

# **Universidad autónoma de Baja California**

Facultad de Contaduría y Administración

Materia: Python

Grupo: 372

Alumno: López Gutiérrez Oscar

- Plantillas

Django posee un componente conocido como “el motor de plantillas”, este motor brinda un poderoso mini-lenguaje para definir detalles de la capa de la aplicación, que visualizará el usuario. Esto refuerza la separación de la lógica de programación y la lógica de presentación.

Las plantillas de Django presentan una característica importante que se llama etiqueta de plantilla. En HTML no se puede escribir código en Python porque los navegadores no lo entienden. Sólo saben HTML. Sabemos que HTML es bastante estático, mientras que Python es mucho más dinámico.

Las **etiquetas de plantilla de Django** nos permiten insertar elementos de Python dentro del HTML.

La configuración de **TEMPLATES** de su proyecto describe cómo Django cargará y creará las plantillas. El archivo de configuración predeterminada configura un backend **DjangoTemplates** cuya opción **APP\_DIRS** está configurada como **True**. Convencionalmente, **DjangoTemplates** busca un subdirectorio «templates» en cada una de las **INSTALLED\_APPS**.

Para imprimir una variable en una plantilla de Django, utilizamos llaves dobles con el nombre de la variable dentro, algo así:

```
{{ posts }}
```

Pero suponiendo que se necesita crear un ciclo for para desplegar la información que está en posts porque “posts” en una lista, lo que se puede hacer es lo siguiente:

```
{% for post in posts %} {{ post }}{% endfor %}
```

Todo lo que pongas entre `{% for %}` y `{% endfor %}` se repetirá para cada objeto de la lista, esto es porque al poner `{% %}` le estamos indicando que adentro de esas llaves se ejecutara código de Python.

Django puede utilizar algo llamado Context. Un contexto es simplemente un conjunto de variables y sus valores asociados. Una plantilla usa estas variables para llenar y evaluar estas etiquetas de bloque.

Un contexto es representado en Django por la clase `Context`, ésta se encuentra en el módulo `django.template`. Su constructor toma un argumento opcional: un diccionario que mapea nombres de variables con valores.

Bibliografía:

<https://docs.djangoproject.com/es/2.1/intro/tutorial03/>

[https://tutorial.djangogirls.org/es/django\\_templates/](https://tutorial.djangogirls.org/es/django_templates/)