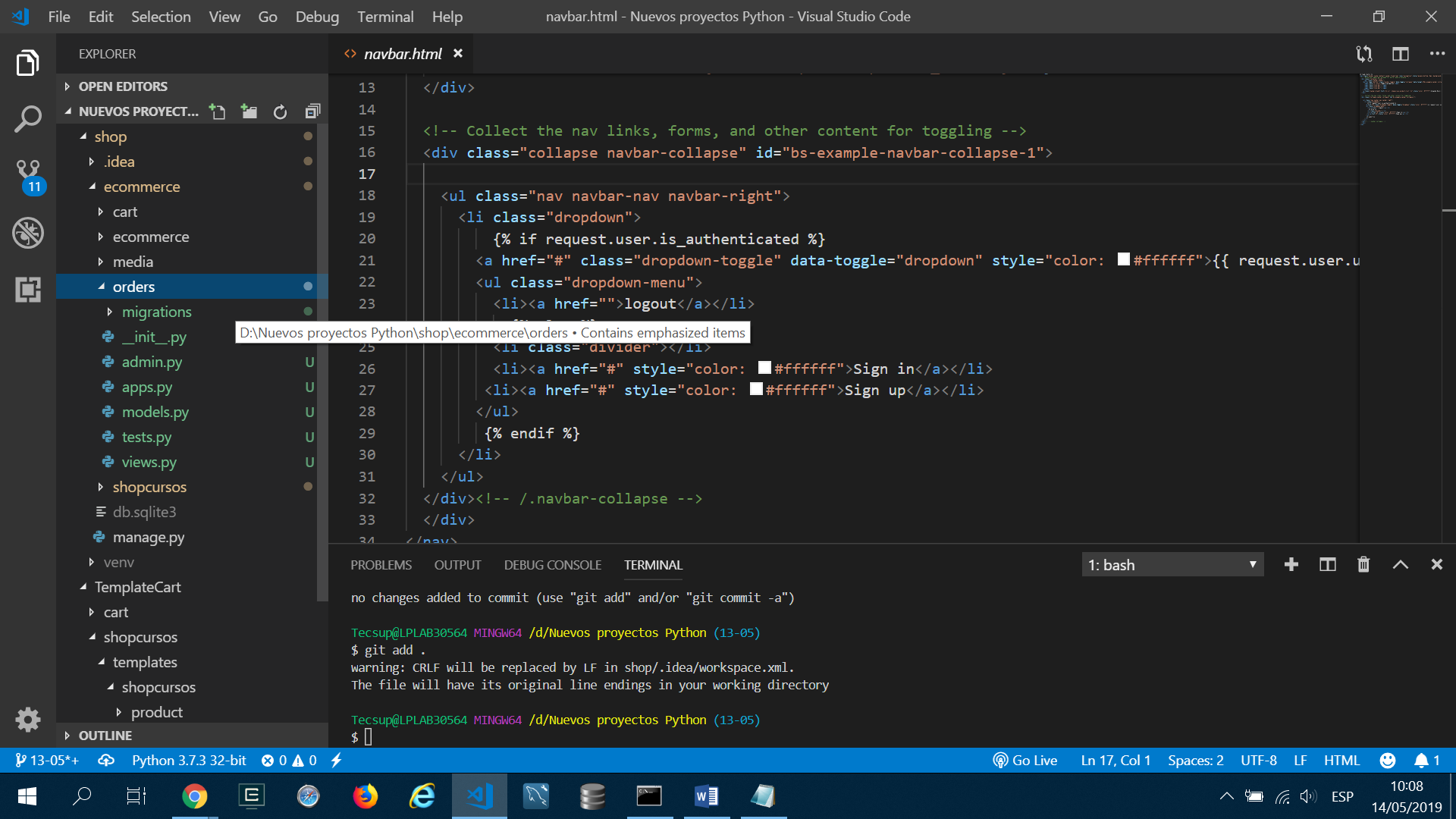
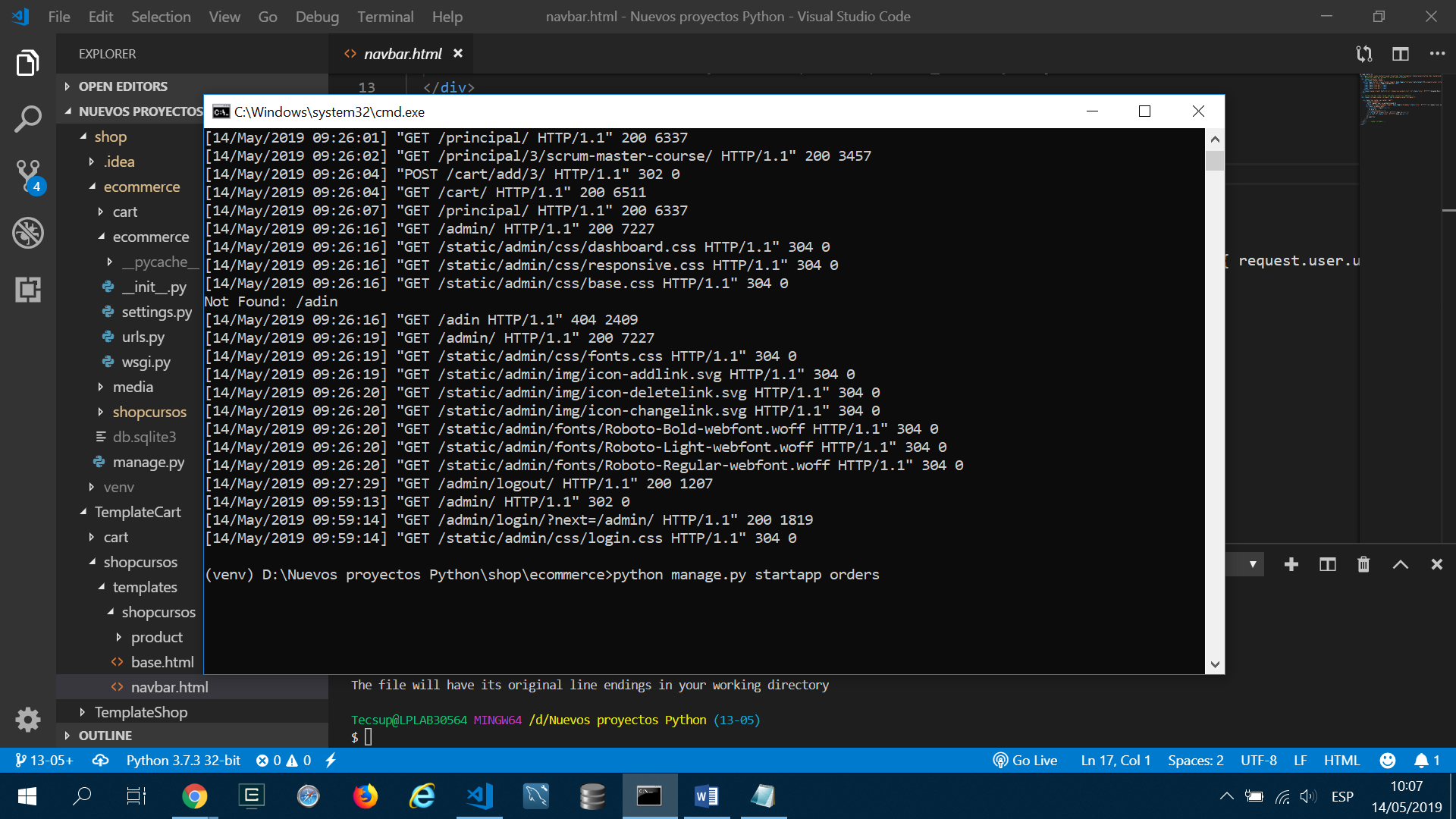
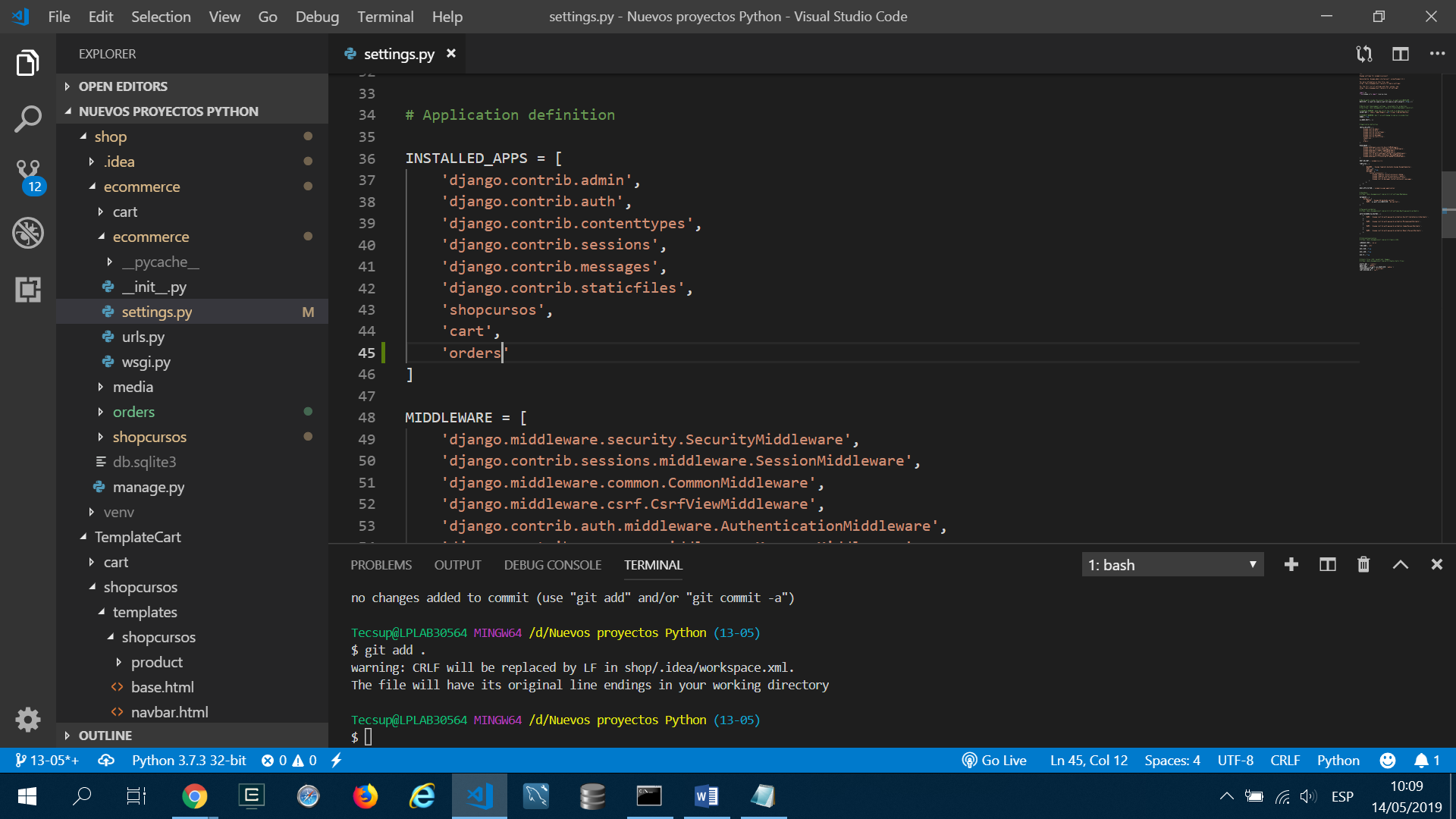
Toda la info está en nuestra session, y cuando estemos seguros lo que irá al final lo agregamos (CHECKOUT) a una nueva tabla llamada ORDERDS.

