

# Django Essential

Реализация Front-end

# Django Essential

## Introduction



Лазорык Михаил

Software developer, 3 года опыта

 mykhailo.lazoryk

 mykhailo-lazoryk



# Django Essential

Тема урока

Реализация Front-end

# Django Essential

## Реализация Front-end

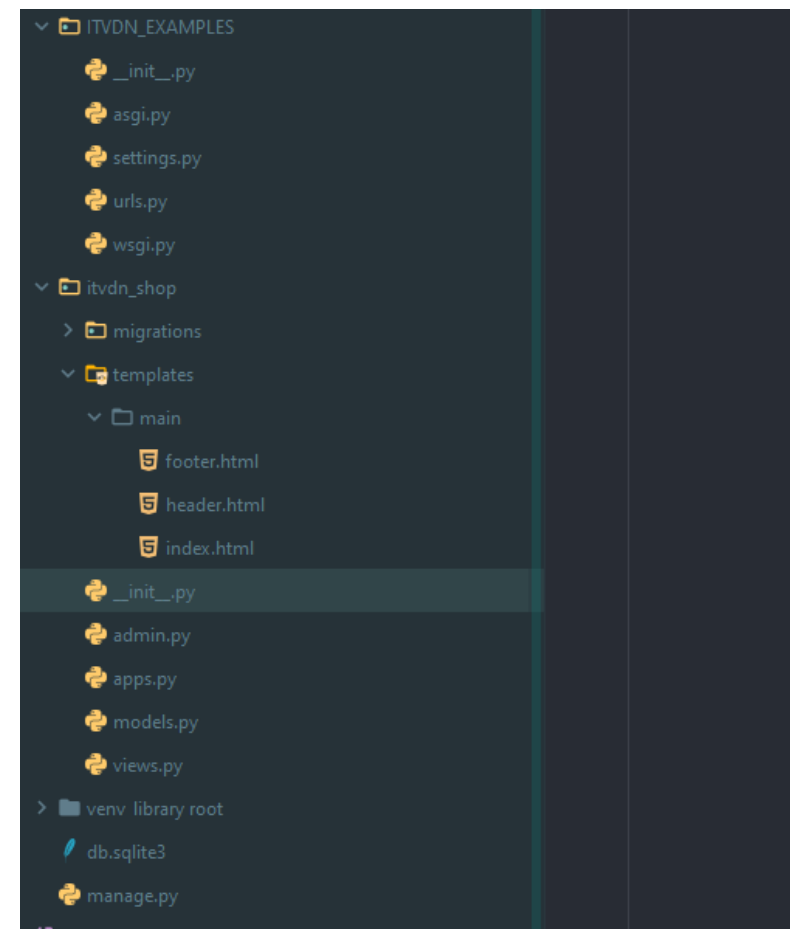
1. Создание index.html
2. Создание шаблонов для приложения
3. Подключение Bootstrap: локально и с помощью ссылки
4. Создание структуры статик файлов.
5. Добавление css, js, jquery и других дополнительных библиотек
6. Рассмотрение типов view и способы их применения

# Django Essential

## Создание index.html

Index.html - “лицо” вашего сайта.

