

```
1 from django.apps import AppConfig
2
3
4 class FollowersConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigIntField'
6     name = 'followers'
7
```

Settings:



Results:

All clear, no errors found

```
1 from django.db import models
2 from django.contrib.auth.models import User
3
4
5 class Follower(models.Model):
6     """
7     Follower model, related to 'owner' and 'followed'.
8     'owner' is a User that is following a User.
9     'followed' is a User that is followed by 'owner'.
10    'unique_together' makes sure a user can't 'double follow' the same user.
11    """
12    owner = models.ForeignKey(
13        User, related_name="following", on_delete=models.CASCADE
14    )
15    followed = models.ForeignKey(
16        User, related_name="followed", on_delete=models.CASCADE
17    )
18    created_at = models.DateTimeField(auto_now_add=True)
19
20 class Meta:
21     ordering = ['-created_at']
22     unique_together = ['owner', 'followed']
23
24 def __str__(self):
25     return f'{self.owner} {self.followed}'
26
```

Settings:



Results:

All clear, no errors found

```

1 from django.db import IntegrityError
2 from rest_framework import serializers
3 from models import Follower
4
5
6 class FollowerSerializer(serializers.ModelSerializer):
7     """
8     Serializer for the Follower model.
9     Create method handles the unique constraint on 'owner' and 'followed'.
10    """
11    owner = serializers.ReadOnlyField(source='owner.username')
12    followed_name = serializers.ReadOnlyField(source='followed.username')
13
14    class Meta:
15        model = Follower
16        fields = [
17            'id', 'owner', 'created_at', 'followed', 'followed_name'
18        ]
19
20    def create(self, validated_data):
21        try:
22            return super().create(validated_data)
23        except IntegrityError:
24            raise serializers.ValidationError({'detail': 'possible duplicate'})
25

```

Settings:



Results:

All clear, no errors found

```

1 from django.conf import settings
2 from django.conf.urls import url
3
4 urlpatterns = [
5     url(r'^$', views.home, name='home'),
6     url(r'^about/$', views.about, name='about'),
7 ]

```

Settings



Results

All clear, no errors found

```

1 from rest_framework import generics, permissions
2 from cookbook_api.permissions import IsOwnerOrReadOnly
3 from .models import Follower
4 from .serializers import FollowerSerializer
5
6
7 - class FollowerList(generics.ListCreateAPIView):
8     """
9     List all followers, i.e. all instances of a user
10    following another user".
11    Create a follower, i.e. follow a user if logged in.
12    Perform_create: associate the current logged in user with a follower.
13    """
14    permission_classes = [permissions.IsAuthenticatedOrReadOnly]
15    queryset = Follower.objects.all()
16    serializer_class = FollowerSerializer
17
18 -     def perform_create(self, serializer):
19         serializer.save(owner=self.request.user)
20
21
22 - class FollowerDetail(generics.RetrieveDestroyAPIView):
23     """
24     Retrieve a follower
25     No Update view, as we either follow or unfollow users
26     Destroy a follower, i.e. unfollow someone if owner
27     """
28    permission_classes = [IsOwnerOrReadOnly]
29    queryset = Follower.objects.all()
30    serializer_class = FollowerSerializer
31

```

Settings:



Results:

All clear, no errors found