Django Template Tags (By DTL)

Let's say template tags are talking about the correct syntax that is to be used from Python inside HTML. You can't just put any Python sh*t you know between two curly braces and get away with it! There is significant structure to be used in there.

Let's start simple, just like the project at chapter 7, we have and app called "app1" That has a view which is meant to send some Python list items back to client side inside an HTML file. Let's take a look at the codes.

```
#/MyProject/app1/views.py
from django.shortcuts import render

def index(request):
    data = {
        "HoHoHo": ["Santa", "Christmas", "Candy"] # Yes my child it's
Christmas at the time I'm writing this!
    }
    return render(request, "app1/index.html", context=data)

<!-- /MyProject/app1/templates/app1/index.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Django Template Tags</title>
</head>
<body>
    <ul>
        {% for word in  HoHoHo %}
        <li>
            {{ word }}
        </li>
        {% endfor %}
    </ul>
</body>
</html>
```

Add the files and give it a test! we'll learn more deeply about DTT from this point on.

URLs in DTT

You remember the unique name we could give to specific urls like this: `path('url', views.index, name="unique_name")`? well we can use that unique name in HTML files too!

```
<body>
    <a href="{% url 'unique_name' %}">
        Click Me!
    </a>
</body>
</html>
```

if statement in DTT

Obvious by the name, you can use conditional statements inside HTML with DTT. The example works with the case where given data is `None`.

```
<body>
    {% if data is not None %}
    <p>{{ data }}</p>
    {% else %}
    <p>Data not available!</p>
    {% endif %}
</body>
</html>
```