Creamos la nueva aplicación ORDERS



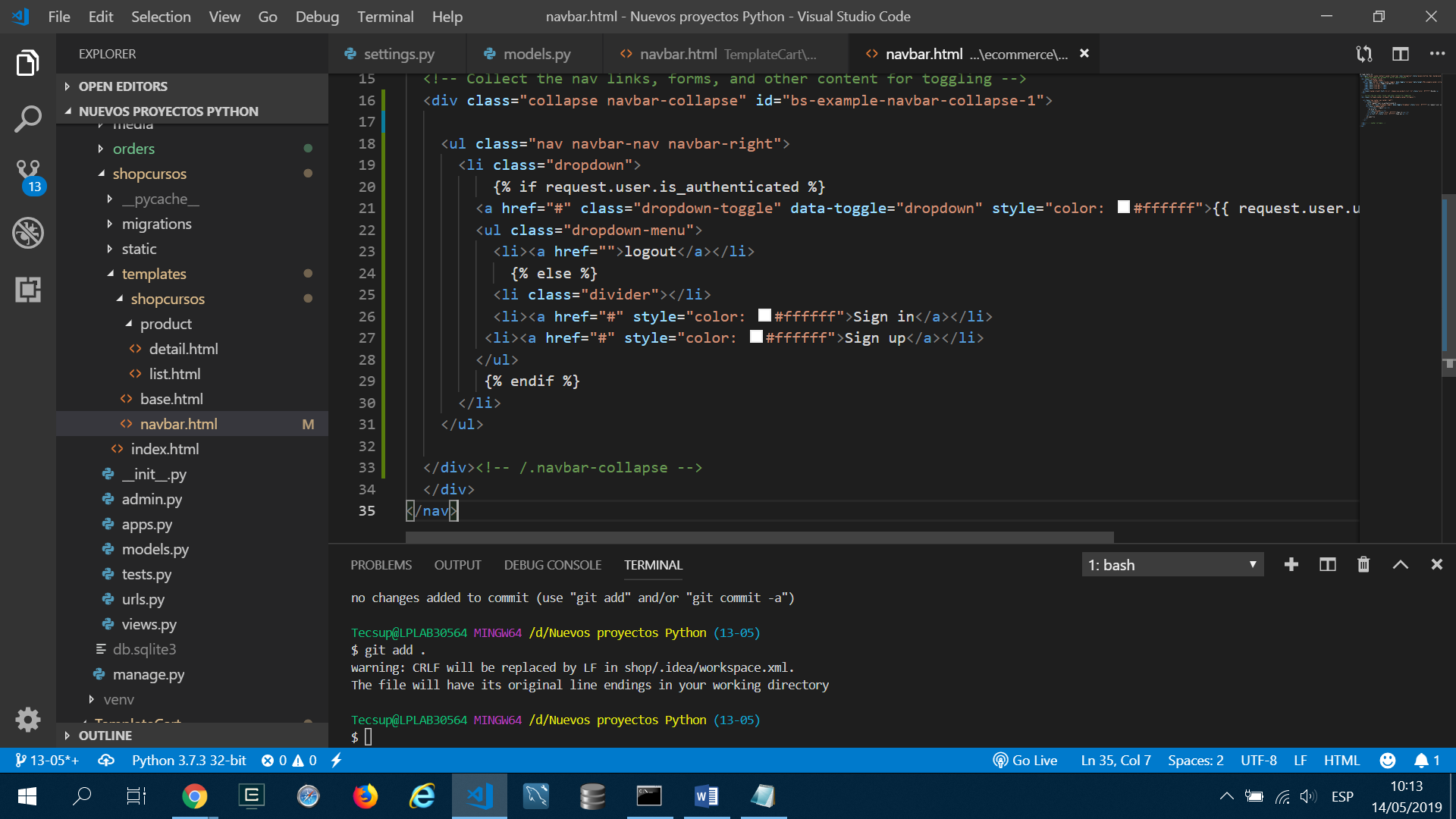
Agregamos la APP a INSTALLED APPS



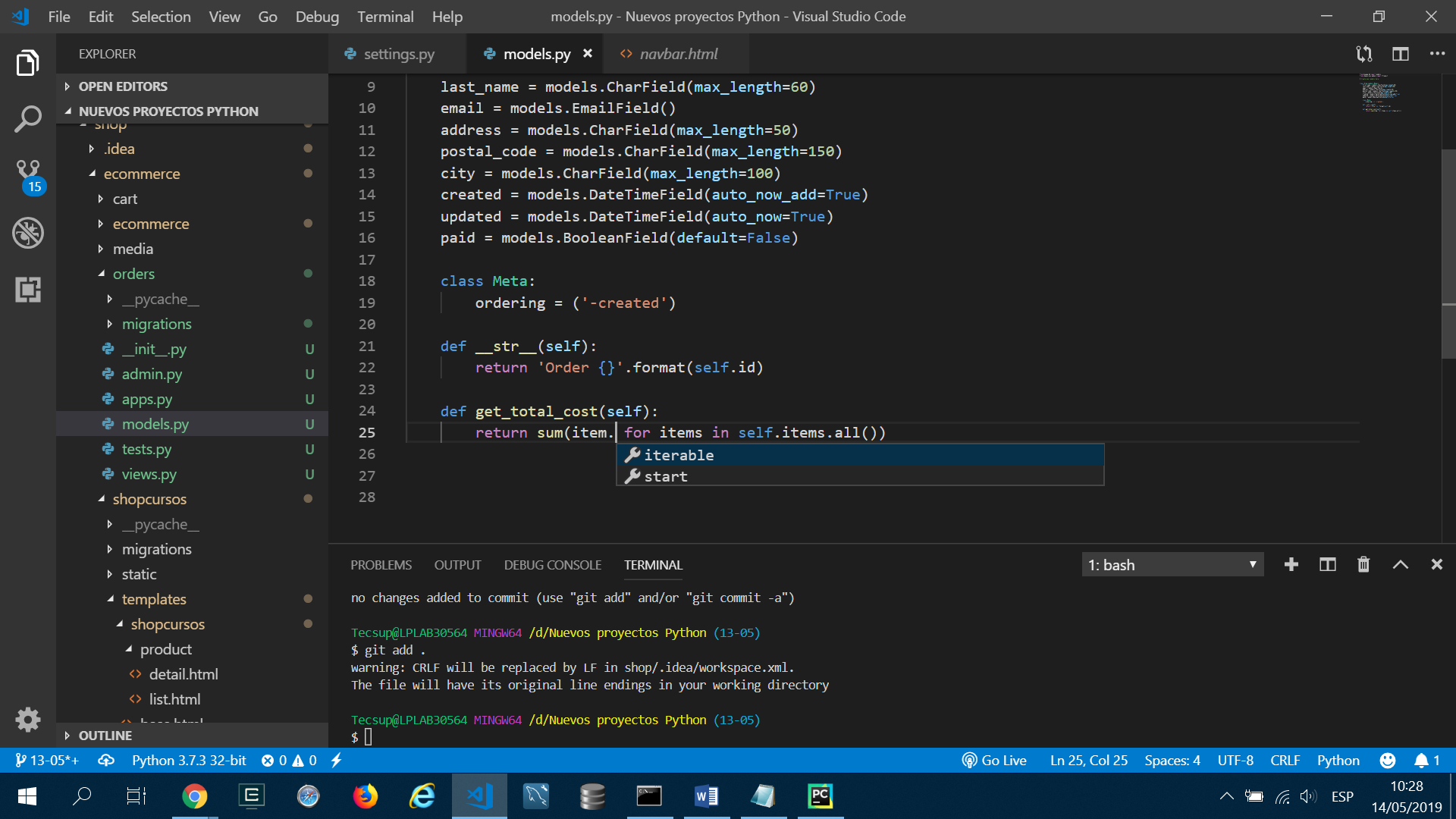
En models de orders:

