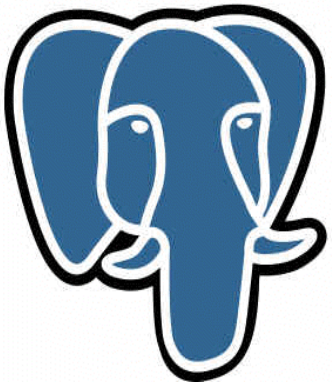
markafoni
Her alışveriş bir aşk

django & PostgreSQL

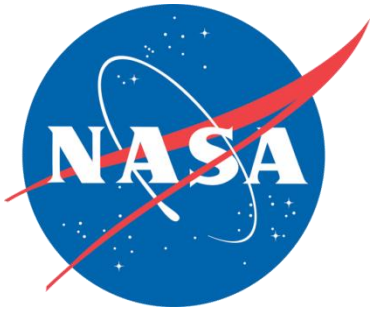


Ege Hanoğlu

Django Nedir?

- Python programlama diliyle yazılmış Web Framework
- Django Reinhardt
- Temmuz 2005'te BSD lisansı ile yayınlanmıştır
- Güncel Sürüm 1.4
- www.djangoproject.com

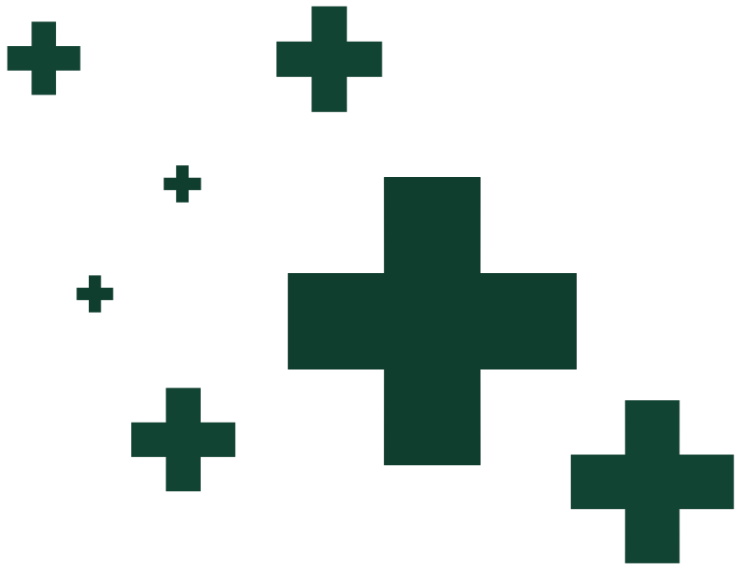
Kullanıldığı Başlıca Siteler



markafoni
Her alışveriş bir aşk

Avantajları Nelerdir?

- Kodlaması basit ve anlaması kolay
- Python kütüphanelerini kullanabilmesi
- Modüler yapıya sahip
- ORM (Object Relational Mapping)
- Daha güvenli
- Otomatik Admin Arayüzü oluşturması
- Internationalization
- Önbellekleme
- Çoklu Veritabanı



