31 December 2019

Django Messages

Quite commonly in web applications, you need to display a one-time notification message (also known as "flash message") to the user after processing a form or some other types of user input.

For this, Django provides full support for cookie- and session-based messaging, for both anonymous and authenticated users. The messages framework allows you to temporarily store messages in one request and retrieve them for display in a subsequent request (usually the next one). Every message is tagged with a specific level that determines its priority (e.g., info, warning, or error).

```python
from django.contrib.messages import constants as messages

MESSAGE_TAGS = {

messages.INFO: '',

    50: 'critical',
}
```

To add a message in the view

```python
from django.contrib import messages

messages.add_message(request, messages.INFO, 'Hello world.')
```

Displaying Messages in the template

```django
{% if messages %}
<ul class="messages">
```

```
{% for message in messages %}
<li{% if message.tags %} class="{{ message.tags }}"{% endif %}>{{ message }}</li> {%
endfor %}

</ul>
{% endif %}
```

Outside of template

We can use get_messages(request) to get all the messages.

Adding messages to CBVs

We need to use a Mixin to add messages and set the name of the message in the attributes.

from django.contrib.messages.views import ___Mixin Name____