

Лабораторная работа №2

Платформа Django

Django

Django – веб-фреймворк для перфекционистов с дедлайнами.

Основной принцип – DRY – Don't Repeat Yourself (Не Повторяйся!).

Свободный фреймворк для веб-приложений на языке Python

Django. Возможности

- ORM, API доступа к БД с поддержкой транзакций;
- Встроенный интерфейс администратора с уже имеющимися переводами на многие языки
- Диспетчер URL на основе регулярных выражений
- Расширяемая система шаблонов с тегами и наследованием
- Система кеширования
- Интернационализация

Django. Возможности

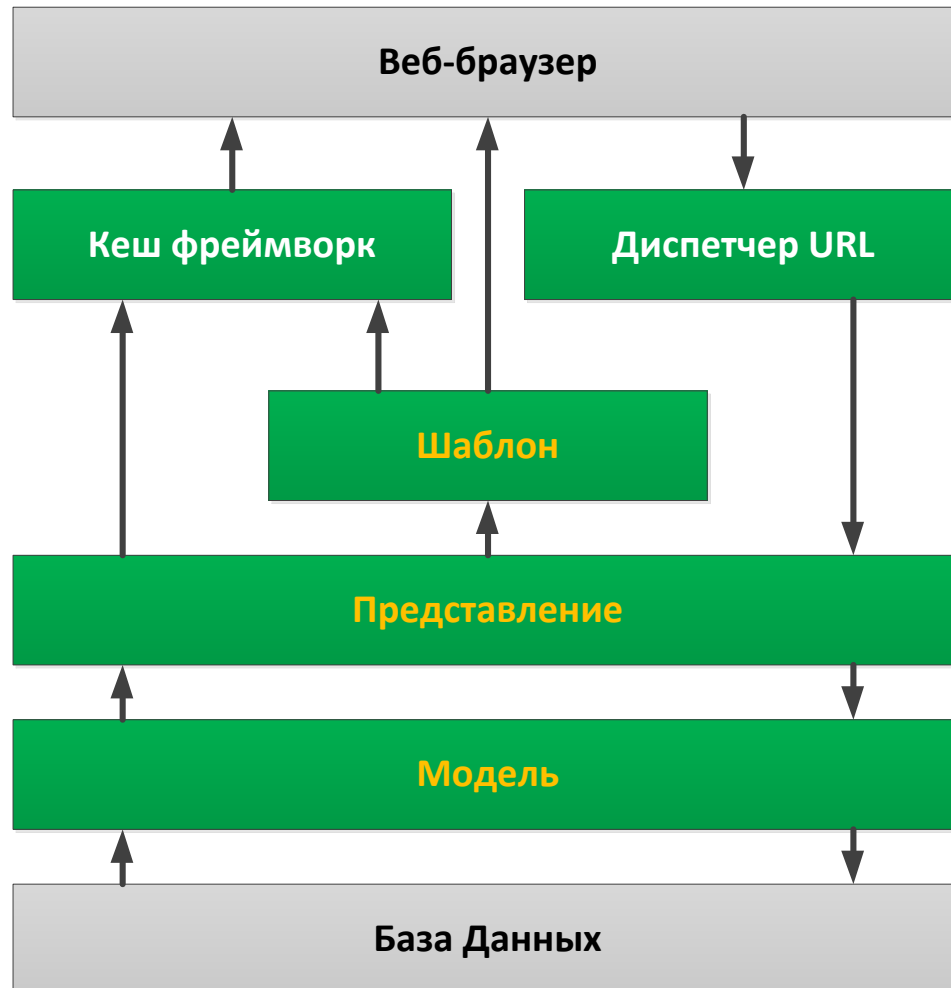
- Подключаемая архитектура приложений, которые можно устанавливать на любые Django-сайты;
- Шаблоны функций контроллеров;
- Авторизация и аутентификация, подключение внешних модулей аутентификации: LDAP, OpenID и тп;
- Библиотека для работы с формами (наследование, построение форм по существующей БД).

