**Single Car Page and Pagination**

1. Create a url in cars/urls.py

path('<int:id>/',views.car\_detail, name='car\_detail'),

1. Create a view

def car\_detail(request,id):

return render(request,'cars/car\_detail.html')

**Here we are using ID of every car**.

1. Create a Car\_detail html page
2. Put h1 tag in car\_detail and go to the home.html and give the car\_detail url
3. In this <a> tag.

<div class="detail">

<h1 class="title">

<a href="{% url 'car\_details' car.id %}">{{ car.car\_title }}</a>

</h1>

<div class="location">

<a href="{% url 'car\_details' car.id %}">

<i class="flaticon-pin"></i>{{ car.state }} {{ car.city }}

</a>

</div>

**Now copy the car Details from car House file**

1. Copy car detail from the CarHouse project
2. From banner start to car detail end

Use the block content and extends base.html

**Now get all the data from Car model in car\_detail**

**Import the Car model**

**View.py.**

**From .models import Car**

def car\_details(request,id):

single\_car = get\_object\_or\_404(Car,pk=id)

data = {

'single\_car': single\_car,

}

return render(request, 'cars/car\_details.html',data)

**car\_detail.html**

Here you can directly single\_car variable for fetching data dynamically.

You don’t need to use for loop

***Like this you can use in this file.***

<div class="sub-banner overview-bgi">

<div class="container breadcrumb-area">

<div class="breadcrumb-areas">

<h1>{{ single\_car.car\_title }}</h1>

<ul class="breadcrumbs">

<li><a href="{% url 'home' %}">Home</a></li>

<li class="active">{{ single\_car.car\_title }}</li>

</ul>

</div>

</div>

</div>

1. **Delete all the static data and carry only one data**
2. **Fetch all the images and also inner images.**
3. **Fetch the description and use safe with description.**
4. **Humanize etc.**

Setup the car page.

Car.html

First fetch all the data from car model.

**Views.py**

def cars(request):

cars = Car.objects.order\_by('-created\_date')

data = {

'cars' : cars,

}

return render(request,'cars/cars.html',data)

**Car.html**

Do the same as you do previously for single car. But here use the loop.

<div class="col-lg-8 col-md-12">

<div class="row">

{% for car in cars %}

<div class="col-lg-6 col-md-6">

**Pagination**

**Views.py**

Here we change the cars with paged\_car bcz we don’t need now to pass in the context.

def cars(request):

cars = Car.objects.order\_by('-created\_date')

paginator = Paginator(cars,4)

page = request.GET.get('page')

paged\_car = paginator.get\_page(page)

data = {

# we use this below (commenetd) line when we are feching the data for cars but when we fetching the pagination

# so use override the cars to paged\_car in context

# 'cars' : paged\_car,

'cars' : paged\_car,

}

return render(request,'cars/cars.html',data)

**Car.html**

**Look every code of line carefully**

<!-- Page navigation start -->

<div class="pagination-box p-box-2 text-center">

<nav aria-label="Page navigation example">

<!-- here we used cars this is context cars not loop car variable -->

{% if cars.has\_other\_pages %}

<ul class="pagination">

{% if cars.has\_previous %}

<li class="page-item">

<a class="page-link" href="?page={{ cars.previous\_page\_number }}"><i class="fa fa-angle-left"></i></a>

</li>

{% else %}

<!-- it copied from above -->

<li class="page-item disabled">

<a class="page-link"><i class="fa fa-angle-left"></i></a>

</li>

{% endif %}

{% for i in cars.paginator.page\_range %}

{% if cars.number == i %}

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

{% else %}

<li class="page-item"><a class="page-link" href="?page={{i}}">{{i}}</a></li>

{% endif %}

{%endfor%}

{% if cars.has\_next %}

<li class="page-item">

<a class="page-link" href="?page={{ cars.next\_page\_number }}"><i class="fa fa-angle-right"></i></a>

</li>

{% else %}

<!-- it copied from above -->

<li class="page-item disabled">

<a class="page-link"><i class="fa fa-angle-right"></i></a>

</li>

{% endif %}

</ul>

{% endif %}

</nav>

</div>

</div>