# PEP8 code validation - products app

## admin.py

**All right**   Save ▾   Share

**Your code**

```python
1   """Set up of django admin site for products app"""
2   from django.contrib import admin
3   from .models import Category, Product
4
5
6   @admin.register(Category)
7 ▾ class CategoryAdmin(admin.ModelAdmin):
8       """
9 ▾     Category model - admin site set up:
10      Show all fields in list display
11      """
12      list_display = ('friendly_name', 'name')
13
14
15  @admin.register(Product)
```

Check again

## forms.py

**All right**   Save ▾   Share

**Your code**

```python
1   """Forms for 'products' app - shop"""
2   from django import forms
3   from .widgets import CustomClearableFileInput
4   from .models import Product, Category
5
6
7 ▾ class ProductForm(forms.ModelForm):
8       """Product form for admin user to add/edit product from frontend"""
9 ▾     class Meta:
10          """
11          Form based on Product model.
12          Helptexts and labels specified for some fields.
13          Image field - use custom file input widget that overrides Django one
14          """
15          model = Product
```

Check again

## models.py

**All right**   Save ▾   Share

**Your code**

```python
1   """models for products app"""
2   from django.db import models
3   from django.db.models.signals import post_save
4   from django.dispatch import receiver
5
6
7 ▾ class Category(models.Model):
8       """Category model for products"""
9
10 ▾    class Meta:
11          """Specify correct spelling for plural - for admin site"""
12          verbose_name_plural = 'Categories'
13
14      name = models.CharField(max_length=50, unique=True)
15      friendly_name = models.CharField(max_length=50)
```

Check again

## test_forms.py

**All right**   Save ▾   Share

**Your code**

```python
1   """Tests for forms in products app - shop"""
2   from django.test import TestCase
3   from .forms import ProductForm
4
5
6 ▾ class TestProductForm(TestCase):
7       """Tests for the ProductForm"""
8 ▾     def test_explicitly_set_field_labels_exist(self):
9           """
10          Check labels are present for the fields where a label was explicitly
11          set in the form Meta class
12          """
13          form = ProductForm()
14          self.assertTrue(
15              form.fields['is_active'].label is None or
```

Check again

# PEP8 code validation - products app

## test_models.py

**PEP8 online**
Check your code for PEP8 requirements

**All right**   Save ▾   Share

**Your code**

```
1   """Unit tests for Models in 'products' app"""
2   from django.test import TestCase
3   from django.db import IntegrityError
4   from .models import Category, Product
5
6
7   class TestCategoryModel(TestCase):
8       """tests for Category model in products app"""
9       @classmethod
10      def setUpTestData(cls):
11          """Create instance of Category model for tests"""
12          Category.objects.create(
13              name='category_and_category',
14              friendly_name='Category & Category'
15          )
```

Check again

## test_views.py

**PEP8 online**
Check your code for PEP8 requirements

**All right**   Save ▾   Share

**Your code**

```
1   """Tests for views in 'product' app (shop)"""
2   from decimal import Decimal
3   from django.test import TestCase
4   from django.contrib.auth.models import User
5   from django.contrib.messages import get_messages
6   from .models import Category, Product
7
8
9   class TestShowProductsView(TestCase):
10      """To test the show_products view - shop page displaying products"""
11      @classmethod
12      def setUp(cls):
13          """
14          Create two instances of Category
15          Create 7 instances of Product, 3 active, 4 inactive
```

Check again

## urls.py

**PEP8 online**
Check your code for PEP8 requirements

**All right**   Save ▾   Share

**Your code**

```
1    """URL paths for the 'products' app (shop)"""
2    from django.urls import path
3    from . import views
4
5
6    urlpatterns = [
7        path('', views.show_products, name='products'),
8        path('<int:product_id>', views.product_details, name='product_details'),
9        path('add/', views.add_product, name='add_product'),
10       path('edit/<int:product_id>/', views.edit_product, name='edit_product'),
11       path(
12           'delete/<int:product_id>/', views.delete_product, name='delete_product'
13       ),
14       path(
15           'toggle/<int:product_id>/',
```

Check again

## views.py

**PEP8 online**
Check your code for PEP8 requirements

**All right**   Save ▾   Share

**Your code**

```
1    """Views for products app - shop pages, product admin"""
2    from django.shortcuts import render, get_object_or_404, redirect, reverse
3    from django.http import Http404
4    from django.db.models import Q
5    from django.db.models.functions import Lower
6    from django.contrib import messages
7    from django.contrib.auth.decorators import login_required
8    from django.views.decorators.http import require_POST
9    from django.core.exceptions import PermissionDenied
10   from .models import Product, Category
11   from .forms import ProductForm
12
13
14   def show_products(request):
15       """
```

Check again