Django. MVC

Основной шаблон разработки – MVC (Model-View-Controller)

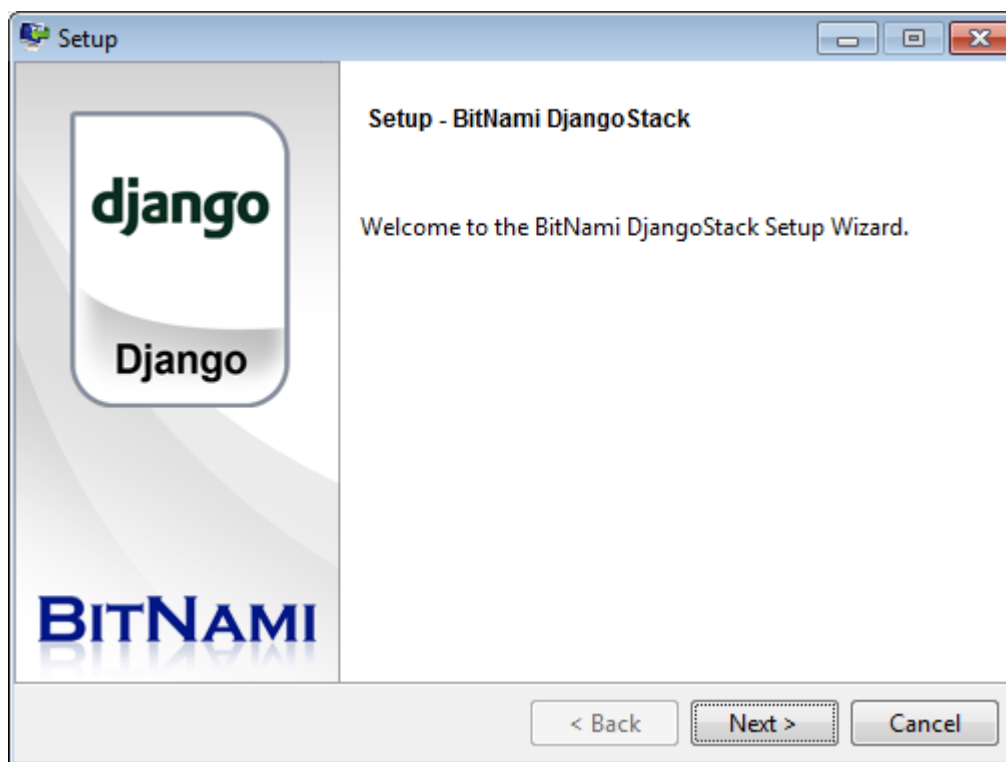
- Модель (model): предоставляет данные(для view) и реагирует на запросы(от controller), изменяя свое состояние;
- Представление (view): отвечает за отображение информации;
- Контроллер (controller): интерпретирует данные от пользователя и информирует модель и представление о необходимости соответствующей реакции.

Django. Как работает приложение



Django. Установка

Полный пакет (Python, Apache, MySQL, PostgreSQL, Python, SQLite) - <http://bitnami.org/stack/djangostack>



Django. Настройка

Установка переменных окружения:

- `set PATH=%PATH%;C:\Program Files\BitNami DjangoStack\python\`
- `set PATH=%PATH%; C:\Program Files\BitNami DjangoStack\apps\django\django\bin`

Django. Создание проекта

Команда создания:

```
django-admin.py startproject cutesite
```

Результат:

```
cutesite/
```

```
__init__.py
```

```
manage.py
```

```
settings.py
```

