

## How to Show image from database in html ( vid -> Blog app -> 4 )

First you need to install —> pip install Pillow

[index.html](#)

```
{% for post in posts.all %}
    
{% endfor %}
```

[urls.py](#)

```
# for image
from django.conf.urls.static import static
from django.conf import settings

urlpatterns = [

]+ static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

[setting.py](#)

```
STATIC_URL = '/static/'
MEDIA_URL = '/media/'
MEDIA_ROOT = BASE_DIR
```

[modals.py](#)

```
class Post(models.Model):
    img = models.ImageField(upload_to='media/')
```

## Example of upload with User (social media)

### 1. Upload Image with User (socialmedia)

[Modals.py](#)

```
class Profile(models.Model):
    user=models.OneToOneField(User,on_delete=models.CASCADE)
    dob=models.DateField(null=True)
    photo = models.ImageField(null=True,blank=True,upload_to='media/')

    def __str__(self):
        return "Profile of user {}".format(self.user.username)
```

[Forms.py](#)

```
class ProfileEditForm(forms.ModelForm):
    class Meta:
        model=Profile
        fields=(
            'dob',
            'photo',
        )
```

[Views.py](#)

```
def edit_profile(request):

    if request.method == "POST":
        Profile_Edit_form = ProfileEditForm(data=request.POST or None,
instance=request.user.profile, files=request.FILES)

        if Profile_Edit_form.is_valid():
            Profile_Edit_form.save()
            return redirect('edit_profile')
```

**2. Show image and If image is not set show default image with User (socialmedia)**

```
{% if request.user.profile.photo%}

        

{% else %}

        

{% endif %}
```

**Upload Multiple Images (social media)**

[https://www.youtube.com/watch?v=jjdeOp\\_E7OU&list=PLKILtxhEt4-RT-GkrDkJDLuRPQfSK-6Yi&index=39](https://www.youtube.com/watch?v=jjdeOp_E7OU&list=PLKILtxhEt4-RT-GkrDkJDLuRPQfSK-6Yi&index=39)

[models.py](#)

```
class Images(models.Model):

    post = models.ForeignKey(Post,on_delete=models.CASCADE)
    image = models.ImageField(upload_to='images/', blank=True, null=True)

    def __str__(self):
        return str(self.post.id)
```

[Views.py](#)

```
from django.forms import modelformset_factory

def post_create(request):

    ImageFormset = modelformset_factory(Images, fields=('image',), extra=4)

    if request.method == 'POST':

        formm=PostCreateForm(request.POST)
        formset = ImageFormset(request.POST or None, request.FILES or None)
```

```

if formm.is_valid() and formset.is_valid():
    post= formm.save(commit=False)
    post.author = request.user
    post.save()
    formm=PostCreateForm() #reseting form

    for f in formset:
        print(f.cleaned_data)
        try:

            photo = Images(post=post, image=f.cleaned_data.get('image'))

            if f.cleaned_data.get('image') is not None:
                photo.save()

        except Exception as e:
            break
    return redirect('post_list')

else:
    print(formm.errors)

else:
    formm = PostCreateForm()
    formset = ImageFormset(queryset=Images.objects.none())

context={

    'form':formm,
    'formset': formset,

}

return render(request,'blog/post_create.html',context)

```

#### post\_create.html

```

{% extends 'blog/base.html' %}
{% block content %}
<form method="post" enctype="multipart/form-data">
    {% csrf_token %}

    {{ form.as_p }}

    {{ formset.management_form }}

    {% for f in formset %}
        {{ f.as_p }}
    {% endfor %}

    <input type="submit" value="Create" class="btn btn-outline-success">

</form>
{% endblock %}

```

#### To Show images, post\_details.html

```

<div class="row gallery">
    {% for p in post.images_set.all %}
        {% if p.image %}

```

```

<div class="col-md-3">
  
</div>
{% endif %}
{% endfor %}
</div>

```

## Edit Multiple Images

[https://www.youtube.com/watch?v=Fuk\\_OVwovXk&list=PLKILtxhEt4-RT-GkrDkJDLuRPQfSK-6Yi&index=41](https://www.youtube.com/watch?v=Fuk_OVwovXk&list=PLKILtxhEt4-RT-GkrDkJDLuRPQfSK-6Yi&index=41)

[Postdetails.html](#)

```

<a href="{% url 'post_edit' id=post.id %}">
    <button>Edit</button>
</a>

```

[Urls.py](#)

```
url(r'^(?P<id>\d+)/post_edit/$', views.post_edit, name='post_edit'),
```

[views.py](#)

```

def post_edit(request, id):
    post = get_object_or_404(Post, id=id)
    ImageFormset = modelformset_factory(Images, fields=('image',), extra=4, max_num=4)
    if post.author != request.user:
        raise Http404()
    if request.method == "POST":

        form = PostEditForm(request.POST or None, instance=post)

        formset = ImageFormset(request.POST or None, request.FILES or None)

        if form.is_valid() and formset.is_valid():

            form.save()

            print(formset.cleaned_data)

            data = Images.objects.filter(post=post)

            for index, f in enumerate(formset):

                if f.cleaned_data:

                    if f.cleaned_data['id'] is None:

                        photo = Images(post=post, image=f.cleaned_data.get('image'))

                        photo.save()

                    elif f.cleaned_data['image'] is False:

                        photo = Images.objects.get(id=request.POST.get('form-' + str(index) + '-id'))

                        photo.delete()

```

```

else:
    photo = Images(post=post, image=f.cleaned_data.get('image'))
    d = Images.objects.get(id=data[index].id)
    d.image = photo.image
    d.save()

    # messages.success(request, "{} has been successfully updated!".format(post.title))

    return HttpResponseRedirect(post.get_absolute_url())
else:
    form = PostEditForm(instance=post)
    formset = ImageFormset(queryset=Images.objects.filter(post=post))
    context = {
        'form': form,
        'post': post,
        'formset': formset,
    }
    return render(request, 'blog/post_edit.html', context)

```

### Postedit.html

```

{% extends 'blog/base.html' %}

{% block content %}

<form method="post" enctype="multipart/form-data">
    {% csrf_token %}
    {{ form.as_p }}

    {{ formset.management_form }}
    {% for f in formset %}
        {{ f.as_p }}
    {% endfor %}

    <input type="submit" value="Update" class="btn btn-success">
</form>

{% endblock %}

```