

テーブルセットの選択を表示するClass Based ListViewを実現しようとしています。サイトが初めてリクエストされた場合、データセットが表示されます。 POSTの提出を希望しますが、GETも問題ありません。

これはfunction based viewsで簡単に処理できる問題ですが、クラスベースのビューでは頭を悩ませるのに苦労します。

私の問題は、さまざまな数のエラーが発生することです。これは、クラス化されたベースのビューについての理解が不十分なために発生します。さまざまなドキュメントを読み、直接クエリリクエストのビューを理解しましたが、クエリステートメントにフォームを追加しようとするたびに、別のエラーが発生します。以下のコードの場合、ValueError: Cannot use None as a query valueを受け取ります。

フォームエントリに応じてクラスベースのListViewのベストプラクティスワークフローは何ですか（そうでなければデータベース全体を選択します）？

これは私のサンプルコードです：

models.py

```
class Profile(models.Model):
    name = models.CharField(_('Name'), max_length=255)

    def __unicode__(self):
        return '%name' % {'name': self.name}

    @staticmethod
    def get_queryset(params):

        date_created = params.get('date_created')
        keyword = params.get('keyword')
        qset = Q(pk__gt = 0)
        if keyword:
            qset &= Q(title__icontains = keyword)
        if date_created:
            qset &= Q(date_created__gte = date_created)
        return qset
```

forms.py

```
class ProfileSearchForm(forms.Form):
    name = forms.CharField(required=False)
```

views.py

```
class ProfileList(ListView):
    model = Profile
```

```

form_class = ProfileSearchForm
context_object_name = 'profiles'
template_name = 'pages/profile/list_profiles.html'
profiles = []

def post(self, request, *args, **kwargs):
    self.show_results = False
    self.object_list = self.get_queryset()
    form = form_class(self.request.POST or None)
    if form.is_valid():
        self.show_results = True
        self.profiles = Profile.objects.filter(name__icontains=form.cleaned_data['name'])
    else:
        self.profiles = Profile.objects.all()
    return self.render_to_response(self.get_context_data(object_list=self.object_list, form=form))

def get_context_data(self, **kwargs):
    context = super(ProfileList, self).get_context_data(**kwargs)
    if not self.profiles:
        self.profiles = Profile.objects.all()
    context.update({
        'profiles': self.profiles
    })
    return context

```

以下に私は仕事をするFBVを追加しました。 **この機能をCBVに変換するにはどうすればよいですか？** 関数ベースのビューでは非常に単純ですが、クラスベースのビューではそうではありません。

```

def list_profiles(request):
    form_class = ProfileSearchForm
    model = Profile
    template_name = 'pages/profile/list_profiles.html'
    paginate_by = 10

    form = form_class(request.POST or None)
    if form.is_valid():
        profile_list = model.objects.filter(name__icontains=form.cleaned_data['name'])
    else:
        profile_list = model.objects.all()

    paginator = Paginator(profile_list, 10) # Show 10 contacts per page
    page = request.GET.get('page')
    try:
        profiles = paginator.page(page)
    except PageNotAnInteger:
        profiles = paginator.page(1)
    except EmptyPage:
        profiles = paginator.page(paginator.num_pages)

    return render_to_response(template_name,

```

```
{'form': form, 'profiles': suppliers,},  
context_instance=RequestContext(request))
```