Creamos un modelo usando la info de la tabla Product, para mostrar la cabecera (con datos de cliente) y el detalle de los productos agregados al carrito.

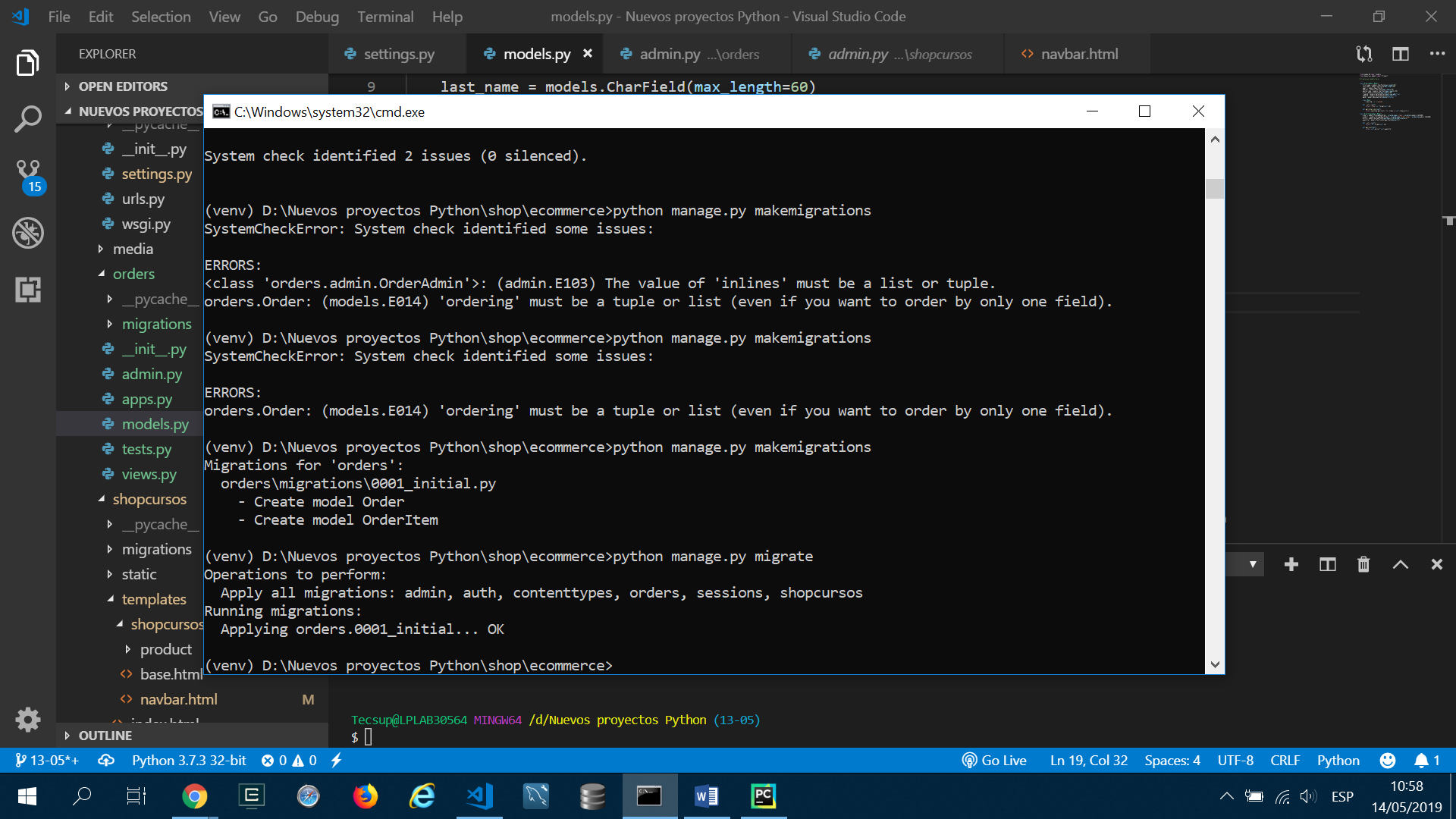
ME FALTO AGREGAR EL HTML AL NAVBAR



Pasando los valores para sumarlos, estos deben pasarse mediante el objeto **self**.



Para mostrar uno después de otro, el modelo OrderItem usará admin.TabularInline.



