

DJANGO HAQIDA MA'LUMOT

Jalolov Tursunbek Sadriddinovich

Osiyo Xalqaro Universiteti assitenti

Odilova Fotima Oltiboyevna

Osiyo Xalqaro Universiteti talabasi

Anatatsiya: Django tez va kengaytiriladigan veb-illovalarni yaratish uchun mo'ljallangan kuchli Python veb-ramkasidir. Ushbu maqola Django-ning asosiy xususiyatlarini, jumladan uning arxitekturasi, tamoyillari va komponentlarini qamrab oladi. MVC (Model-View-Controller) modeliga, marshrutlash tizimiga, shuningdek, o'rnatilgan xavfsizlik va boshqaruv funktsiyalariga alohida e'tibor beriladi. Xulosa Django-ning asosiy afzalliklari va cheklovlari, shuningdek, zamonaviy veb-dasturlashda mashhurligi haqida gapiradi.

KIRISH

Django veb-illovalarni ishlab chiqishni tezlashtirish uchun mo'ljallangan yuqori darajadagi Python veb-ramkasidir. "Batareyalar mavjud" falsafasi bilan ishlab chiqilgan bo'lib, u ishlab chiquvchilarga xavfsiz, kengaytiriladigan va ishonchli veb-illovalarni yaratish uchun ko'plab o'rnatilgan vositalar va xususiyatlarni taqdim etadi. Django avtomatik ravishda kod ishlab chiqarish va komponentlarni qayta ishlatish va qo'l mehnatini minimallashtirish kabi zamonaviy rivojlanish tamoyillaridan foydalanish orqali tez rivojlanishni qo'llab-quvvatlaydi.

Django-ning asosiy maqsadi ishlab chiquvchilarga odatiy vazifalarga emas, balki ilovaning biznes mantig'iga e'tibor qaratish imkonini berishdir. ORM (Ob'ektga aloqador xaritalash), autentifikatsiya va boshqaruv tizimi kabi ko'plab o'rnatilgan xususiyatlar tufayli ramka oddiy va murakkab veb-illovalarni yaratish uchun eng mashhur vositalardan biriga aylandi.

Asosiy qism

1. Django Architecture

Django mashhur MVC (Model-View-Controller) modeliga o'xshash Model-View-Template (MVT) arxitektura modelidan foydalanadi. MVT ning asosiy tarkibiy qismlari:

Model ma'lumotlar bilan ishlash va ma'lumotlar bazasi bilan o'zaro aloqa qilish uchun mas'ul komponent hisoblanadi. Django ma'lumotlar bazasi so'rovlarini avtomatlashtirish uchun o'rnatilgan ORM dan foydalanadi, bu esa ishlab chiquvchilarga Python sinflari orqali ma'lumotlar bilan ishlash imkonini beradi.

Ko'rish - ilovaning biznes mantig'ini o'z ichiga oladi va foydalanuvchiga qanday ma'lumotlarni uzatish va uni qanday qayta ishlash kerakligini belgilaydi. Django da ko'rinish turli formatlardagi (HTML, JSON va boshqalar) ma'lumotlarni o'z ichiga olgan HTTP javobini qaytaradi.

Shablon - HTML sahifalar ko'rinishidagi ma'lumotlarni ko'rsatish uchun javobgar. Django o'zining shablon tizimidan foydalanadi, bu mantiq va ma'lumotlarni ko'rsatishni ajratishni osonlashtiradi.

2. Django asosiy komponentlari

2.1. Django ORM

Django ning asosiy xususiyatlaridan biri uning ORM (Object-Relational Mapping) dir - bu tizim SQL so'rovlarini bevosita ishlatmasdan, Python ob'ektlari orqali ma'lumotlar bazalari bilan ishlash imkonini beradi. ORM avtomatik ravishda modellar (sinflar) asosida SQL so'rovlarini yaratadi va ma'lumotlar boshqaruvini soddalashtiradi, ma'lumotlar bazalari bilan ishlashni yanada intuitiv va xavfsiz qiladi.

Djangodagi oddiy modelga misol:

```
python
```

```
django.db import
```

```
sinf kitobi(modellar.Model):
```

```
sarlavha = modellar.CharField (maksimal_uzunlik=200)
```

```
muallif = models.CharField (maksimal_uzunlik=100)
```

```
Published_date = models.DateField()
```

Keyinchalik, SQL so'rovlarini qo'lda yozmasdan turib, `.save()`, `.filter()`, `.all()` va boshqalar kabi usullar orqali ma'lumotlar bazasi bilan o'zaro aloqada bo'lishingiz mumkin.

2.2. Marshrutlash va URL konfiguratsiyasi

Django-da marshrutlash `urls.py` fayllari orqali boshqariladi, ular qaysi ko'rinishlar qaysi URL-lar uchun so'rovlarni bajarishini belgilaydi. Bu URL manzillarni muayyan funksiyalar yoki ko'rish sinflari bilan bog'lashni osonlashtiradi.

