models.py

```python
from django.db import models


class ClassBlog(models.Model):
    title = models.CharField(max_length=100)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
    body = models.TextField()


    def __str__(self):
        return self.title
```

admin.py

```python
from django.contrib import admin
from .models import ClassBlog


admin.site.register(ClassBlog)
```

crud (proj) urls.py

```python
from django.contrib import admin
from django.urls import path, include
import viewcrud.urls
import viewcrud.views
import classcrud.urls
import classcrud.views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', viewcrud.views.welcome, name="welcome"),
    path('funccrud/', include(viewcrud.urls)),
    path('classcrud/', include(classcrud.urls)),
]
```

class crud (app) urls.py

```python
from django.urls import path
from . import views

urlpatterns = [

    path('', views.BlogView.as_view(), name='list'),
    path('newblog/', views.BlogCreate.as_view(), name='new'),
    path('detail/<int:pk>', views.BlogDetail.as_view(), name='detail'),
    path('update/<int:pk>', views.BlogUpdate.as_view(), name='change'),
    path('delete/<int:pk>', views.BlogDelete.as_view(), name='del'),


]
```

views.py

```python
from django.shortcuts import render
from django.utils import timezone

from .models import ClassBlog
from django.urls import reverse_lazy
from django.views.generic.list import ListView # 데이터 보여주기
from django.views.generic.detail import DetailView
from django.views.generic.edit import CreateView, UpdateView, DeleteView #데이터 추가

class BlogView(ListView):
    model = ClassBlog

class BlogCreate(CreateView):
    model = ClassBlog
    fields = ['title', 'body']
    success_url = reverse_lazy('list')

class BlogDetail(DetailView):
    model = ClassBlog

class BlogUpdate(UpdateView):
    model = ClassBlog
    fields = ['title', 'body']
    success_url = reverse_lazy('list')

class BlogDelete(DeleteView):
    model = ClassBlog
    success_url = reverse_lazy('list')
```

templates/app이름

classblog_list.html

```html
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="utf-8">
    <title>싀알류디연습</title>
</head>
<body>
<a href="{% url 'new' %}"> 추가 </a>


<br>
{%for blog in object_list%}
    <h2>{{blog.title}}</h2>
    <h5>{{blog.created_at}}</h5>
    <p>{{blog.body}}</p>

    <a href = "{% url 'detail' blog.id %}" > 자세히보기 </a> <br>
    <a href = "{% url 'change' blog.id %}" > 수정 </a> <br>
    <a href = "{% url 'del' blog.id %}" > 삭제 </a> <br>

    <hr>
{%endfor%}


</body>
</html>
```

classblog_detail.html

```html
<h1>본문 자세히 보기 페이지</h1>
<hr>
<h2> 제목 : {{object.title}} </h2>
<h4> 날짜 : {{object.created_at}}</h4>
본문 : {{object.body}}
```

classblog_form.html

```html
<form method='POST'>
    {% csrf_token %}
    {{form.as_p}}
    <input type="submit" value="추가하기">
</form>
```

classblog_confirm_delete.html

```html
<form method='POST'>
    {% csrf_token %}
    진짜루다가 지울거야? <br><br>
    <input type = "submit" value ="어 진짜 지울거야">
</form>
```


pagination

```python
class BlogView(ListView):
    #template_name = 'classcrud/list.html'
    model = ClassBlog
    paginate_by = 3
```

```django
{% if is_paginated %}
    {% if page_obj.has_previous %}
        <a href="?page={{ page_obj.previous_page_number }}">
            <span>Prev</span>
        </a>
    {% endif %}

    {% if page_obj.has_next %}
        <a href="?page={{ page_obj.next_page_number }}">
            <span>Next</span>
        </a>
    {% endif %}
{% endif %}
```