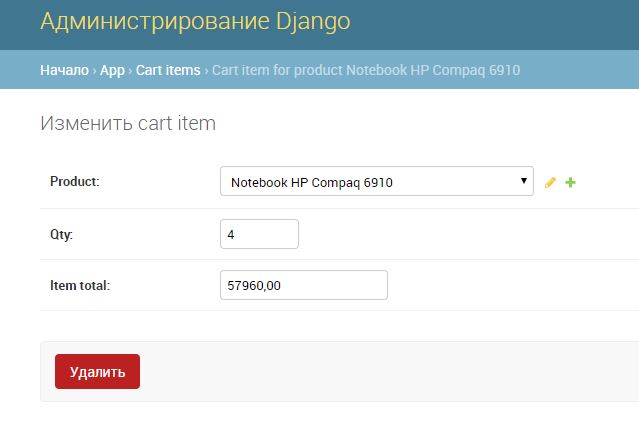
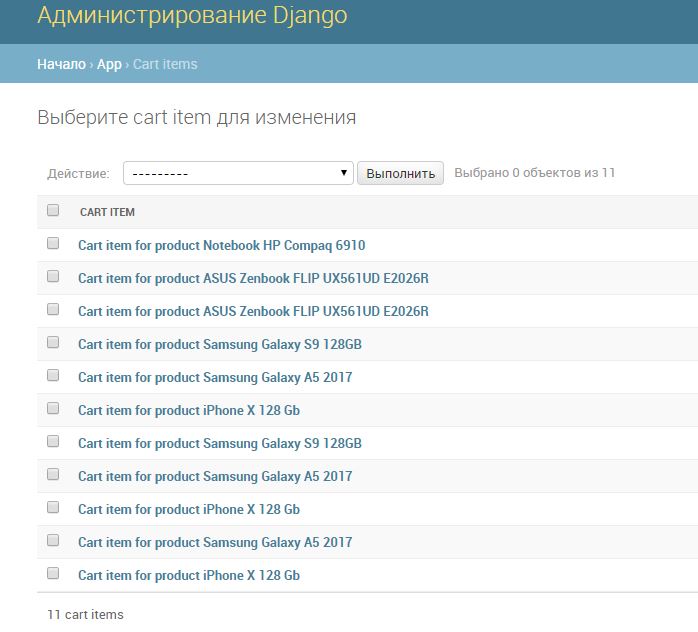
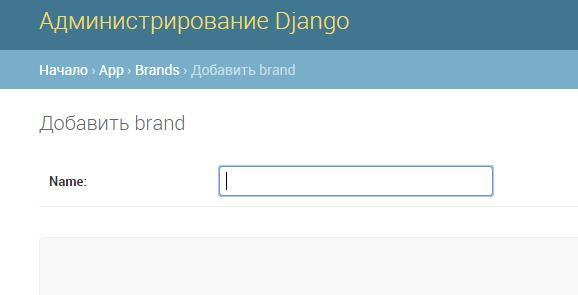
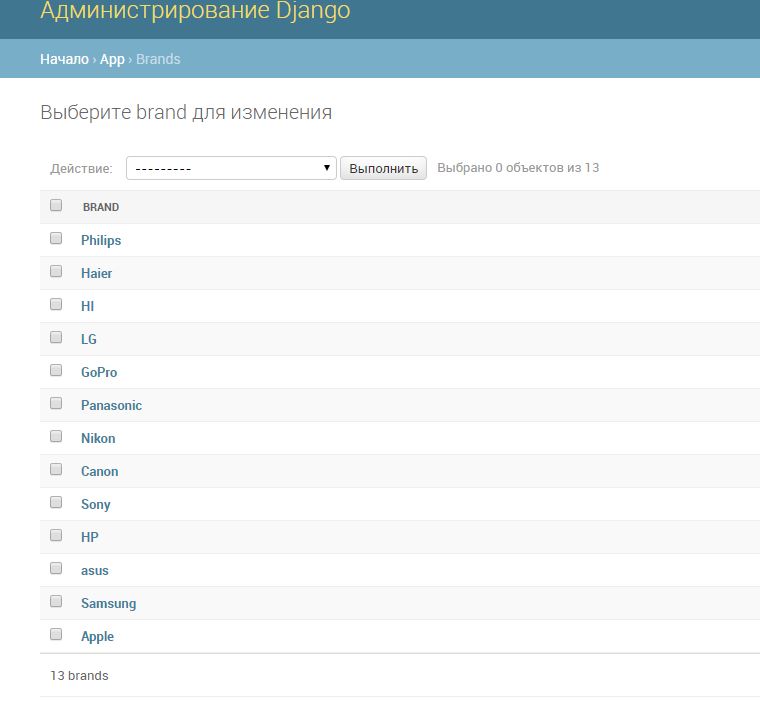
