

```
1 from django.shortcuts import get_object_or_404, render
2 # import the 404 as well
3 from django.core.paginator import EmptyPage, PageNotAnInteger, Paginator
4 # choices at the same folder
5 from .choices import price_choices, state_choices, bedroom_choices
6 # load database into view
7 from .models import Listing
8
9
10 def index(request):
11     listings = Listing.objects.order_by('-
list_date').filter(is_published=True)
12     paginator = Paginator(listings, 3)
13     page = request.GET.get('page')
14     paged_listings = paginator.get_page(page)
15     context = {
16         'listings': paged_listings
17     }
18     return render(request, 'listings/listings.html', context)
19
20 # add additional parameter listing_id route
21
22
23 def listing(request, listing_id):
24     # get an id or return 404
25     listing = get_object_or_404(Listing, pk=listing_id)
26     context = {
27         'listing': listing
28     }
29     return render(request, 'listings/listing.html', context)
30
31
32 def search(request):
33     queryset_list = Listing.objects.order_by('-list_date')
34     # keywords search, step 1 make sure no empty strings
35     # step 2 then look up the keywords
36     if 'keywords' in request.GET:
37         keywords = request.GET['keywords']
38         if keywords:
39             queryset_list = queryset_list.filter(
description__icontains=keywords)
40     # second search field City, need to be exact but case insensitive
41     if 'city' in request.GET:
42         city = request.GET['city']
43         if city:
44             queryset_list = queryset_list.filter(
city__iexact=city)
45     # add state search
46     if 'state' in request.GET:
47         state = request.GET['state']
48         if state:
49             queryset_list = queryset_list.filter(
city__iexact=state)
50     # search bedrooms with less than option
51     if 'bedrooms' in request.GET:
52         bedrooms = request.GET['bedrooms']
53         if bedrooms:
54             queryset_list = queryset_list.filter(
bedrooms__lte=bedrooms)
```

```
59     # search price less than
60     if 'price' in request.GET:
61         price = request.GET['price']
62         if price:
63             queryset_list = queryset_list.filter(
64                 price__lte=price)
65     # keep request object values
66     context = {
67         'state_choices': state_choices,
68         'price_choices': price_choices,
69         "bedroom_choices": bedroom_choices,
70         'listings': queryset_list,
71         'values': request.GET
72     }
73     return render(request, 'listings/search.html', context)
74
```