

```
1 from django.apps import AppConfig
2
3
4 class ShoppingListConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'shoppinglist'
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 ShoppingList(models.Model):
6     """
7     For adding items and quantities to the shopping list.
8     """
9     owner = models.ForeignKey(User, on_delete=models.CASCADE)
10    name = models.CharField(max_length=200)
11    quantity = models.CharField(max_length=200)
12    created_at = models.DateTimeField(auto_now_add=True)
13    updated_at = models.DateTimeField(auto_now=True)
14
15    def __str__(self):
16        return f'{self.name} - {self.quantity}'
17
18    class Meta:
19        ordering = ['-created_at']
20

```

Settings:



Results:

All clear, no errors found

```

1 from rest_framework import serializers
2 from .models import Shoppinglist
3
4
5 class ShoppinglistSerializer(serializers.ModelSerializer):
6     """
7     Serializer for the shoppinglist model.
8     """
9     class Meta:
10         model = Shoppinglist
11         fields = ('id', 'name', 'quantity', 'created_at', 'updated_at')
12
13
14 class ShoppinglistDetailSerializer(serializers.ModelSerializer):
15     """
16     Serializer for the shoppinglist model used in Detail view.
17     """
18     class Meta:
19         model = Shoppinglist
20         fields = ('id', 'name', 'quantity', 'created_at', 'updated_at')
21

```

Settings:



Results:

All clear, no errors found

```

1 from django.urls import path
2 from views import ShoppingListView, ShoppingListDetailView
3
4
5 urlpatterns = [
6     path('shoppinglist/', ShoppingListView.as_view()),
7     path('shoppinglist/<int:pk>/', ShoppingListDetailView.as_view())
8 ]
9

```

Settings



Results

All clear, no errors found

```

1 from rest_framework import generics
2 from .models import Shoppinglist
3 from .serializers import ShoppinglistSerializer, ShoppinglistDetailSerializer
4 from cookbook_api.permissions import IsOwnerOrReadOnly
5 from django.shortcuts import get_object_or_404
6 from rest_framework.response import Response
7
8
9 class ShoppinglistView(generics.ListCreateAPIView):
10     """
11     View for creating shopping list item if user is logged in.
12     """
13     queryset = Shoppinglist.objects.all()
14     serializer_class = ShoppinglistSerializer
15     permission_classes = [IsOwnerOrReadOnly]
16
17     def get_queryset(self):
18         user = self.request.user
19         return Shoppinglist.objects.filter(owner=user)
20
21     def perform_create(self, serializer):
22         serializer.save(owner=self.request.user)
23
24
25 class ShoppinglistDetailView(generics.RetrieveUpdateDestroyAPIView):
26     """
27     Retrieving, updating, and deleting a shoppinglist item.
28     """
29     queryset = Shoppinglist.objects.all()
30     serializer_class = ShoppinglistDetailSerializer
31     permission_classes = [IsOwnerOrReadOnly]
32
33     def get_queryset(self):
34         user = self.request.user
35         return Shoppinglist.objects.filter(owner=user)
36

```

Settings:



Results:

All clear, no errors found