Speaker: Udemy Instructor Rathan Kumar | Notebook: Django Project: Car Listing



- 1. Use a Django package PAGINATOR to enable page activity.
- 2. Update the CARS\VIEWS.PY

```
×
      EXPLORER
                                        urls.py pages
                                                         urls.py cars M

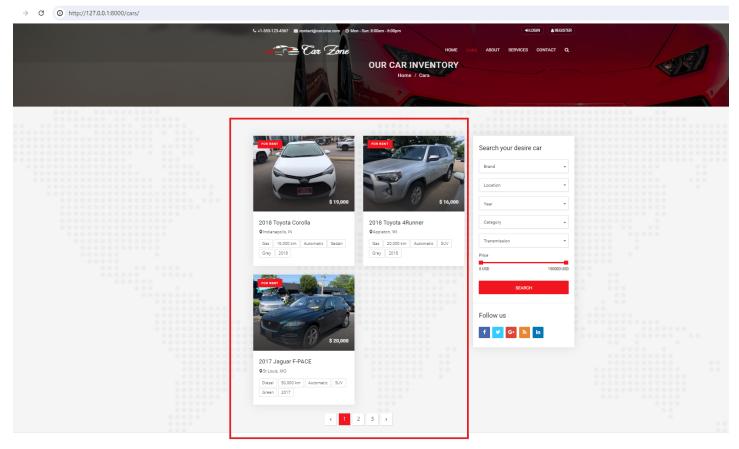
• views.py cars M X  
• car_detail.html U

                                                                                                         cars.html M
    CARZONE-PROJECT
Ф
      v cars
                                       from django.shortcuts import render, get_object_or_404
      > migrations
                                      from _models import Car
                                       from django.core.paginator import EmptyPage, PageNotAnInteger, Paginator
      admin.py
      apps.py
      tests.py
      views.py
                                       def cars(request):
딚
                                            cars = Car.objects.order_by('-created_date')
     > media
      pages
      > _pycache_
                                            paginator = Paginator(cars, 3)
      _init__py
      admin.py
                                            page = request.GET.get('page')
      apps.py
                                            paged_cars = paginator.get_page(page)
      models.py
      tests.py
      urls.py
                                            data = {
      views.py
                                                 'cars': paged_cars,
     > photos
      > Resources
                                  20
                                            return render(request, 'cars/cars.html', data)
     > static
     templates
                                       def car_detail(request, id):
       cars.html
      includes
                                            single_car = get_object_or_404(Car, pk=id)
       o navbar.html
       topbar.html
                                                 'single_car': single_car,
      pages
       about.html
       ocontact.html
       O home.html
                                            return render(request, 'cars/car_detail.html', data)
      o base.html
       .gitignore
     carzone_db

■ db.sqlite3

     manage.py
```

3. Run your CARS.HTML



4. To activate the PAGE BUTTONS, go to CARS.HTML,

FROM:

TO:

Checks if there is a previous page:

Depending on how car items per page is displayed, it can show several page numbers, the loop below will do this:

```
<!---Loops thru multiple page numbers -->

{% for i in cars.paginator.page_range %}

<!-- make the page number active if we are on it-->

{% if cars.number == i %}

class="page-item"><a class="page-link active" href="#">{{ i }}</a>
{% else %}

class="page-item"><a class="page-link" href="!page={{i}}">{{ i }}</a>
{% endif %}

{% endfor %}
```

Checks if there is a next page:

```
{% if cars.has_next %}

{% if cars.has_next %}

{li class="page-item">

{a class="page-link" href="?page={{cars.next_page_number}}">

class="fa fa-angle-right"></a>

{% else %}

{li class="page-item disabled">

{a class="page-link"><i class="fa fa-angle-right"></i>
{% else %}

{li class="page-item disabled">

{a class="page-link"><i class="fa fa-angle-right"></i>
{% endif %}
```

The entire pagination code block should be:

```
_init_py
                                     <div class="col-lg-8 col-md-12">
apps.pymodels.py
                                        <div class="pagination-box p-box-2 text-center">
urls.py
                                                 {% if cars.has_other_pages %}

∨ pages

> migrations

_init_py
                                                              class="fa fa-angle-left"></i>
apps.pymodels.py
                                                     tests.py urls.py
photos
> static
                                                     \label{linear} $$ \class="page-item"><a\cdot class="page-link active" href="#">{{ `i }}</a>
                                                     {% else %}
 navbar.html
                                                     \label{linear} $$ $$ \sim class="page-link" \cdot href="\underline{?page=\{\{i\}\}}" > {\{\{i\}\}\} </a> 
                                                     {% endfor %}

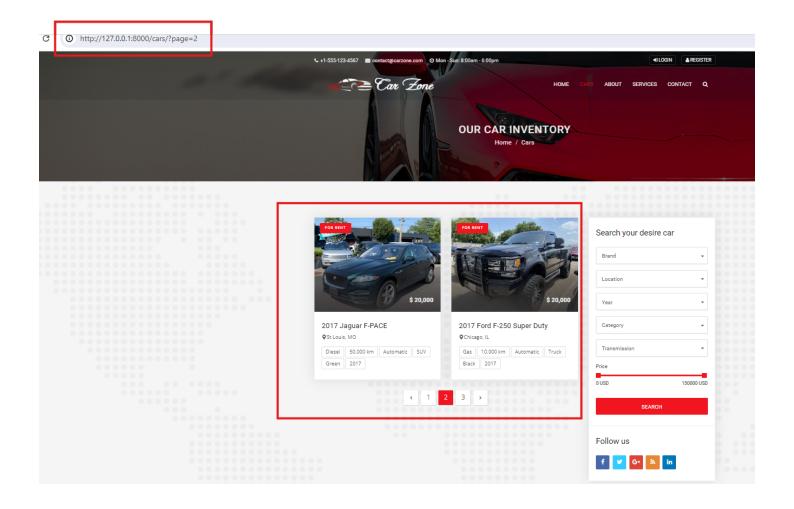
    about.html

 ♦ home.html♦ services.html
                                                     class="page-item">
ø .gitignore≡ carzone_db
                                                        <a class="page-link" href="?page={{cars.next_page_number}}"><i</pre>

    db.sqlite3

                                                               class="fa fa-angle-right"></i></a>
                                                     {% else %}
                                                     {% endif %}
```

5. Run the server.



Copyright © Personal Digital Notebooks | By Rosilie | Date Printed: Oct. 30, 2025, 8:25 a.m.