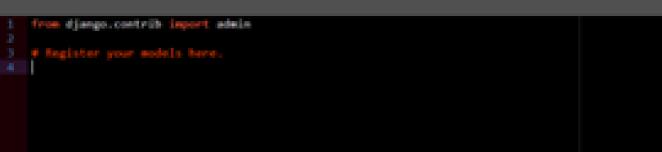


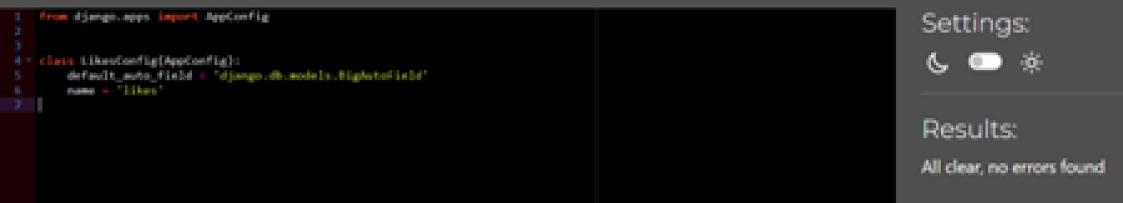
te CI Python Linter





All clear, no errors found





```
from django de import models
                                                                                                                            Settings:
row diango.contrib.auth.models import User
from posts.models import Post
class tike(wodels.Model):
   Like model, related to 'owner' and 'post'.
                                                                                                                            Results:
   'owner' is a User instance and 'post' is a Post instance.
   "unique_together" makes oure a user can't like the same post twice.
                                                                                                                            All clear, no errors found
   owner - models.foreignKey(User, on_delete-models.CASCASE)
   post = models.ForeignKey(
       Post, related names 'likes', on delete-models CASCACE
   created at = models.Outslimsfield(auto now add-True)
   Charles Budger
       ordering = ['-created.st']
       unique together = ['owner', 'post']
   def _str_(salf):
      return f'(self.owner) (self.post)'
```

```
ros diango, do import IntegrityError
                                                                                                                              Settings:
rom rest framework import serializers
ros likes models isport like
rlass Liketorializer(sorializers.Modeltorializer):
   terializer for the Like model
   The create method handles the unique constraint on 'owner' and 'post'
   owner = serializers.ReadDelyfield(source:'owner.usernome')
   alleren Median
       model - Like
       fields : ['id', 'created_st', 'oseer', 'post']
   def create(self, validated_data):
       Sept.
           return super().create(validated_data)
       succest Integrityferor:
           raise serializers. Validatiosfrom()
               'detail': 'possible duplicate'
```



Results:

All clear, no errors found



```
Settings:
 ros cookbook and paradectors import TobanarOrthaadOnly
 ron likes models import tike
 ros likes carializars import tilutarializar
 lass LikeList(generics.ListCreateN*IVies):
   List likes or create a like if loaged in.
   permission_classes = [permissions.lsbstheeticatedOrReadOrly]
   serializer class - Libelerializer
   queryart : Like.objects.sil()
   def perform create(nelf, nerializer):
       serializer.save(owner-self.request.user)
class LikeDetail(generics.RetrieveDestroyAFDView):
   Astrieve a like or delete it by id if you own it.
   permission_classes = [Is0-ner0rhead0nly]
   serializer class = Libeterializer
   queryset = Like.objects.sll()
```

ros rest framework import generics, permissions



Results:

All clear, no errors found