Чтобы создать, используем следующую структуру



# Django Essential

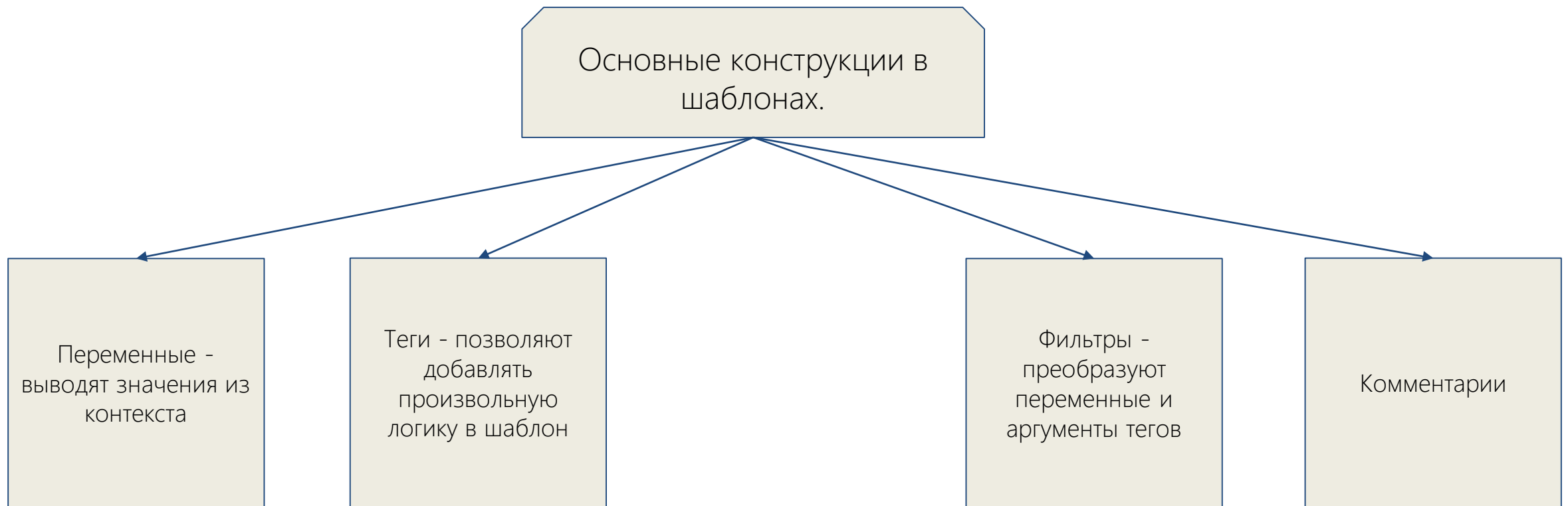
## Создание шаблонов для приложения

Пример подключения шаблонов к `index.html`.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Title</title>
6 </head>
7 <body>
8     {% include "main/header.html" %}  Подключение header
9     {% block content %}
10         This is content  Место для контента,
11     {% endblock %}        которое в зависимости от
12 </body>                  страницы будет меняться
13     {% include "main/footer.html" %}  Подключение footer
14 </html>
```

# Django Essential

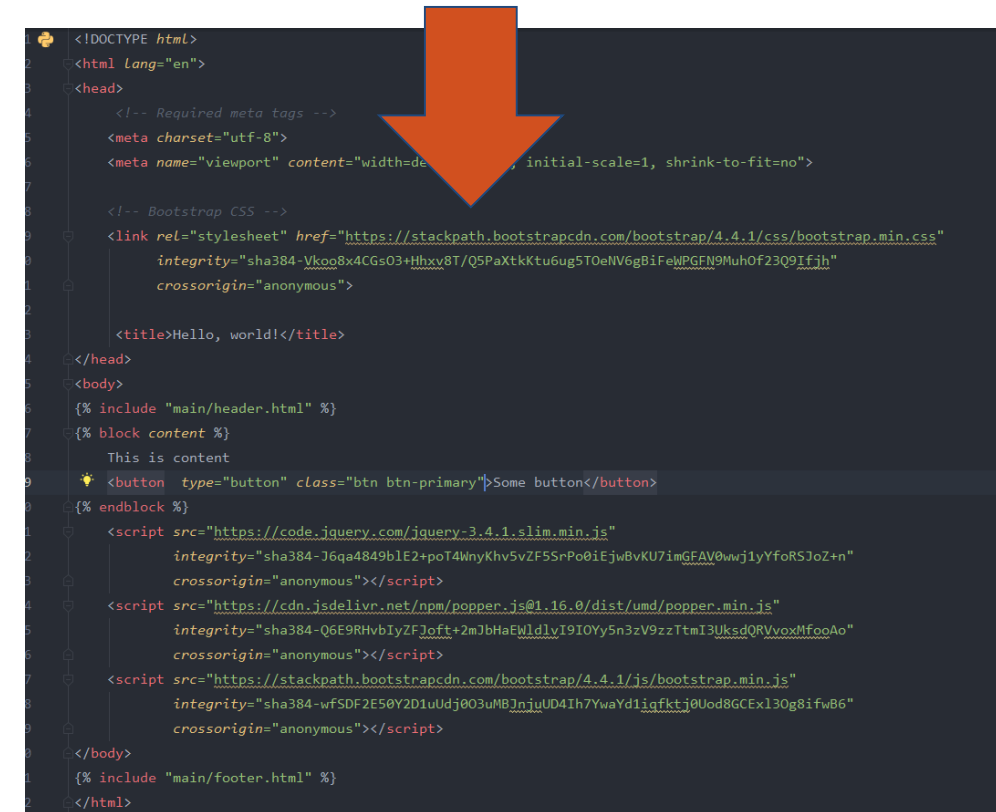
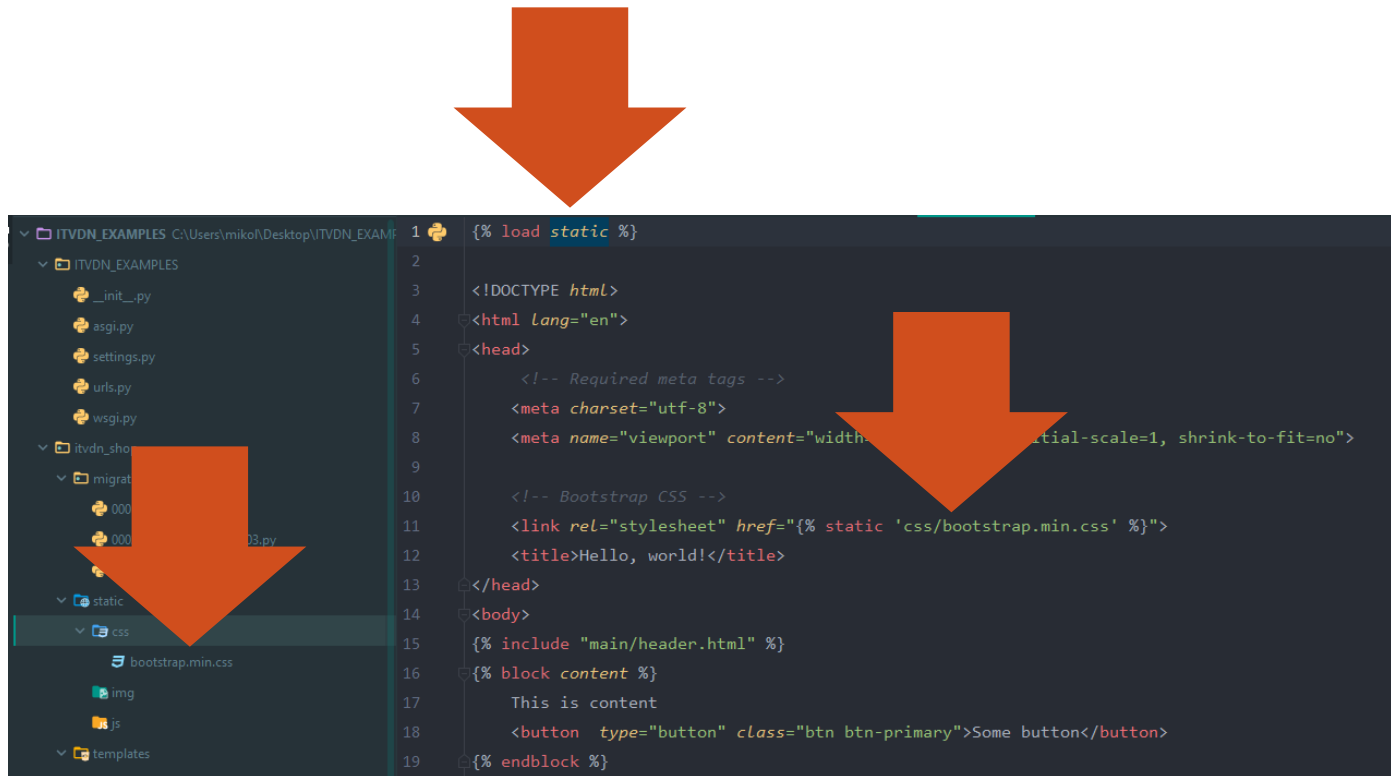
## Создание шаблонов для приложения



# Django Essential

## Подключение Bootstrap: локально и с помощью ссылки

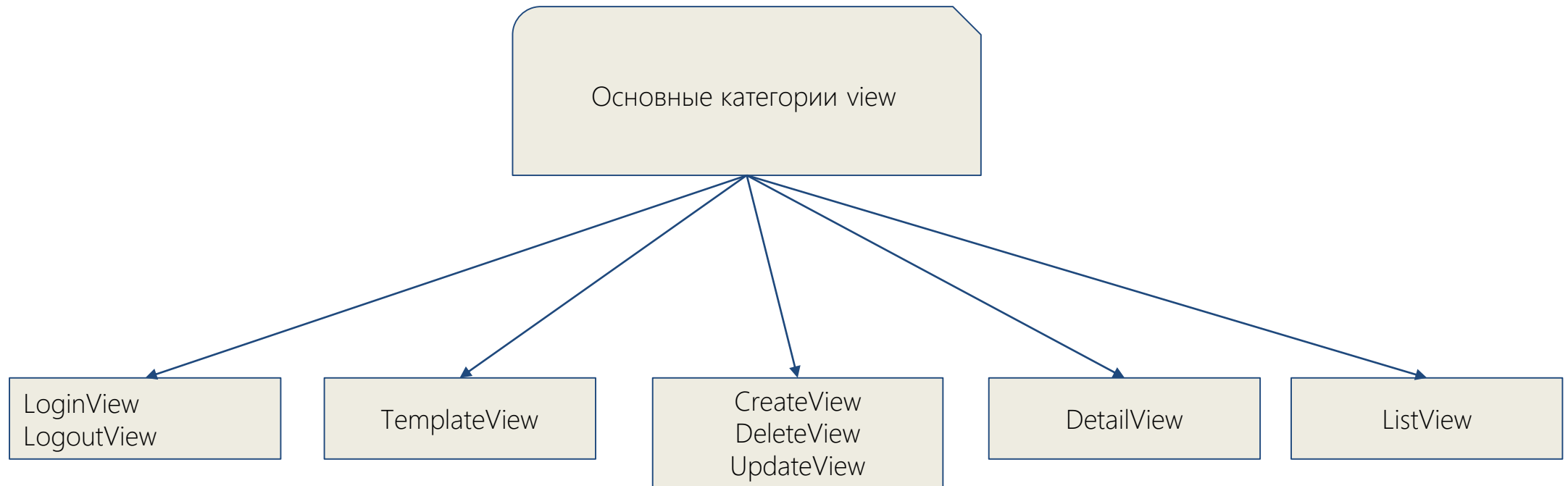
Для подключения **bootstrap** используем CDN с официального сайта. Также есть способ через загрузку файлов и их подключения через статик.





# Django Essential

## Рассмотрение типов view и способы их применения



# Django Essential

## ListView

### Attributes

	Defined in
<code>allow_empty = True</code>	<a href="#">MultipleObjectMixin</a>
<code>content_type = None</code>	<a href="#">TemplateResponseMixin</a>
<code>context_object_name = None</code>	<a href="#">MultipleObjectMixin</a>
<code>extra_context = None</code>	<a href="#">ContextMixin</a>
<code>http_method_names = ['get', 'post', 'put', 'patch', 'delete', 'head', 'options', 'trace']</code>	<a href="#">View</a>