markafoni
Her alışveriş bir aşk

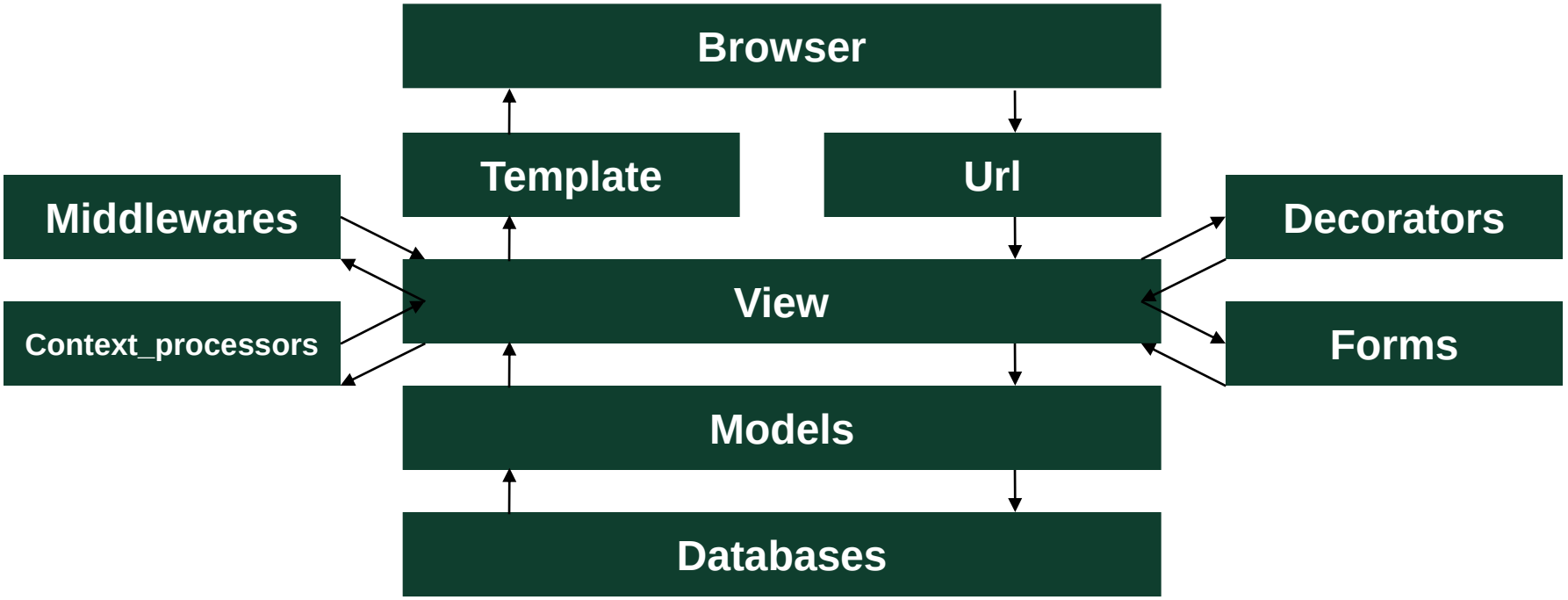
Neden Daha Güvenli?

- CSRF(Cross Site Request Forgery) Korumalı
- HTML Escaping
- ORM

Django'nun Bileşenleri Nelerdir?

markafoni
Her alışveriş bir aşk

Django'nun Çalışma Yapısı Nasıldır?



markafoni
Her alışveriş bir aşk

Python'da Sanal Ortam Nasıl Kurulur?

- easy_install virtualenv
- virtualenv pgday_env
- cd pgday_env/
- source bin/activate
- pip install -r requirements.txt
  - requirements.txt:
    - Django==1.3.1
    - PIL==1.1.7
    - psycopg2==2.4.2
    - python-memcached==1.47
    - ipython==0.11
    - django-debug-toolbar==0.9.4

markafoni
Her alışveriş bir aşk

Django Projesi Nasıl Başlatılır?

- django-admin.py startproject pgday

```
pgday/
__init__.py
manage.py
settings.py
urls.py
```

markafoni
Her alışveriş bir aşk

Postgresql Veritabanı Nasıl Yaratılır?

- createdb pgday -h localhost -U postgres
- Veritabanı yaratıldıktan sonra veritabanı ayarlarının settings.py'e girilmesi gereklidir.

```
DATABASES = {
'default': {'ENGINE': 'django.db.backends.postgresql_psycopg2',
        'NAME': 'pgday',
        'USER': 'postgres',
        'PASSWORD': '123123',
        'HOST': '127.0.0.1',
        'PORT': '5432',}
}
```

markafoni
Her alışveriş bir aşk

Django Aplikasyonu Nasıl Başlatılır?

- django-admin.py startapp product

```
product/
__init__.py
models.py
tests.py
views.py
```

Aplikasyon yaratıldıktan sonra settings.py dosyasındaki INSTALLED_APPS'a "pgday.product" satırını eklemeniz gerekir.

markafoni
Her alışveriş bir aşk

Product/models.py

```python
# -*- coding: utf-8 -*-
from django.db import models
from django.conf import settings
import os

class Category(models.Model):
    name = models.CharField(max_length = 64)

    class Meta:
        db_table = 'mycategory'
        verbose_name_plural = "Categories"

class Product(models.Model):
    name = models.CharField(max_length = 128)
    barcode = models.CharField(max_length = 64)
    external_url = models.URLField(verify_exists = True)
    description = models.TextField(null = True, blank = True, verbose_name = u"Açıklama")
    is_active = models.BooleanField(default = True)
    created_at = models.DateTimeField(auto_now_add = True)
    categories = models.ManyToManyField(Category)
    stock_count = models.PositiveIntegerField()
    sold_count = models.PositiveIntegerField()
    image = models.ImageField(upload_to = os.path.join(settings.UPLOAD_ROOT, "product_image"))

    class Meta:
        ordering = ['-created_at', 'name']
        unique_together = ['name', 'barcode']

    def __unicode__(self):
        return "%s (%s)" % (self.name, self.barcode)

class Comment(models.Model):
    product = models.ForeignKey(Product)
    content = models.TextField()
    email = models.EmailField(max_length = 64)
    ip = models.IPAddressField()
    created_at = models.DateTimeField(auto_now_add = True)
    is_approved = models.BooleanField(default = False)
```

