

Software Engineering

Eng: Malek Almosanif

Eng: Fadwa Fuad Ali

Django Template Engine :

هي نظام يُستخدم لعرض البيانات بطريقة منظمة ومنسقة في واجهات HTML ويفصل منطق البرمجة (Backend) وتصميم العرض (frontend).

كــ تعريف أــ حق:

هو محرك قوالب فــ Django يُستخدم لأنشاء صفحات HTML ديناميكية عن طريق دمج ملفات HTML مع البيانات التي يرسلها الــ View.

استخراج ماقبة:

- لعرض البيانات القادمة من الـ database على شكل صفحات HTML.
- لفصل منطق البرمجة (Python) عن تنسيق الواجهة (HTML).
- لتكرار عناصر الصفحة (مثل الجداول، القوائم) بطريقة ديناميكية دون كتابة نفس الكود مراراً.
- ل توفير وسائل تحكم مثل if, for, filters, وغيرها.

كيفية عملة:

- إنشاء ملف HTML يحتوي على تعليمات Django (وليس HTML فقط..)
- يرسل البيانات إلى هذا الملف View
- يدمج البيانات مع ملف Template Engine ويعرضها للمستخدم.

- النوع الافتراضي في Django .
- خفيف، بسيط، ومصمم ليتكامل مباشرة مع Django .
- يستخدم { ... } % للتعليمات، و { ... } لعرض البيانات
- يدعم الوراثة بين القوالب، الفلاتر، التكرار، الشروط، وغيرها.
- المفضل للمشاريع اللي تبغين تسويها بسرعة وببساطة..

Jinja2

- محرك قوالب خارجي، لكنه مدعوم رسمياً من Django
- أسرع من DTL في الأداء.
- صمم ليكون أكثر مرونة ويشبه Python أكثر.
- تنفيذ عمليات أكثر تعقيداً.
- استخدام هيكل بابيثونية داخل القوالب.

mako

- سريع جدًا ويعتمد على بايثون.
- منken أقل تكاملاً مع . Django

Chameleon

- يعتمد على XML.
- ممتاز للأداء لكن نادر الاستخدام.

أُبَرِزُ الْأَنْخَتْلَانْ فَلَتْ

نوع المحرك	سرعةه	مدى السهولة	تكاملة مع Django	هيئات	
DTL	جيده	سهل للغاية	مدفع معه	جيد للفصل بين الواجهة والمنطق	الافتراضي.
JINJA2	اسرع من Dtl	متوسطة السهولة.	مدعوم	أقرب لبايتون، فلاتر قوية	يستخدم في Flask

هناسب لـ الـأداء العـالـي	عنـ جـدا وـقـوي	ـحتاج تـكامـال ـيدـوي	. عـقدـ.	ـسـريع جـدا	MAKO
يـسـتـخـدـمـ فـي بعـضـ أـنـظـهـهـ CMS	ـوـتـازـفـيـ الـأـدـاءـ	ـتكـامـلـةـ مـتوـسـطـاـ	ـلاـيـعـتـبـرـ بـسيـطـاـ ـلـانـهـ يـعـتـهـدـ عـلـىـ XML	ـسـريعـ جـداـ	CHAMELEON
غـيرـ مـفـضـلـ حالـياـ	ـيـشـبـهـ أـسـلـوبـ ـالـبـرـجـةـ ـالـكـلاـسيـكـيـةـ	ـلـاـيـدـعـهـ رـسـيـاـ	ـصـعبـ وـقـدـ حـمـ.	ـبـطـيءـ	CHEETAH

فلدر نر انقوابب في

Django Template Language

(DTL)

هي أدوات قوية تساعد على تنسيق البيانات
أو تعدلها داخل القالب قبل عرضها
للمستخدم، وكل هذا بدون كتابة كود بايثون
في الواجهة.

أنواع الفلدرات في DTL

تنسيق النصوص

أو القيم.

فلدرات
التنسيق

تعديل أو معالجة
القوائم أو السلسلة
النصية.

فلاتر القوائم
والسلسلة

إعطاء قيمة بديلة
أو فحص القيمة.

فلدقر الشرطة
والتتعامل مع الواقع

**لُسْبِقُ التَّوَارِيخِ
وَالْأَوْقَانَ**

**فَلَدْ قَرُ الْتَّارِيخِ
وَالزَّمْنَ**

**التعامل مع HTML
والنصوص البرمجية.**

**فلاتر الأماكن
والعرض**

نوع الفلتر	يفيد في	—
التنسيق	جعل المخرجات مرتبة للمستخدم	سريع وبسيط
القواعد والسلسل	التعامل مع بيانات متكررة	هناز للجدال والقوائم
القيم الشرطية	تقديم بدائل ذكية عند نقص البيانات.	مفید في التعامل مع البيانات غير الكتلية
التاريخ والزمن	تنسيق مخرجات التواریخ بشكل انيق	مفید للهداونات والواقع الزمنية
الأمان والعرض	حماية او تزيين النصوص البرمجية.	HTML Markdown

فلتری الخواص

- فلتر يقوم بعمل "—" مكان كل فراغ.
- فلتر يقوم بتلوين اسم "الله" باللون الأحمر.

كُوٰہ ڪرنسن سماپقا فی انشاء المشاريع

- انشاء مشروع جديد "fadwaspro2" فية تطبيق جديد.
- اضافه او تسجيل التطبيق fadwa في settings.py
- عمل url باسم url
- عمل ملف للتطبيق با اسم urls.py و اضافه اول رابط views

إنشاء الفلتر

- إنشاء ملف جديد في التطبيق ب اسم `templatetags`
- اضافه الفلاتر الي نصنعها داخل مجلد `templatetags` ب اسم `"fadwa_filter1" "fadwa_filter2"`

بـ حـاـخـنـ fadwa_filter1

```
from django import template  
  
import re  
  
register = template.Library()  
  
(@register.filter(name='fadwa_filter1')  
  
Def f1(value):  
  
if isinstance(value, str):  
  
return value.replace(' ','_')  
  
return value
```

بـ ڦاڱر ڦاڱر

```
from django import template  
import re  
  
register = template.Library()  
  
(@register.filter(name='fadwa_filter1')  
Def f1(value):  
if isinstance(value, str):  
return re.sub(r'\d', '*', value)  
return value
```

بـ آخر fadwa_filter3

```
from django import template  
import re  
  
register = template.Library()  
  
(@register.filter(name='fadwa_filter1')  
  
Def f1(value):  
  
if isinstance(value, str):  
  
return re.sub(r'()',r'<span style=color:red;">\1</span>',value)  
  
return value
```

استخدام الفلتر

- نحمل الفلتر بداخل القالب

```
{%fadwa_filter1 load %} •
```

استخدام الفلتر على البيانات

```
<p>{["nice filter" | fadwa_filter1}</p>  
<p>{["nice filter1 and2" | fadwa_filter2}</p>
```

و مهلاً نجوة و انشكر ...