1. **Django (3.0) Crash Course Tutorials (pt 11)**

inlineformset\_factory is allowed multiple record to be added into Database.

Step below required:

* Import the inlineformset\_factory to the views:

**from django.forms import inlineformset\_factory**

* Assign variable in the views, first index is parent model second is child model.

**OrderFormSet = inlineformset\_factory(Customer,Order,fields=('customer','product','status'),extra=2)**

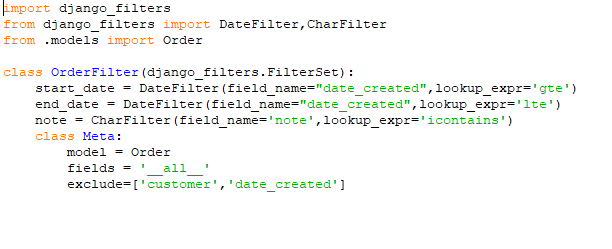
* Assign variable to reference instance for the selected record. "queryset” to ignore any record existent in the database.

**formset = OrderFormSet(queryset=Order.objects.none(),instance=customer)**

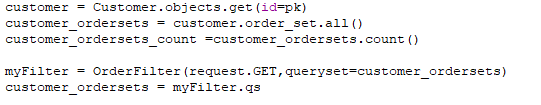
1. **Django (3.0) Crash Course Tutorials (pt 12)**
   1. pip install django-filter
   2. include 'django\_filter', in the setting.py under main project.



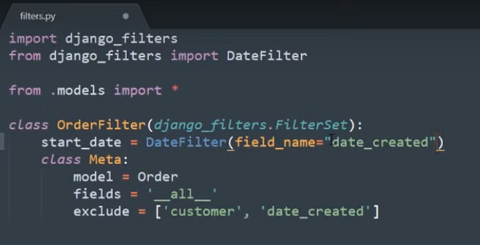
* 1. Create filters.py under app folder.

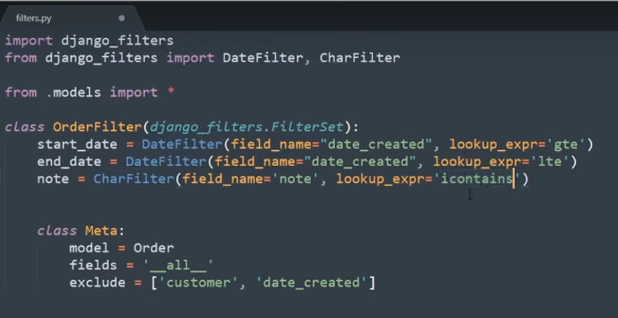


* + 1. customer\_ordersets = customer.order\_set.all()
    2. myFilter = OrderFilter(request.GET,queryset=customer\_ordersets)
    3. customer\_ordersets = myFilter.qs
    4. add customer\_ordersets in context.



* 1. Customise date field for search function
     1. Under filters.py
        1. from django\_filters import DateFilter,CharFilter
        2. exclude=['customer','date\_created']
        3. start\_date = DateFilter(field\_name="date\_created",lookup\_expr='gte')
        4. end\_date = DateFilter(field\_name="date\_created",lookup\_expr='lte')
        5. note = CharFilter(field\_name='note',lookup\_expr='icontains')





1. **Django (3.0) Crash Course Tutorials (pt 13)**
   1. **Register**
   2. **Import to views**
      1. from django.contrib.auth.forms import UserCreationForm
      2. from django.contrib import messages
      3. create def:-
      4. def registerPage(request):
         1. form = CreateUserForm()
         2. if request.method == "POST":
         3. form= CreateUserForm(request.POST)
         4. if form.is\_valid():
         5. form.save()
         6. user = form.cleaned\_date.get('username')
         7. messages.seccess(request,'Account was created for ' + user)
         8. return redirect('accounts:login\_page')
         10. context = {'form':form}
         11. return render(request,'accounts/register.html',context)add messages script on login.html to
      5. show message:-
         1. {% for message in messages %}
         2. <p id="messages">{{message}}</p>
         3. {% endfor %}
      6. Create login def:-
         1. def loginPage(request):
         2. if request.method == "POST":
         3. username = request.POST.get('username')
         4. password = request.POST.get('password')
         5. user = authenticate(request,username=username,password=password)
         6. if user != None:
         7. login(request,user)
         8. return redirect('accounts:dashboard')
         9. else:
         10. messages.info(request,"username or Password is incorrect")
         12. context = {}
         13. return render(request,'accounts/login.html',context)
   3. create forms.py
      1. from django.contrib.auth.forms import UserCreationForm
      2. from django.contrib.auth.models import User
      3. from django import forms
      4. create customise fields for the register form: -

class CreateUserForm(UserCreationForm):

class Meta:

model = User

fields = ['username','email','password1','password2']