

## تكميل: مقارنة أنظمة القوالب المشابهة لـ Django Template Language (DTL)

### المقدمة

يهدف هذا التكميل إلى دراسة ومقارنة أربعة أنظمة قوالب (Template Engines) مشابهة في آلية عملها لنظام القوالب الخاص بـ Django، المعروف باسم **Django Template Language (DTL)**، مع تقديم أمثلة عملية وشرح معمق للفروقات الجوهرية بينها.

### أنظمة القوالب المختارة للمقارنة

1. **Jinja2** - لغة Python (يستخدم مع Flask وFastAPI).
2. **Twig** - لغة PHP (يستخدم في Laravel وSymfony).
3. **EJS** - لغة JavaScript (يستخدم في Node.js وExpress).
4. **Handlebars.js** - لغة JavaScript (يستخدم في تطبيقات الويب).

#### 1. Jinja2

- **الوصف:** محرك قوالب بلغة Python، يعتمد على نفس بنية DTL تقريبًا، ويدعم الوراثة والفلاتر.
- **مثال:**

```
<h1>{{ title }}</h1>
{% if user %} مرحبًا {{ user.name }}{% endif %}
```

- **الفروقات بعمق:**
- أسرع من DTL في معالجة القوالب.
- يدعم قدرات أوسع في الفلاتر والتخصيص.
- قابلية الدمج مع تطبيقات Python غير Django.
- صياغته مشابهة لـ DTL، ما يسهل الانتقال بينهما.

#### 2. Twig

- **الوصف:** محرك قوالب بلغة PHP، يتميز بالفلاتر المتقدمة والماكروز لإعادة استخدام الأكواد.
- **مثال:**

```
<h1>{{ title }}</h1>
{% if user %}مرحبًا {{ user.name }}{% endif %}
```

#### الفروقات بعمق:

- يعمل حصريًا مع بيئة PHP.
- يدعم الماكروز التي تسهل إعادة استخدام المكونات.
- أمان مدمج ضد حقن الأكواد الضارة.
- قدرات تخصيص أوسع من DTL خاصة في بيئة Laravel.

### EJS .3

- **الوصف:** محرك قوالب يدمج JavaScript داخل HTML مباشرة، مناسب لتطبيقات Node.js.
- **مثال:**

```
<h1><%= title %></h1>
<% if (user) { %>مرحبًا <%= user.name %><% } %>
```

#### الفروقات بعمق:

- يمكن كتابة أي كود JavaScript داخل القالب.
- مرونة عالية جدًا لكن الأمان أقل إذا لم تُعقم البيانات.
- الصياغة تختلف جذريًا عن DTL (تعتمد <% %> بدل (% %)).
- مثالية للمشاريع التي تعتمد على Node.js كمحرك رئيسي.

### Handlebars.js .4

- **الوصف:** محرك قوالب آمن وبسيط يمنع تنفيذ كود JavaScript مباشرة.
- **مثال:**

```
<h1>{{title}}</h1>
{{#if user}}مرحبًا {{user.name}}{{/if}}
```

#### الفروقات بعمق:

- يمنع إدخال أي أكواد برمجية داخل القالب مما يجعله أكثر أمانًا.
- المنطق محدود، ويعتمد على الـ helpers لإضافة وظائف متقدمة.
- مناسب للمشاريع التي تحتاج واجهات آمنة وخالية من التعقيد.

## جدول المقارنة

المعيار	DTL (Django)	Jinja2	Twig	EJS	Handlebars
اللغة	Python	Python	PHP	JavaScript	JavaScript
القوس المستخدم	{% , {{ }} %}	{% , {{ }} %}	{% , {{ }} %}	<% , <%= %> %>	{{ }}
دعم المنطق if/for	نعم	نعم	نعم	نعم	نعم
تنفيذ كود برمجي	لا	لا	لا	نعم	لا
الأمان	عالي	عالي	عالي	متوسط	عالي
المرونة	متوسطة	عالية	عالية	عالية	متوسطة
بيئة الاستخدام	/Django Python	عام Python	PHP (Laravel...)	/Node.js Express	أي تطبيق ويب

## الخاتمة

جميع هذه الأنظمة تشبه DTL في فكرة دمج المتغيرات والمنطق داخل HTML، لكن اختيار الأنسب يعتمد على:

- **Python** ➤ Jinja2 لمرونته وتوافقته.
- **PHP** ➤ Twig لقوة التخصيص والأمان.
- **Node.js** ➤ EJS إذا كانت الأولوية للمرونة، أو Handlebars إذا كانت الأولوية للأمان.