**Problemas con TUPLAS:**

class Meta:

ordering = ('-created')

class Meta:

ordering = ['-created']

o ordering = ('-created',)

**Otro problema:**

**Wrong:**

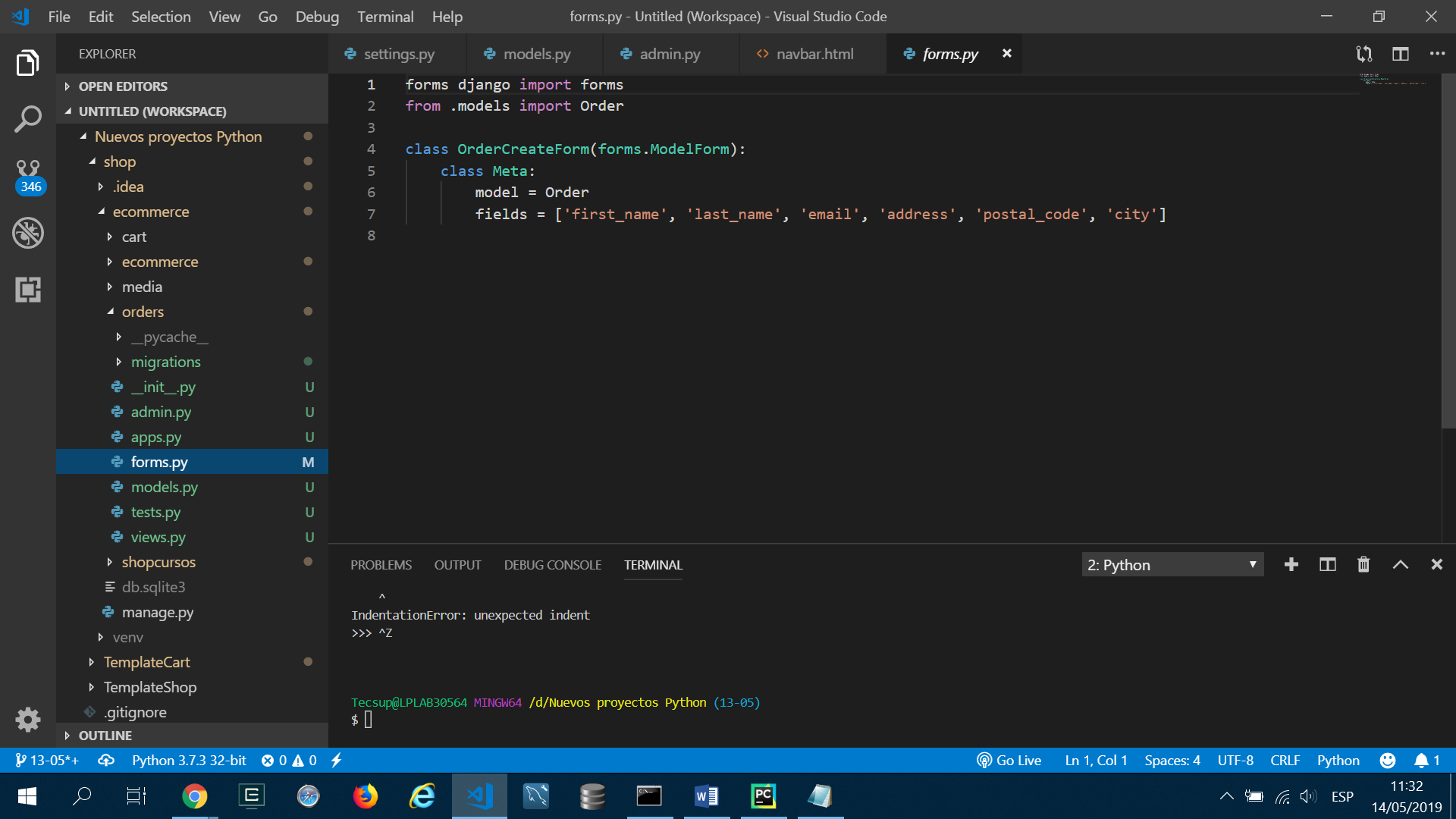
inlines = OrderItemInline

Right:

inlines = [OrderItemInline]

CLICK en Checkout => formulario para usuario => Se pinta la orden con los datos del usuario.

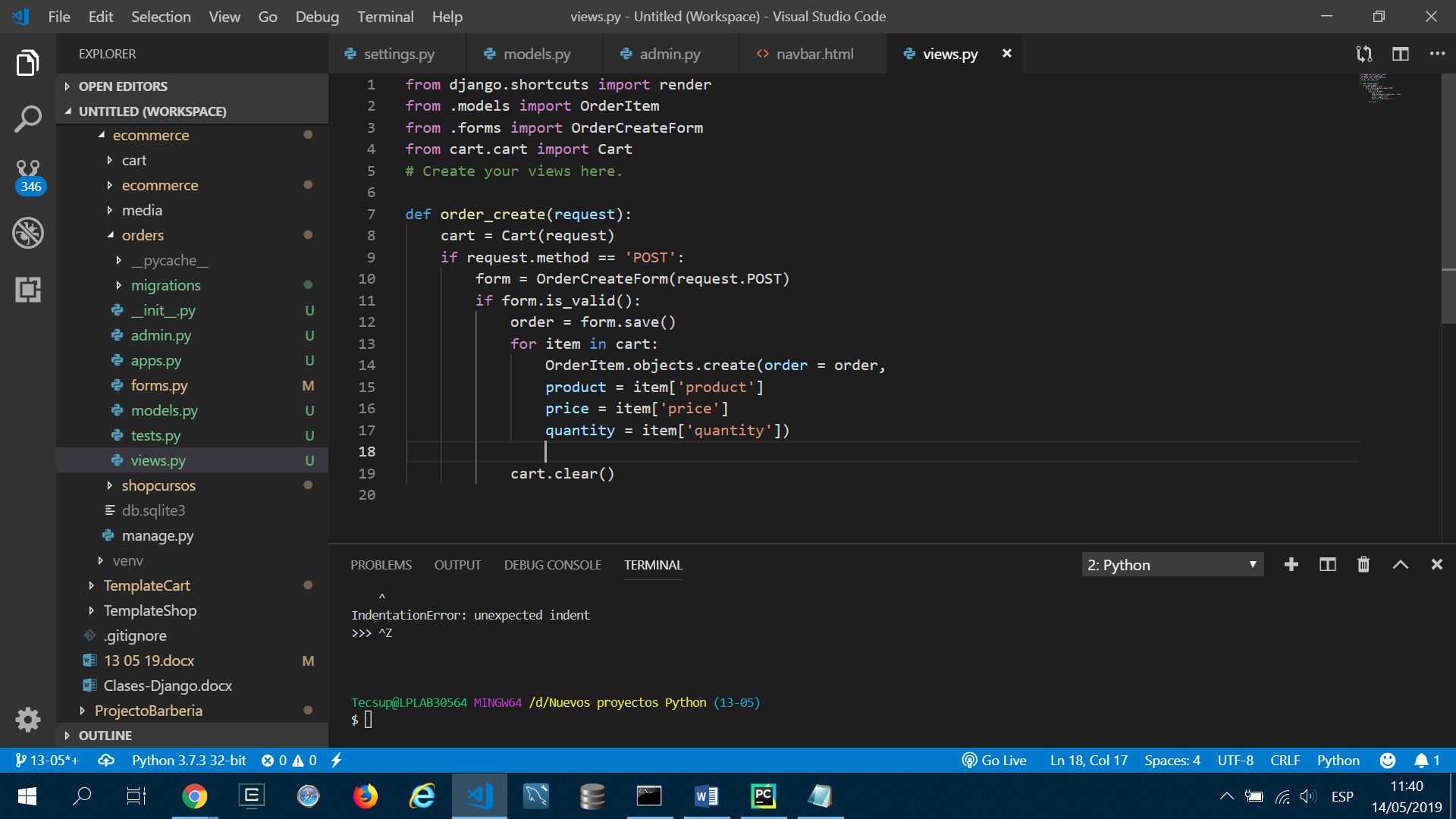
CREAMOS FORMULARIO:



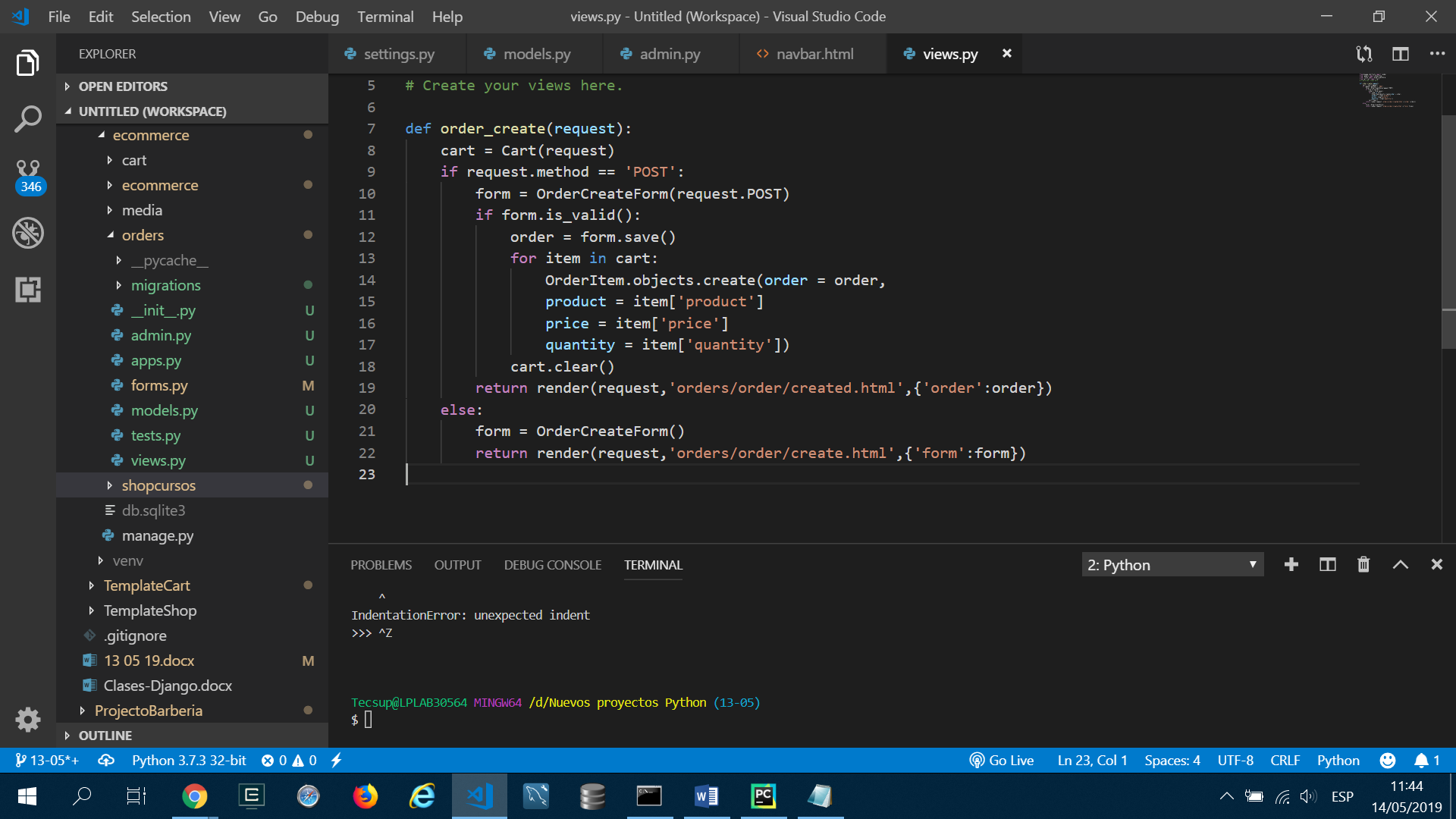
EN VIEWS.PY

CREAMOS EL VIEW PARA ENVIAR LA INFO A TRAVÉS DE CONTEXT

USAMOS MÉTODO CLEAR para borrar los datos de la sesión una vez se envíen los datos del usuario.

