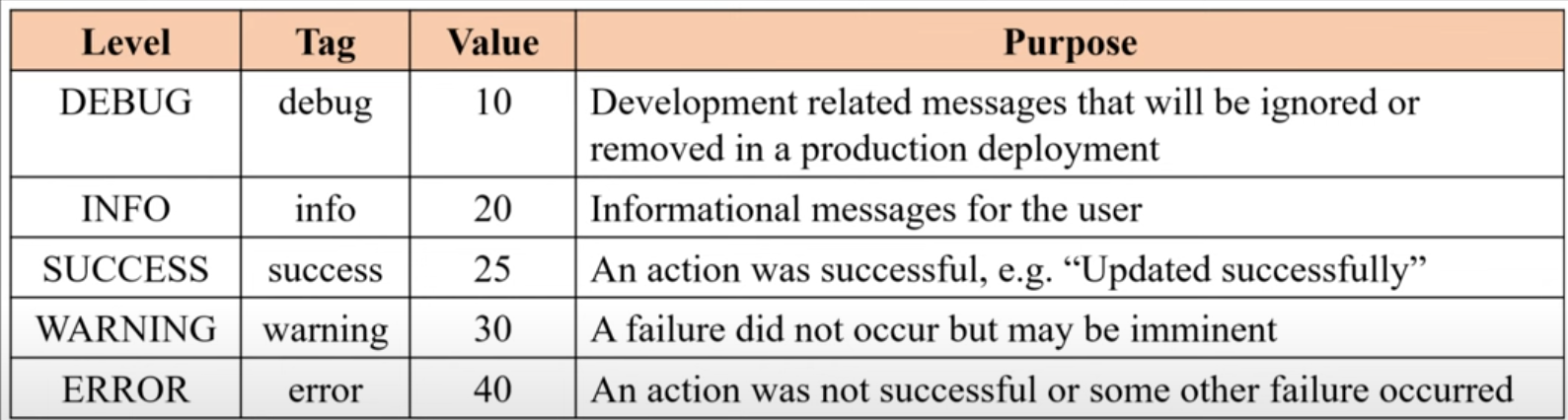
**Display Success and Error Messages in Django 5**



From this list,we only get the values above or equal 20.

# Display Success and Error Messages in Django 5

## 1. Setup in views.py

from django.shortcuts import render

from django.contrib import messages

# Create your views here.

def home(request):

    # ✅ Method 1: Using add\_message()

    messages.add\_message(request, messages.SUCCESS, 'Your message has been successfully created!')

    messages.add\_message(request, messages.INFO, 'Your message contains some information')

    messages.add\_message(request, messages.WARNING, 'Your message has some warning')

    messages.add\_message(request, messages.ERROR, 'Your message has some error!')

    # ✅ Method 2: Direct shortcut functions

    messages.success(request, 'This is success')

    messages.info(request, 'This is info')

    messages.warning(request, 'This is warning')

    messages.error(request, 'This is error')

    messages.debug(request, 'This is debug')

    # ✅ Checking and changing message level

    print(messages.get\_level(request))   # Default = 20 (INFO)

    messages.set\_level(request, messages.DEBUG)  # Now debug messages will also display

    messages.debug(request, 'This is debug after set !!!')

    print(messages.get\_level(request))

    return render(request, 'student/home.html')

def registration(request):

    return render(request, 'student/registration.html')

## 2. Template Example (home.html)

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Home</title>

</head>

<body>

    <h1>Home PAGE</h1>

    <!-- ✅ Displaying Messages -->

    {% if messages %}

        {% for message in messages %}

            <!-- Normal usage -->

            <span class="{{ message.tags }}">{{ message }}</span> <br>

            <!-- Bootstrap integration (alert classes) -->

            <span class="alert alert-{{ message.tags }}">{{ message }}</span> <br>

        {% endfor %}

    {% endif %}

</body>

</html>

Another way using another bootstarp class:

{% if messages %}

  {% for message in messages %}

      <div class="alert alert-{{ message.tags }} alert-dismissible fade show">

          {{ message }}

          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>

      </div>

  {% endfor %}

{% endif %}

## 3. Changing Default Class Names (Custom Tags)

By default Django uses tags like:

* success → success
* error → error

but in bootstarp class are like

* success → alert-success
* error → alert-error

now for that we can use like this in html for bootstarp handing

            <!-- Bootstrap integration (alert classes) -->

            <span class="alert alert-{{ message.tags }}">{{ message }}</span>

But in bootstarp have no error class in bootstarp have danger in that so for that we can modification in settings.py

We can override them in **settings.py**.

from django.contrib.messages import constants as messages\_s

MESSAGE\_TAGS = {

    messages\_s.ERROR: 'danger',   # Now ERROR → 'danger'

}

👉 After this, in template it will render class alert-danger instead of alert-error.

## 4. Notes & Tips

* **Message Levels**:
  + DEBUG = 10
  + INFO = 20
  + SUCCESS = 25
  + WARNING = 30
  + ERROR = 40
* **Default Level** = INFO (20), so only messages ≥ 20 are shown.
* You can change with messages.set\_level(request, messages.DEBUG) to include lower ones.

TER THAT OUR ERROR CLASS CHANGE INTO DANGER NAME

Complete example

# 📘 Display Success and Error Messages in Django 5 (Complete Example)

## 1. **models.py**

from django.db import models

# Create your models here.

class User(models.Model):

    name = models.CharField(max\_length=100)

    email = models.EmailField(max\_length=100)

    passward = models.CharField(max\_length=100)

## 2. **forms.py**

from django import forms

from student.models import User

class StudentRegistration(forms.ModelForm):

    class Meta:

        model = User

        fields = ['name', 'email', 'passward']

## 3. **views.py**

from django.shortcuts import render

from django.contrib import messages

from student.forms import StudentRegistration

# Home view (optional messages examples)

def home(request):

    # Example ways to add messages:

    # messages.add\_message(request, messages.SUCCESS, 'Your message has been successfully created!')

    # messages.success(request, 'This is success')

    # messages.info(request, 'This is info')

    # messages.warning(request, 'This is warning')

    # messages.error(request, 'This is error')

    # messages.debug(request, 'This is debug')

    return render(request, 'student/home.html')

# Registration view

def registration(request):

    if request.method == 'POST':

        fm = StudentRegistration(request.POST)

        if fm.is\_valid():

            fm.save()

            # ✅ Success message after form submission

            messages.success(request, 'Form Successfully Submitted!')

        else:

            # ✅ Error message if validation fails

            messages.error(request, 'There was an error in the form. Please check again.')

    else:

        fm = StudentRegistration()

    return render(request, 'student/registration.html', {'fm': fm})

## 4. **registration.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Registration</title>

    <!-- Bootstrap for better alerts (optional) -->

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css">

</head>

<body>

    <h1>Registration Page</h1>

    <form action="" method="POST" novalidate>

        {% csrf\_token %}

        {{ fm.as\_p }}

        <input type="submit" value="Submit">

    </form>

    <!-- ✅ Displaying Messages -->

    {% if messages %}

        {% for message in messages %}

            <div class="alert alert-{{ message.tags }}">

                {{ message }}

            </div>

        {% endfor %}

    {% endif %}

</body>

</html>

## 5. **Customizing Message Tags**

In **settings.py**:

from django.contrib.messages import constants as messages\_s

MESSAGE\_TAGS = {

    messages\_s.ERROR: 'danger',  # Changes ERROR into 'danger' (Bootstrap class)

}

👉 Now error messages will render as alert-danger.

## 6. **Message Levels**

| **Level** | **Value** |
| --- | --- |
| DEBUG | 10 |
| INFO | 20 |
| SUCCESS | 25 |
| WARNING | 30 |
| ERROR | 40 |

👉 By default Django shows **20 and above** (INFO, SUCCESS, WARNING, ERROR).  
👉 To include **DEBUG**, use:

messages.set\_level(request, messages.DEBUG)

✅ With this setup, whenever the form is submitted successfully, a **Success message** is shown, and if the form is invalid, an **Error message** is displayed.