<code>model = None</code>	<a href="#">MultipleObjectMixin</a>
<code>ordering = None</code>	<a href="#">MultipleObjectMixin</a>
<code>page_kwarg = 'page'</code>	<a href="#">MultipleObjectMixin</a>
<code>paginate_by = None</code>	<a href="#">MultipleObjectMixin</a>
<code>paginate_orphans = 0</code>	<a href="#">MultipleObjectMixin</a>
<code>paginator_class = &lt;class 'django.core.paginator.Paginator'&gt;</code>	<a href="#">MultipleObjectMixin</a>
<code>queryset = None</code>	<a href="#">MultipleObjectMixin</a>
<code>response_class = &lt;class 'django.template.response.TemplateResponse'&gt;</code>	<a href="#">TemplateResponseMixin</a>
<code>template_engine = None</code>	<a href="#">TemplateResponseMixin</a>
<code>template_name = None</code>	<a href="#">TemplateResponseMixin</a>
<code>template_name_suffix = '_list'</code>	<a href="#">MultipleObjectTemplateResponseMixin</a>

<https://ccbv.co.uk/projects/Django/2.2/django.views.generic.list/ListView/>

# Django Essential

## ListView

### Methods

[Expand](#) [Collapse](#)

<code>def _allowed_methods(self):</code>	<a href="#">View</a>
<code>def as_view(cls, **initkwargs):</code>	<a href="#">View</a>
<code>def dispatch(self, request, *args, **kwargs):</code>	<a href="#">View</a>
<code>def get(self, request, *args, **kwargs):</code>	<a href="#">BaseListView</a>
<code>def get_allow_empty(self):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_context_data(self, *, object_list=None, **kwargs):</code>	
<code>def get_context_object_name(self, object_list):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_ordering(self):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_paginate_by(self, queryset):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_paginate_orphans(self):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_paginator(self, queryset, per_page, orphans=0, allow_empty_first_page=True, **kwargs):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_queryset(self):</code>	<a href="#">MultipleObjectMixin</a>
<code>def get_template_names(self):</code>	
<code>def http_method_not_allowed(self, request, *args, **kwargs):</code>	<a href="#">View</a>
<code>def __init__(self, **kwargs):</code>	<a href="#">View</a>
<code>def options(self, request, *args, **kwargs):</code>	<a href="#">View</a>
<code>def paginate_queryset(self, queryset, page_size):</code>	<a href="#">MultipleObjectMixin</a>
<code>def render_to_response(self, context, **response_kwargs):</code>	<a href="#">TemplateResponseMixin</a>
<code>def setup(self, request, *args, **kwargs):</code>	<a href="#">View</a>

<https://ccbv.co.uk/projects/Django/2.2/django.views.generic.list/ListView/>



# Проверка знаний

TestProvider.com



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://TestProvider.com)

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.

# Django Essential

Спасибо за внимание! До новых встреч!



Лазорык Михаил  
Software developer



# Информационный видеосервис для разработчиков программного обеспечения