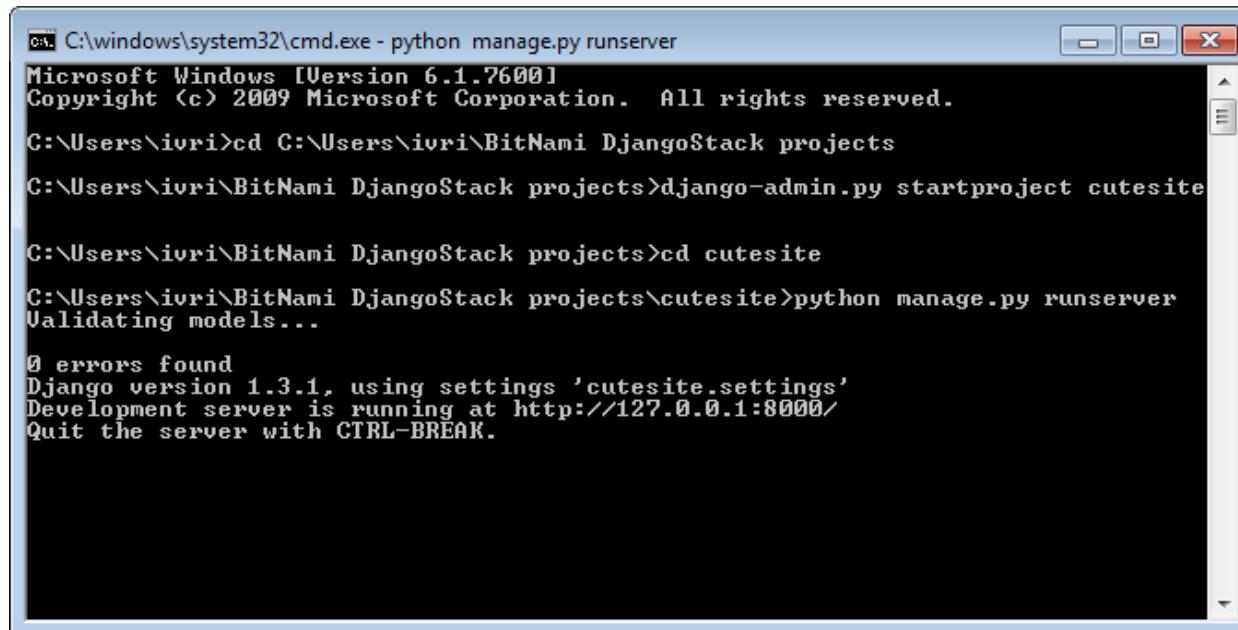
```
urls.py
```

Django. Запуск проекта

Команда запуска:

python manage.py runserver

Результат:



```
C:\windows\system32\cmd.exe - python manage.py runserver
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ivri>cd C:\Users\ivri\BitNami DjangoStack projects
C:\Users\ivri\BitNami DjangoStack projects>django-admin.py startproject cutesite

C:\Users\ivri\BitNami DjangoStack projects>cd cutesite
C:\Users\ivri\BitNami DjangoStack projects\cutesite>python manage.py runserver
Validating models...

0 errors found
Django version 1.3.1, using settings 'cutesite.settings'
Development server is running at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Django. Hello, World

Urls.py

```
from django.conf.urls.defaults import *  
from cutesite.views import hello
```

```
urlpatterns = patterns(",  
    ('^hello/$', hello),  
)
```

Views.py:

```
# -*- coding: utf-8 -*-  
from django.http import HttpResponse  
def hello(request):  
    return HttpResponse("Здравствуй, Мир")
```

Django. Динамический контент

Urls.py

```
from django.conf.urls.defaults import *
from cutesite.views import hello, current_datetime
urlpatterns = patterns("",
    ('^hello/$', hello),
    ('^time/$', current_datetime),
)
```

Views.py:

```
import datetime
def current_datetime(request):
    now = datetime.datetime.now()
    html = "<html><body>It is now %s.</body></html>" % now
    return HttpResponse(html)
```

Django. Динамические URL

Urls.py

```
(r'^time/plus/(\d{1,2})/$', hours_ahead),
```

Views.py:

```
from django.http import Http404, HttpResponse
```

```
import datetime
```

```
def hours_ahead(request, offset):
```

```
    try:
```

```
        offset = int(offset)
```

```
    except ValueError:
```

```
        raise Http404()
```

```
    next_time= datetime.datetime.now() + datetime.timedelta(hours=offset)
```

```
    html = "<html><body>In %s hour(s), it will be %s.</body></html>" % (offset,  
next_time)
```

```
    return HttpResponse(html)
```

Django. Использование шаблонов

settings.py

```
import os.path
```

```
TEMPLATE_DIRS = (  
    os.path.join(os.path.dirname(__file__), 'templates').replace('\\', '/'),  
)
```

views.py:

```
from django.shortcuts import render_to_response
```

```
...
```

```
def current_datetime(request):
```

```
    current_datetime = datetime.datetime.now()
```

```
    return render_to_response('current_datetime.html', locals())
```

current_datetime.html:

```
<html><body>It is now {{ current_datetime }}</body></html>
```

Django. Наследование шаблонов

base.html:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">
<html lang="en">
<head>
<title>{% block title %}{% endblock %}</title>
</head>
<body>
<h1>My helpful timestamp site</h1>
{% block content %}{% endblock %}
{% block footer %}
<hr>
<p>Thanks for visiting my site.</p>
{% endblock %}
</body> </html>
```

Django. Наследование шаблонов

Current_datetime.html:

```
{% extends "base.html" %}  
{% block title %}The current time{% endblock %}  
{% block content %}  
<p>It is now {{ current_datetime }}.</p>  
{% endblock %}
```

Hours_ahead.html

```
{% extends "base.html" %}  
{% block title %}Future time{% endblock %}  
{% block content %}  
<p>In {{ offset }} hour(s), it will be {{ next_time }}.</p>  
{% endblock %}
```