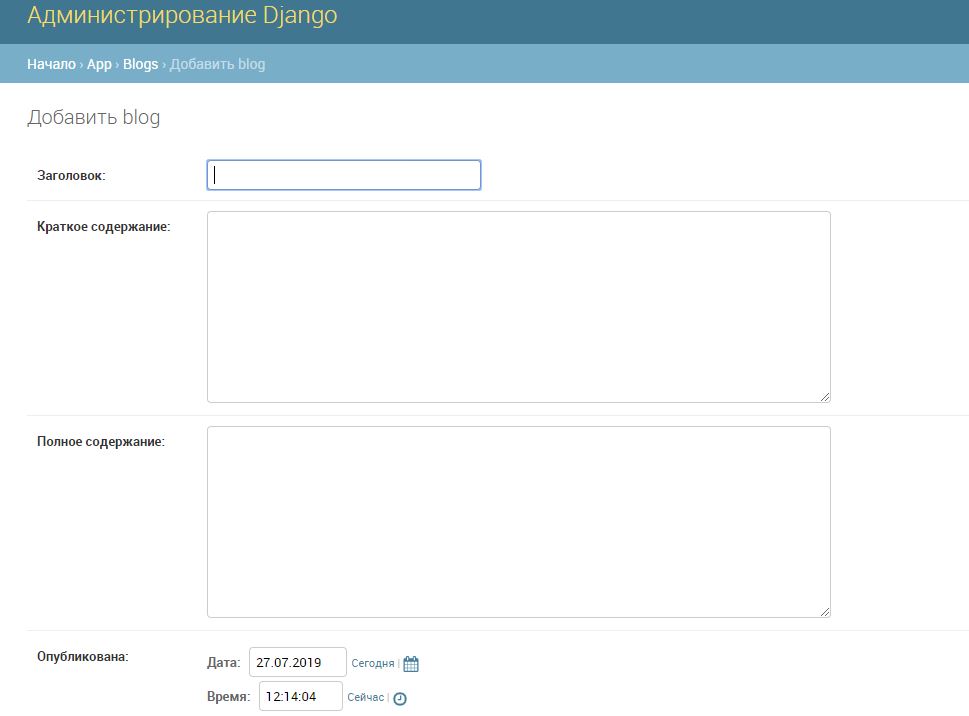
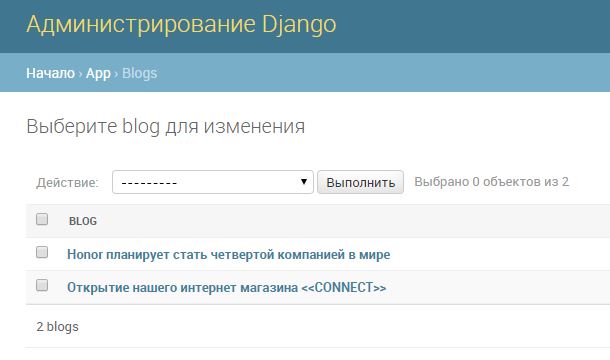
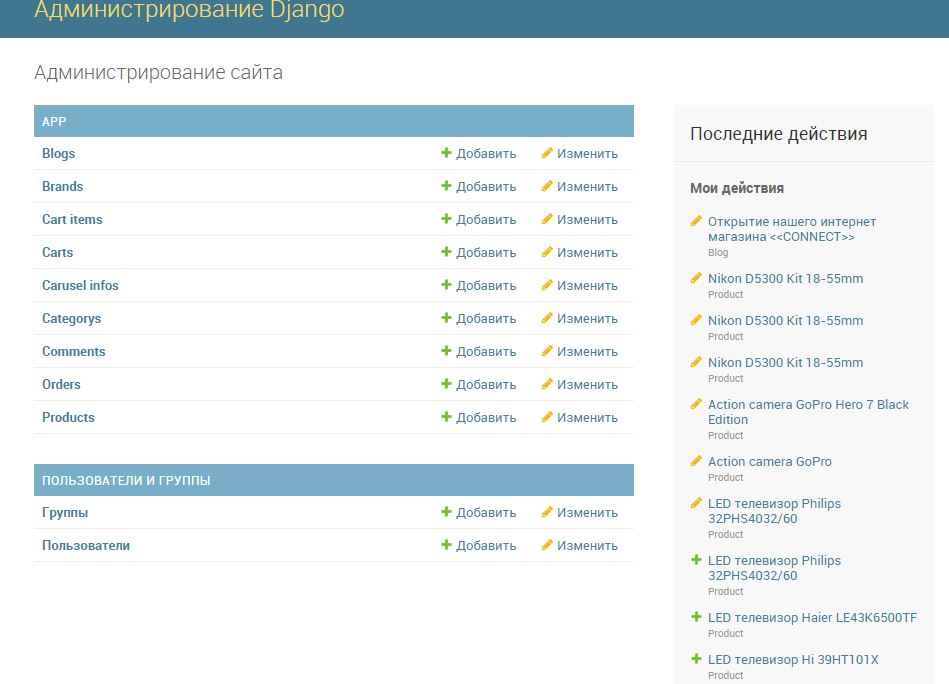
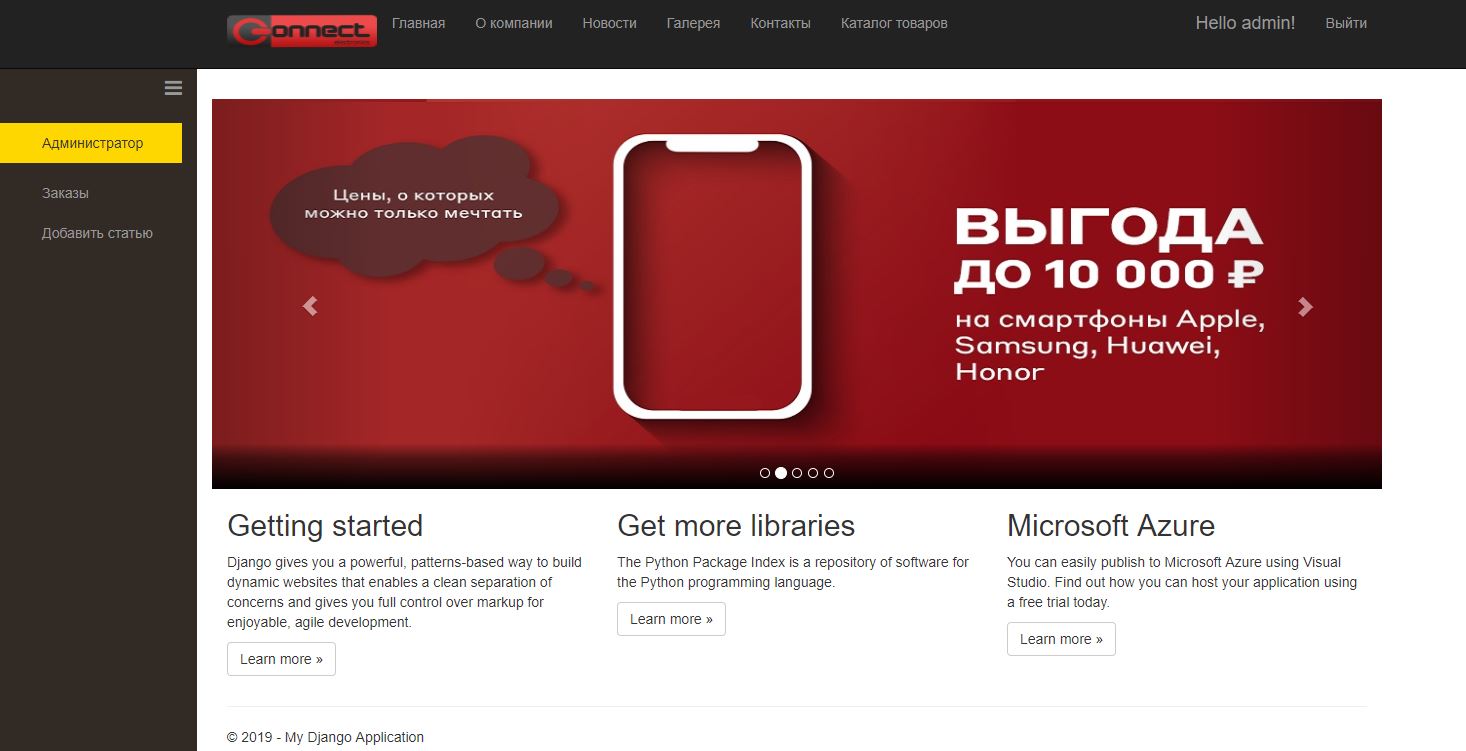
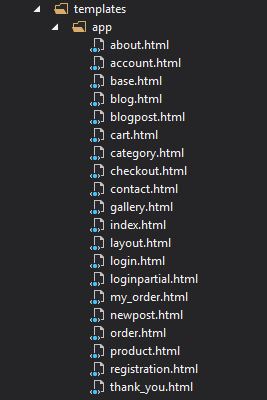