markafoni
Her alışveriş bir aşk

Product/admin.py

```
from django.contrib import admin
from pgday.product.models import Category, Product, Comment

admin.site.register(Category)
admin.site.register(Product)
admin.site.register(Comment)
```

markafoni
Her alışveriş bir aşk

Product/views.py

```python
# -*- coding: utf-8 -*-
from django.http import HttpResponseRedirect
from django.template import RequestContext
from django.shortcuts import get_object_or_404, render_to_response
from django.views.decorators.cache import cache_page
from django.core.urlresolvers import reverse
from django.contrib.auth.decorators import login_required, user_passes_test
from pgday.product.models import Product, Comment
from pgday.product.forms import CommentForm

@cache_page(300)
def product_list(request):
    products = Product.objects.filter(is_active = True).only("id", "name", "description")
    return render_to_response("product/productlist.html",
                              {"products" : products},
                              context_instance = RequestContext(request))

@login_required
def product_detail(request, product_id):
    product = get_object_or_404(Product, id = product_id)
    comments = product.comment_set.all()
    if request.POST:
        commentform = CommentForm(request.POST)
        if commentform.is_valid():
            commentform.save(request, product)
    else:
        commentform = CommentForm()
    return render_to_response("product/productdetail.html",
                              {"product" : product, "comments" : comments, "commentform" : commentform},
                              context_instance = RequestContext(request))

@user_passes_test(lambda u: u.is_staff or u.is_superuser)
def approve_comment(request, product_id, comment_id):
    comment = get_object_or_404(Comment, id = comment_id, product__id = product_id)
    comment.is_approved = True
    comment.save()
    return HttpResponseRedirect(reverse("product-detail", args=[product_id]))
```

Product/forms.py

```
# -*- coding: utf-8 -*-
from django import forms
from pgday.product.models import Comment

class CommentForm(forms.ModelForm):
    class Meta:
        model = Comment
        exclude = ["parent_comment", "product", "created_at", "ip", "is_approved"]

    def save(self, request, product):
        comment = Comment(product = product,
                          email = self.cleaned_data["email"],
                          content = self.cleaned_data["content"],
                          ip = request.META["REMOTE_ADDR"])
        comment.save()
        return comment
```

Urls.py & Product/urls.py

```
from django.conf.urls.defaults import patterns, include, url

from django.contrib import admin
admin.autodiscover()

urlpatterns = patterns('',
    url(r'^$', 'pgday.views.home', name='home'),
    url(r'^product/', include('pgday.product.urls')),
    url(r'^admin/', include(admin.site.urls)),
)
```

```
from django.conf.urls.defaults import patterns, include, url

urlpatterns = patterns('',
    url(r'^list/$', 'pgday.product.views.product_list', name='product-list'),
    url(r'^detail/(?P<product_id>\d+)/$', 'pgday.product.views.product_detail', name='product-detail'),
    url(r'^(?P<product_id>\d+)/approve/comment/(?P<comment_id>\d+)/$', 'pgday.product.views.approve_comment', name='approve-comment'),
)
```

markafoni
Her alışveriş bir aşk

Templates/product/productlist.html

```
{% if products %}
<ul>
  {% for product in products %}
    <li><a href="{% url product-detail product.id %}">{{product.name}}</a></li>
  {% endfor %}
</ul>
{% endif %}
```

markafoni
Her alışveriş bir aşk

Templates/product/productdetail.html

```
<table>
  <tr>
    <td>Ürün Adı</td>
    <td>{{product.name}}</td>
  </tr>
  <tr>
    <td>Ürün Açıklaması</td>
    <td>{{product.description}}</td>
  </tr>
</table>

{% if comments %}
<ul>
  {% for comment in comments %}
    <li>{{comment.content}}{% if not comment.is_approved %}<a href="{% url approve-comment product.id comment.id %}">Onayla</a>{% endif %}</li>
  {% endfor %}
</ul>
{% endif %}

<form action="{% url product-detail product.id %}" method="POST">
  {% csrf_token %}
  <table>
    {{commentform.as_table}}
    <tr>
      <td colspan="2"><input type="submit" value="Kaydet" /></td>
    </tr>
  </table>
</form>
```

