


python

 Copy code

```
# views.py
from django.core.paginator import Paginator
from django.shortcuts import render
from .models import YourModel


def your_view(request):
    queryset = YourModel.objects.all()

    # Number of items per page
    items_per_page = 10

    paginator = Paginator(queryset, items_per_page)
    page_number = request.GET.get('page')
    page_obj = paginator.get_page(page_number)

    return render(request, 'your_template.html', {'page_obj': page_obj})
```

html

 Copy code

```
<!-- your_template.html -->
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Your Page</title>
</head>
<body>
  <h1>Your Paginated Content</h1>

  <!-- Display paginated content -->
  {% for item in page_obj %}
    <p>{{ item }}</p>
  {% endfor %}
```

```

<!-- Pagination navigation -->
{% if page_obj.has_other_pages %}
    <div class="pagination">
        <span class="step-links">
            {% if page_obj.has_previous %}
                <a href="?page=1">&laquo; first</a>
                <a href="?page={{ page_obj.previous_page_number }}">previous</a>
            {% endif %}

            <span class="current">
                Page {{ page_obj.number }} of {{ page_obj.paginator.num_pages }}.
            </span>

            {% if page_obj.has_next %}
                <a href="?page={{ page_obj.next_page_number }}">next</a>
                <a href="?page={{ page_obj.paginator.num_pages }}">last &raquo;</a>
            {% endif %}
        </span>
    </div>
{% endif %}
</body>
</html>

```

In this example:

- We use Django's Paginator class to split the queryset into pages with a specified number of items per page.
- In the template, we loop through the paginated content using the `page_obj` variable.
- We provide navigation links to navigate between pages, using query parameters in the URL to specify the current page number.

By following these steps, you can implement pagination in your Django project to divide content across multiple pages. Adjust the code according to your specific models, views, and templates.