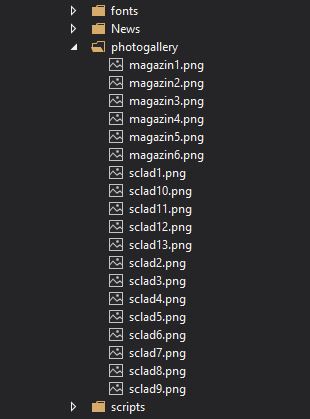
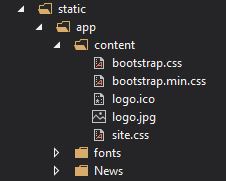
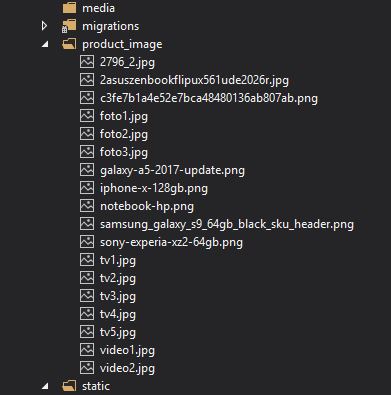
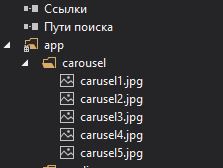
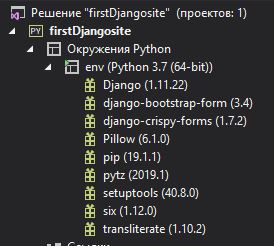
]

if settings.DEBUG:

urlpatterns += static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

urlpatterns += staticfiles\_urlpatterns()

**Скриншоты**

****