URL manzilini sozlash misoli:

```
python
```

```
django.urls import yo'lidan
```

```
dan. ko'rishlarni import qilish
```

```
urlpatterns = [
```

```
path(books/, views.book_list, name=book_list),
```

```
path(books/<int:id>/, views.book_detail, name=book_detail),
```

```
]
```

2.3. Django shablon tizimi

Django shablonlari HTML tarkibini yaratish va ma'lumotlarni dinamik ravishda ko'rsatish uchun o'rnatilgan teglar va filtrlarni qo'llab-quvvatlash uchun ishlatiladi. Shablonlarning asosiy afzalligi ularning soddaligi va moslashuvchanligidir: ishlab chiquvchilar mantiqni displeydan ajratib, sahifalarni osongina yaratishi va o'zgartirishi mumkin.

Oddiy shablonga misol:

```
html
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
<title>Kitoblar ro'yxati</title>
</head>
<tana>
<h1>Kitoblar</h1>
<ul>
{% kitoblardagi kitob uchun %}
<li>{{ book.title }} muallifi {{ book.author }}</li>
{% end for %}
</ul>
</body>
</html>
```

2.4. O'rnatilgan xavfsizlik xususiyatlari

Django xavfsizlikka katta e'tibor beradi. U keng tarqalgan hujumlarga qarshi o'rnatilgan himoya mexanizmlariga ega, masalan:

Saytlararo skript (XSS): Shablonlar sukut bo'yicha zararli skript kiritishdan himoya qilish uchun ma'lumotlardan qochishadi.

Saytlararo so'rovlarni qalbakilashtirish (CSRF): so'rovlarni soxtalashtirishdan o'rnatilgan himoya.

SQL in'ektsiyasi: ORM dan foydalanish avtomatik ravishda SQL in'ektsiyasini oldini oladi.

Parolni shifrlash: Django foydalanuvchi parollarini himoya qilish uchun xavfsiz xeshlash usullaridan foydalanadi.

2.5. Django boshqaruv paneli

Django-ning eng jozibali jihatlaridan biri uning avtomatlashtirilgan boshqaruv panelidir. Ushbu panel amaliy modellar asosida yaratilgan va ma'lumotlarni boshqarish uchun qulay interfeysni ta'minlaydi. U CRUD operatsiyalarini (yaratish, o'qish, yangilash, o'chirish) qo'shimcha kod yozishni talab qilmasdan amalga oshirish imkonini beradi.

Administrator paneliga model qo'shishga misol:

```
python
django.contrib import administratoridan
dan .models import Kitob
admin.site.register(Kitob)
```

3. Djangoning afzalliklari va kamchiliklari

Afzalliklar:

Tez ishlab chiqish: Ko'plab o'rnatilgan vositalar va odatiy vazifalarni avtomatlashtirish tufayli Django sizga ilovalarni tezroq yaratishga imkon beradi.

- Xavfsizlik: Django ilovalarni umumiy tahdidlardan himoya qilish uchun ajoyib vositalarni taqdim etadi.

- Scalability: Django ham kichik loyihalarni, ham murakkab korporativ ilovalarni qo'llab-quvvatlaydi.

- Hamjamiyat va hujjatlar: resurslarning katta tanlovi, batafsil hujjatlar va faol hamjamiyat.

Kamchiliklar:

- Kichik loyihalar uchun murakkab: Kichik ilovalar uchun Django tuzilishi va ko'plab o'rnatilgan funksiyalari tufayli juda og'ir ko'rinishi mumkin.

- Slow versiya yangilanishlari: Ramka barqaror bo'lsa-da, yangi versiyalarga yangilash qo'shimcha harakat talab qilishi mumkin.

Xulosa.

Django oddiy veb-saytlar yaratish hamda murakkab korporativ tizimlarni ishlab chiqish uchun mos bo'lgan veb-ishlab chiquvchilar uchun kuchli va qulay vositadir. Bu sizga ko'plab o'rnatilgan funktsiyalar va vazifalarni avtomatlashtirish bilan ishlab chiqish jarayonini tezlashtirishga imkon beradi, shu bilan birga yuqori darajadagi xavfsizlik va kengayishni ta'minlaydi. Django rivojlanishda davom etmoqda va afzalliklari va faol hamjamiyat tufayli veb-ishlab chiquvchilar orasida mashhur tanlov bo'lib qolmoqda. Biroq, kichikroq loyihalar uchun Flask kabi engilroq ramkalar afzalroq bo'lishi mumkin. Biroq, Django xavfsizlik, ishonchlilik va rivojlanish tezligiga e'tibor qaratgan holda ilovalarni ishlab chiqish uchun ajoyib tanlov bo'lib qolmoqda.

FOYDALANILGAN ADABIYOTLAR RO'YXATI:

1. Sadriddinovich, J. T. (2024). BASICS OF PSYCHOLOGICAL SERVICE. PSIXOLOGIYA VA SOTSIOLOGIYA ILMIY JURNALI, 2(4), 61-67.

2. Jalolov, T. S. (2024). PYTHONNING MATEMATIK KUTUBXONALARINI O'RGANISH: KENG QAMROVLI QO'LLANMA. BIOLOGIYA VA KIMYO FANLARI ILMIY JURNALI, 2(5), 71-77.

3. Django rasmiy hujjatlari. <https://www.djangoproject.com/>

4. Yangi boshlanuvchilar uchun Django, Uilyam S. Vinsent.

5. Djangoning ikki qoshig'i, Audrey Roy Grinfeld, Daniel Roy Grinfeld.

6. Django REST Framework Documentation. <https://www.django-rest-framework.org/>