

## CI Python Linter

```
from django.apps import AppConfig
4 = class ProfilesConfig(AppConfig):
       default moto field : "djamgo.db.models.bighutofield"
       mane + "profiles"
```

Settings:





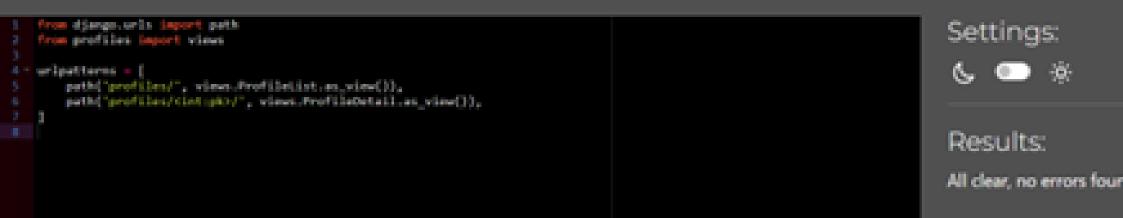


Results:

All clear, no errors found

```
from disease, the import models.
                                                                                                                             Settings:
from django.db.models.signals import post_save
from diango.comtrib.auth.models import ther
class Profile(models.Model):
    The model for the profiles.
                                                                                                                             Results:
    owner - models.OneToOneField(User, on_delste-models.CASCADE)
    created at a models.Oatelimefield(outs now add:True)
                                                                                                                             All clear, no errors found
    updated at a models.DataTimeField(auto.now-True)
    nome = models Charffeld(max_length=255, blank=True)
    contant - models.Tertfield(black-from)
    image a models. Imagefield[
        upload to-'images/', default-'../profile_outline_erj/ou'
    chess Retail
        ordering - ["-created at"]
    def _str_(self):
        return f"(self.owner)'s profile"
    create_profile(sender, instance, created, "%wargs):
    of createds
        Profile objects create(owner-instance)
post_save.connect(create_profile, sender:User)
```

```
Settings:
 ros followers, models import follower
class ProfileSerializer(serializers, ModelSerializer);
    tarializer for the Profile model.
                                                                                                                               Results:
    owner = serializers.ReadOnlyfield(source='owner.username')
    is owner = serializers.SerializerMethodfield[]
    following id = serializers.terializer#sthodField()
                                                                                                                               All clear, no errors found
   posts_count = serializers.ReadOnly(ield()
   followers count - serializers. ReadOnlyfield()
    following count a serializers.ReadOnlyfield()
   def get_is_owner(self, obj):
        request = self.context['request']
        return request user III obij owner
   def get fellowing id(self, obi):
        user - self.context['request'].user
       if uper in authenticated:
           following = Follower.objects.filter(
                owner weer, followed robit owner
            D. Filest ()
           return following id if following else None
        nations Some
   others Metal
       model - Frafile
        filelies - I
            "id", "owner", "created_at", "splated_at", "name",
            'content', 'image', 'is owner', 'following id',
            'posts count', 'followers count', 'following count',
```



```
queryset = Profile.objects.annotate(
        posts_count-Count('ouner_post', distinct-True),
        fellowers_count:Count('owner_fellowed', distinct:True),
        following count-tount('owner_following', distinct-frue)
    3. order by("created at")
    perializer class - Profileterializer
    filliar backends = 1
        filters.Orderingfilter,
       Djangofilterbackend.
    filterset_fields = [
        'owner following followed profile',
        'owner_followed_owner_profile',
   ordering fields = I
        "positive counts".
        "followers_count".
        "fullowing count".
        'owner_following_created st'.
        'owner followed created at'
class ProfileDetail(generics_MatriavethdateMFTVisu):
    Astrieve or update a profile if you're the owner.
    permission classes - [fobmerOrReadOnly]
   queryset | Profile.objects.annotate(
        posts count-Count('owner_post', distinct-True),
        followers_count=fount('owner_followed', distinct=True),
        following count-fount('owner_following', distinct-True)
    ) order by('-created at')
    serializer class - Profileterializer
```



Settings:



Results:



