markafoni
Her alışveriş bir aşk

Django ORM'si Nasıl Kullanılır?

```
Product.objects.filter(name__icontains = "Touch", is_active = True).exclude(stock_count = 0).order_by("-created_at")

SELECT "product_product"."id", "product_product"."name", "product_product"."barcode", "product_product"."external_url",
"product_product"."description", "product_product"."is_active", "product_product"."created_at",
"product_product"."stock_count", "product_product"."sold_count", "product_product"."image"
FROM "product_product"
WHERE (UPPER("product_product"."name"::text) LIKE UPPER(%Touch%)
AND "product_product"."is_active" = True
AND NOT ("product_product"."stock_count" = 0 ))
ORDER BY "product_product"."created_at" DESC
```

```
d = datetime.datetime.now() - datetime.timedelta(days = 3)
Product.objects.filter(Q(name__istartswith = 'Pu') | Q(created_at__gt = d)).only("id", "name")

SELECT "product_product"."id", "product_product"."name" FROM "product_product"
WHERE (UPPER("product_product"."name"::text) LIKE UPPER(Pu%)
OR "product_product"."created_at" > 2012-05-08 15:55:39.040286 )
ORDER BY "product_product"."created_at" DESC, "product_product"."name" ASC
```

```
Product.objects.filter(stock_count__lte = F('sold_count')).update(is_active = False)

UPDATE "product_product" SET "is_active" = false WHERE "product_product"."stock_count" <=  "product_product"."sold_count
```

Django ORM'si Nasıl Kullanılır? (2)

```
Product.objects.using("default").all().aggregate(Sum("sold_count"))

SELECT SUM("product_product"."sold_count") AS "sold_count__sum" FROM "product_product"


category, is_created = Category.objects.get_or_create(name = "Erkek")

SELECT "mycategory"."id", "mycategory"."name" FROM "mycategory" WHERE "mycategory"."name" = E\'Erkek\'
INSERT INTO "mycategory" ("name") VALUES (E\'Erkek\')'


active_products = Product.objects.filter(is_active = True).only("name")
active_products.count() -> 2
Product.objects.update(is_active = True)
active_products.count() -> 3


Comment.objects.select_related().filter(content__regex=r'[0-9]')

SELECT "product_comment"."id", "product_comment"."product_id", "product_comment"."content",
"product_comment"."email", "product_comment"."ip", "product_comment"."created_at",
"product_comment"."is_approved", "product_product"."id", "product_product"."name", "product_product"."barcode",
"product_product"."external_url", "product_product"."description", "product_product"."is_active",
"product_product"."created_at", "product_product"."stock_count", "product_product"."sold_count",
"product_product"."image" FROM "product_comment" INNER JOIN "product_product" ON
("product_comment"."product_id" = "product_product"."id") WHERE "product_comment"."content" ~ E\'[0-9]\'  LIMIT 21
```

markafoni
Her alışveriş bir aşk

Django'da Transaction Nasıl Yapılır?

Django her metod'un başında transaction'ı başlatır ve sonunda transaction'ı otomatik olarak commit eder.

Bunu ayarlamak istediğimiz durumlarda aşağıdaki örnekteki gibi transaction'ı manuel olarak commit etmemiz gerekir.

```
@transaction.commit_manually
def payment(request, order):
    try:
        order.close()
        transaction.commit()
    else:
transaction.rollback()
raise Exception()
```

markafoni
Her alışveriş bir aşk

Veritabanından Dump Data Nasıl Alınır?

```
python manage.py dumpdata –format json –indent 2 > initial_data.json
```

Veritabanından Alınan Data Nasıl Import Edilir?

```
python manage.py syncdb
```

markafoni
Her alışveriş bir aşk

Django1.4 ile Hangi Özellikler Gelmiştir?

```
product = Product.objects.select_for_update().get(id = 1)
product.sold_count += 2
product.save()

Category.objects.bulk_create([Category(name = u"Erkek"),
                              Category(name = u"Kadın"),
                              Category(name = u"Aksesuar")])

@sensitive_post_parameters("credit_card_number", "cvv2")
def payment(request):
```

CSRF koruması artık PUT ve DELETE metodlarını da kapsıyor.

markafoni
Her alışveriş bir aşk

Teşekkürler





/egehanoglu