

# Django delete view

## Django Delete View

- Django delete view used to delete a record from the database through user interaction.
- It identifies the object based on the primary key.

### function based delete view

- Look at the code below

```
from django.http import HttpResponseRedirect
from django.shortcuts import render, get_object_or_404
from my_app.models import Contact

def fbv_delete_view(request, pk):
    obj = get_object_or_404(Contact, id=pk)
    if request.method.lower() == 'post':
        obj.delete()
        return HttpResponseRedirect('Delete success')
    context = {'object': obj}
    return render(request, 'delete_template.html', context)
```

- For GET request, it shows info about the deleted object.
- Once user confirms the delete with POST request then it deletes the object and returns the response.

delete\_template.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Delete Object</title>
</head>
<body>
    <h1>Delete Object</h1>
    <p>Are you sure you want to delete the object with the following
details?</p>
    <p>Object ID: {{ object.id }}</p>
    <p>Object Name: {{ object.name }}</p>
```

```

    <!-- Add more details as needed -->

    <form action="{% url 'fbv_delete_view' object.id %}" method="post">
        {% csrf_token %}
        <button type="submit">Delete</button>
    </form>
</body>
</html>

```

## class based delete view

- Django provides the generic view DeleteView to delete object.
- Look at the code below.

```

from django.http import HttpResponseRedirect
from django.views.generic.edit import DeleteView
from my_app.models import Contact

class CBVDeleteView(DeleteView):
    pk_url_kwarg = 'pk'
    queryset = Contact.objects.all()
    template_name = 'delete_template.html'

    def form_valid(self, form):
        self.object = self.get_object()
        self.object.delete()
        return HttpResponseRedirect('Delete success')

```

- It works same as the function based delete view implemented above.
- pk\_url\_kwarg - to identify the object or the record
- Attributes queryset or model or method get\_object() is required to get the object.

## configure urls

- open my\_app/urls.py and add below code to it.

```

from django.urls import path
from my_app import views

urlpatterns = [
    # ...
    path('fbv_delete_view/<int:pk>', views.fbv_delete_view,
name='fbv_delete_view'),
    path('cbv_delete_view/<int:pk>', views.CBVDeleteView.as_view(),
name='cbv_delete_view'),
    # ...
]

```

- Open url [http://127.0.0.1:8000/my\\_app/fbv\\_delete\\_view/1](http://127.0.0.1:8000/my_app/fbv_delete_view/1) to delete the object using function based delete view.
- Open url [http://127.0.0.1:8000/my\\_app/cbv\\_delete\\_view/1](http://127.0.0.1:8000/my_app/cbv_delete_view/1) to delete the object using class based delete view.
- Replace the 1 with any id in the contact table.
- It throws 404 status code if id does not exist in the table.

## References¶

- <https://docs.djangoproject.com/en/dev/ref/class-based-views/generic-editing/#django.views.generic.edit.DeleteView>
- <https://ccbv.co.uk/projects/Django/4.1/django.views.generic.edit/DeleteView/>