

## What is Django Template Language? Why it is need in Django Web application Development?

**Django Template Language (DTL)** is a powerful templating engine provided by Django to dynamically generate HTML, XML, or other text-based formats. It allows you to embed Python-like expressions inside HTML code to display data, control flow, and reuse code efficiently.

### **Dynamic Content Rendering:**

It helps in rendering dynamic content like user data, product lists, etc., without hardcoding them into HTML.

### **Separation of Concerns (SoC):**

Keeps the business logic (in views/models) separate from presentation logic (in templates), following the MVC (Model-View-Controller) architecture.

### **Security:**

By default, DTL escapes variables to prevent **Cross-Site Scripting (XSS)** attacks.

### **Code Reusability:**

Supports template inheritance, making it easy to reuse layouts (e.g., base templates) across different pages.

### **Built-in Filters and Tags:**

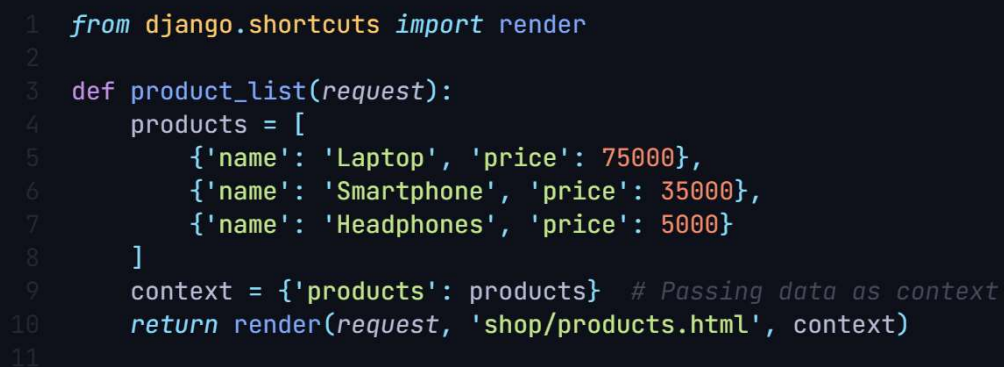
Provides filters (like date, upper, length) and tags (for, if, include) to manipulate data directly within templates.

## How do you pass context variables from a Django view to a template, and how do you access them in the template?

In Django, you pass data from a view to a template using a **context dictionary**. This dictionary contains key-value pairs where the key is the variable name accessible in the template, and the value is the actual data.

### 1. Example: Passing Data from View to Template

views.py



```
1 from django.shortcuts import render
2
3 def product_list(request):
4     products = [
5         {'name': 'Laptop', 'price': 75000},
6         {'name': 'Smartphone', 'price': 35000},
7         {'name': 'Headphones', 'price': 5000}
8     ]
9     context = {'products': products} # Passing data as context
10    return render(request, 'shop/products.html', context)
11
```

## 2. Accessing Context Variables in the Template

templates/shop/products.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Product List</title>
5 </head>
6 <body>
7   <h1>Available Products:</h1>
8   <ul>
9     {% for product in products %}
10      <li>{{ product.name }} - ₹{{ product.price }}</li>
11     {% empty %}
12      <li>No products available.</li>
13     {% endfor %}
14   </ul>
15 </body>
16 </html>
17
```