**Adding 'messages' in delete-generic-class-based-views**

from django.contrib import messages

class PostDeleteView(SuccessMessageMixin, DeleteView):

model = BlogPost

template\_name = 'blogs/blog-delete.html'

success\_message = 'Post deleted successfully!'

success\_url = "/"

def get\_object(self):

slug = self.kwargs.get('title\_slug')

return get\_object\_or\_404(BlogPost, slug=slug)

def delete(self, request, \*args, \*\*kwargs):

messages.success(self.request, self.success\_message)

return super(PostDeleteView, self).delete(request, \*args, \*\*kwargs)

# Notice, we didn’t add ‘delete’ function in CreateView and UpdateView

class PostCreateView(SuccessMessageMixin, CreateView):

model = BlogPost

template\_name = 'blogs/blog-form.html'

fields = ['title', 'content', 'image']

# %(fieldname)s -> in your model

success\_message = '%(title)s created successfully!'

class PostUpdateView(SuccessMessageMixin, UpdateView):

model = BlogPost

template\_name = 'blogs/blog-update.html'

fields = ['title', 'content', 'image']

success\_message = 'Post updated successfully!'

def get\_object(self):

slug = self.kwargs.get('title\_slug')

return get\_object\_or\_404(BlogPost, slug=slug)