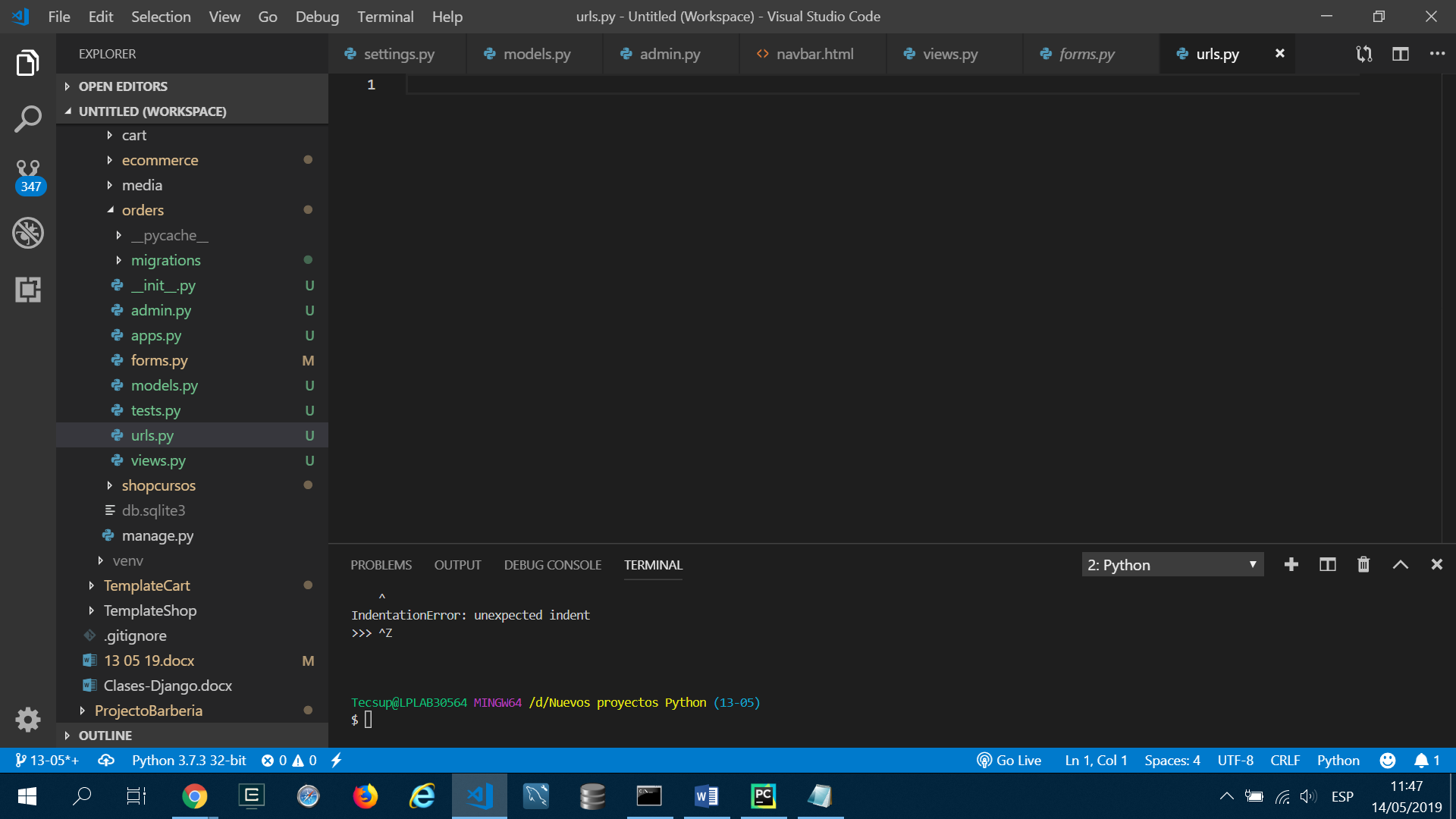


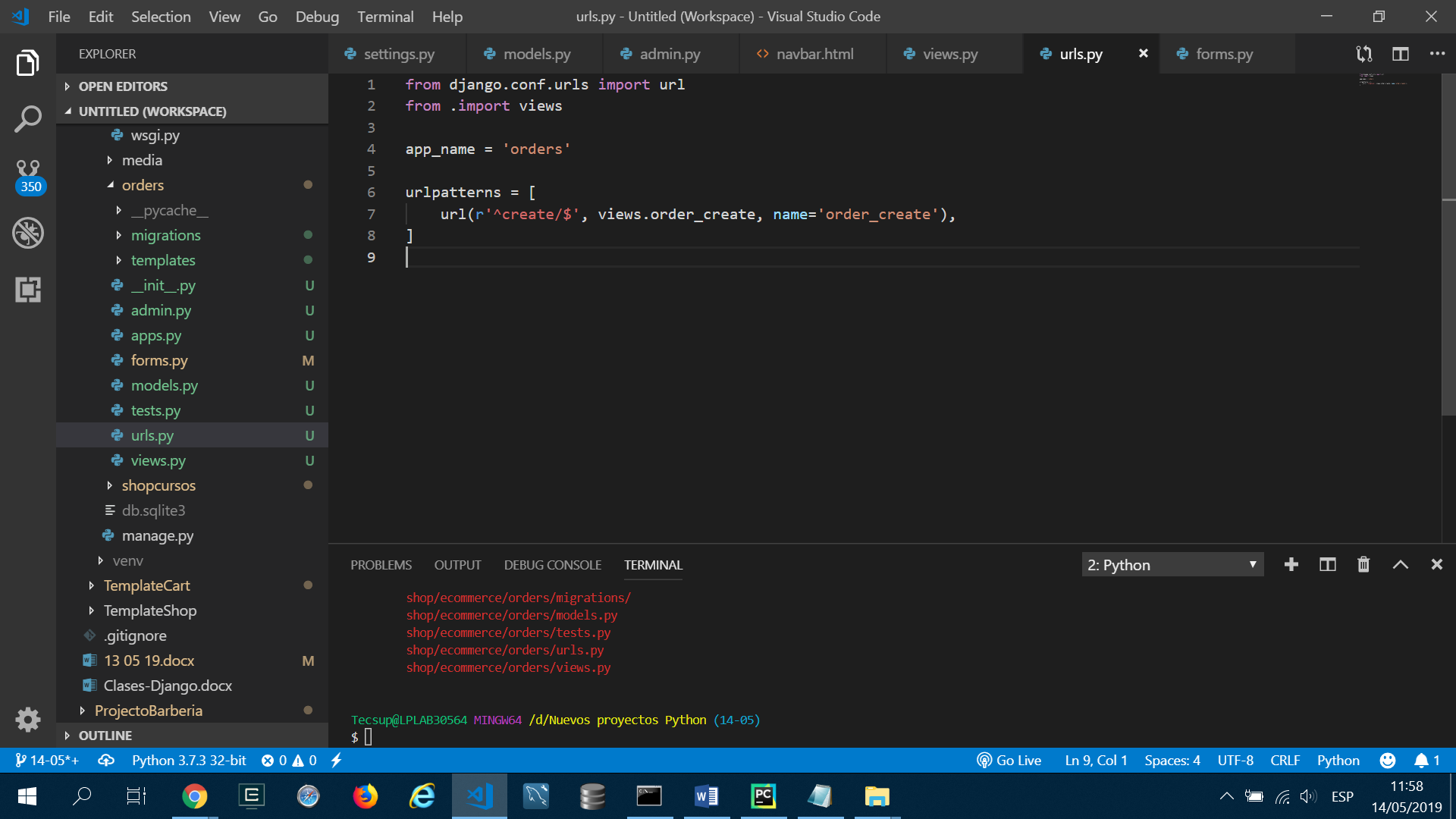
Le estamos pasando la plantilla y la información que está dentro de order:



El form retorna un id, y un formulario nuevo.

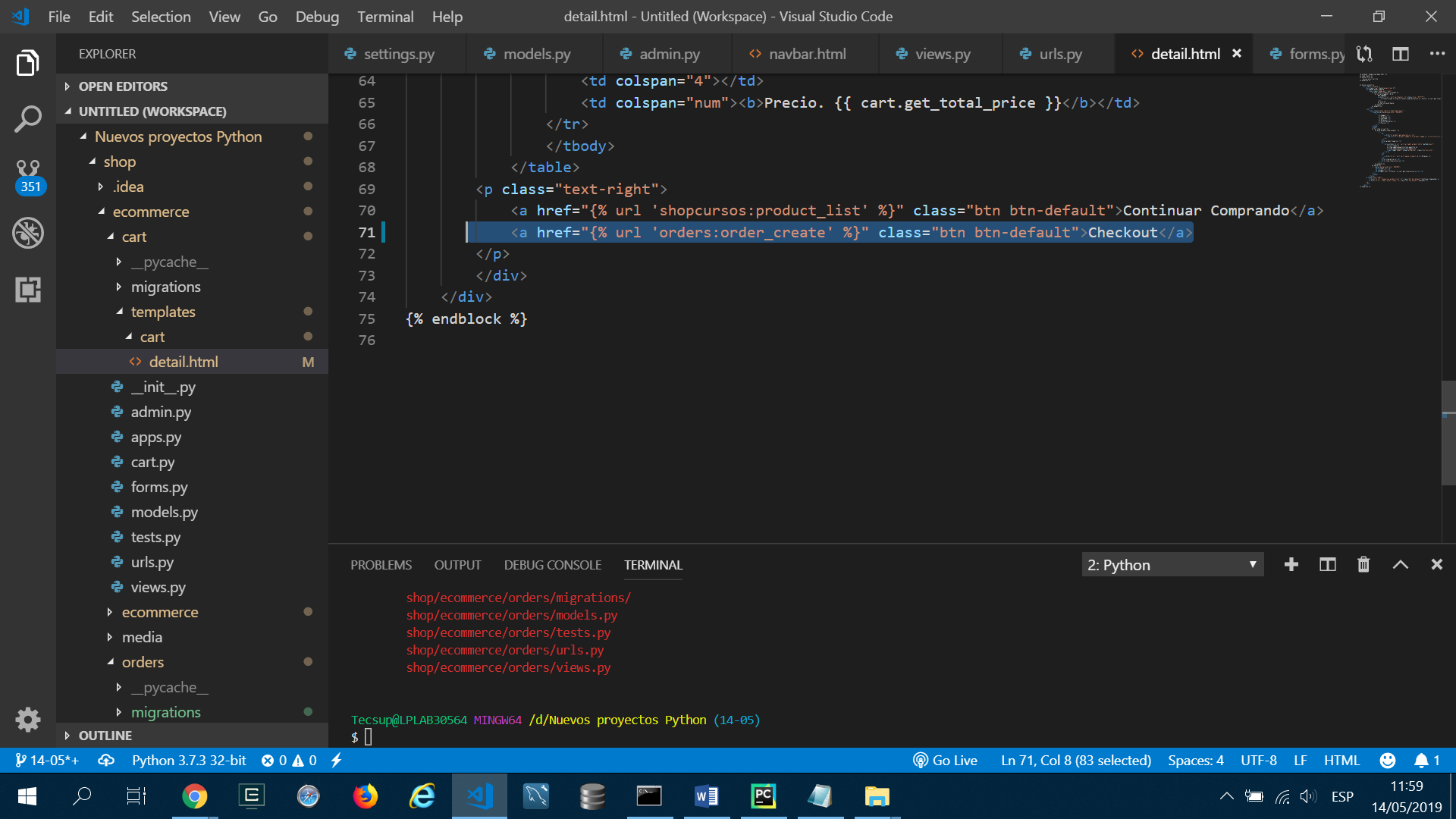
Creamos las URLS





Agregamos botón CHECKOUT en templates de cart

Y la llamada desde el método order\_create de la app orders.



Se puede personalizar los templates de DJANGO:

<https://www.webforefront.com/django/admincustomlayout.html>

Authentication:

<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication>