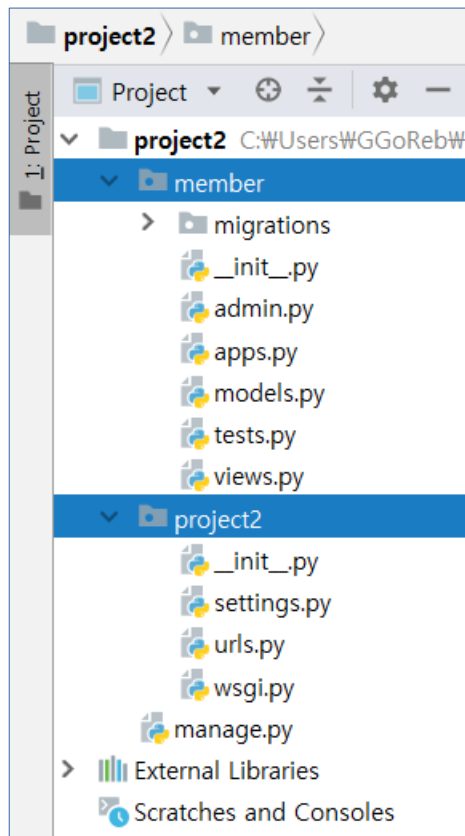


■ 회원가입 / 로그인

● 프로젝트 / 앱 생성

- `django-admin startproject project2`
- `python manage.py startapp member`



■ 회원가입

● 회원 정보 입력 페이지

– member/templates/member/join.html

```
<form method="post">
  {% csrf_token %}
  아이디 : <input type="text" name="user_id"><br>
  비밀번호 : <input type="text" name="user_pw"><br>
  이름 : <input type="text" name="user_name"><br>
  <input type="submit" value="가입">
</form>
```

– member/views.py

```
def join(request):
    if request.method == 'GET':
        return render(request, 'member/join.html', {})
    else:
        user_id = request.POST['user_id']
        user_pw = request.POST['user_pw']
        user_name = request.POST['user_name']
        return HttpResponse('가입 완료' + user_id + user_pw + user_name)
```

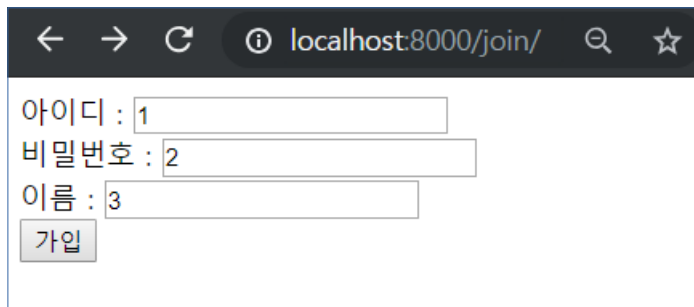
■ 회원가입

● 회원 정보 입력 페이지

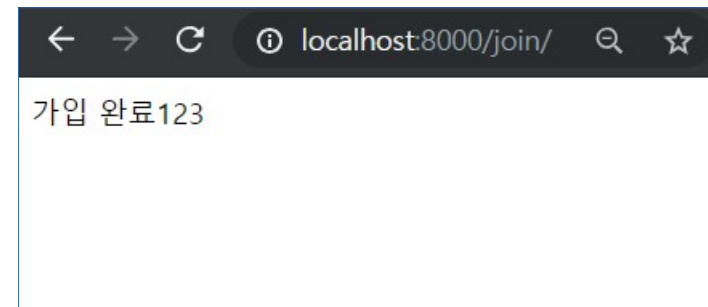
– project2/urls.py

```
from django.contrib import admin
from django.urls import path
from member.views import *

urlpatterns = [
    path('admin/', admin.site.urls),
    path('join/', join, name='join'),
]
```



A screenshot of a web browser window showing a registration form. The address bar displays 'localhost:8000/join/'. The form contains three input fields labeled '아이디 : 1', '비밀번호 : 2', and '이름 : 3'. Below these fields is a button labeled '가입'.



A screenshot of a web browser window showing the result of the registration. The address bar displays 'localhost:8000/join/'. The page content shows the text '가입 완료123'.

■ 회원가입

● 회원 정보 저장






– member/models.py

```
from django.db import models

class Member(models.Model):
    user_id = models.CharField(max_length=50, primary_key=True)
    user_pw = models.CharField(max_length=50)
    user_name = models.CharField(max_length=50)
    c_date = models.DateTimeField()
```

– python manage.py makemigrations

– python manage.py migrate

 member_member	
 user_id	varchar(50)
 user_pw	varchar(50)
 user_name	varchar(50)
 c_date	datetime

■ 회원가입

● 회원 정보 저장 - 1 (Member 생성 및 저장)

- member/views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from .models import Member
from django.utils import timezone
```

```
def join(request):
```

```
    if request.method == 'GET':
```

```
        return render(request, 'member/join.html', {})
```

```
    else:
```

```
        user_id = request.POST['user_id']
```

```
        user_pw = request.POST['user_pw']
```

```
        user_name = request.POST['user_name']
```

```
        member = Member(user_id=user_id, user_pw=user_pw, user_name=user_name)
```

```
        member.c_date = timezone.now()
```

```
        member.save()
```

```
        return HttpResponse('가입 완료' + user_id + user_pw + user_name)
```

	user_id	user_pw	user_name	c_date
	필터	필터	필터	필터
1	a	1	aaa	2018-10-22 06:11

■ 회원가입

● 회원 정보 저장 - 2 (form 활용)

- member/forms.py

```
from django.forms import forms, TextInput, PasswordInput
import django.forms as forms
from .models import Member

class JoinForm(forms.ModelForm):
    class Meta:
        model = Member
        fields = ['user_id', 'user_pw', 'user_name']
        widgets = {
            'user_id': TextInput(),
            'user_pw': PasswordInput(),
            'user_name': TextInput()
        }
```

■ 회원가입

● 회원 정보 저장 - 2 (form 활용)

- member/views.py

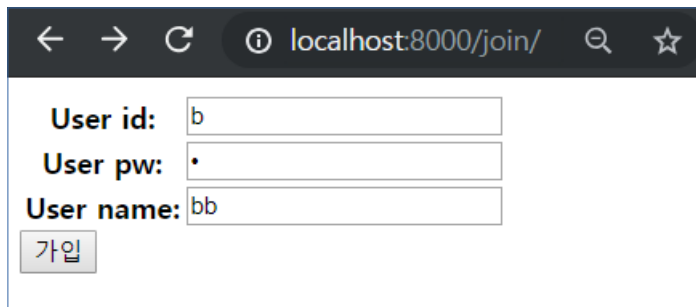
```
...
def join(request):
    if request.method == 'GET':
        form = JoinForm()
        return render(request, 'member/join.html', {'form': form})
    else:
        form = JoinForm(request.POST)
        if form.is_valid():
            member = form.save(commit=False)
            member.c_date = timezone.now()
            member.save()
            return HttpResponse('가입 완료')
        else:
            return HttpResponse('오류')
```

■ 회원가입

● 회원 정보 저장 - 2 (form 활용)

- member/templates/member/join.html

```
<form method="post">
  {% csrf_token %}
  <table>
    {{ form.as_table }}
  </table>
  <input type="submit" value="가입">
</form>
```

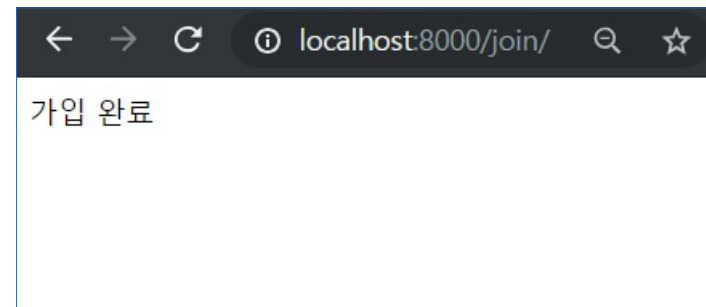


← → ↻ ⓘ localhost:8000/join/ 🔍 ☆

User id:

User pw:

User name:



← → ↻ ⓘ localhost:8000/join/ 🔍 ☆

가입 완료

■ 로그인

● 로그인 페이지

– member/templates/member/login.html

```
<form method="post">
  {% csrf_token %}
  <table>
    {{ form.as_table }}
  </table>
  <input type="submit" value="로그인">
</form>
```

– member/forms.py

```
class LoginForm(forms.ModelForm):
    class Meta:
        model = Member
        fields = ['user_id', 'user_pw']
        widgets = {
            'user_id': TextInput(),
            'user_pw': PasswordInput(),
        }
```

■ 로그인

● 로그인 페이지

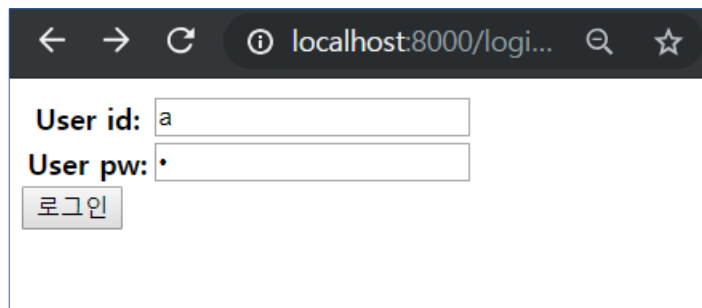
– member/views.py

```
def login(request):
    if request.method == 'GET':
        form = LoginForm()
        return render(request, 'member/login.html', {'form': form})
    else:
        user_id = request.POST['user_id']
        user_pw = request.POST['user_pw']
        member = Member.objects.get(user_id=user_id, user_pw=user_pw)
        if member:
            return HttpResponseRedirect('로그인 성공')
        else:
            return HttpResponseRedirect('로그인 실패')
```

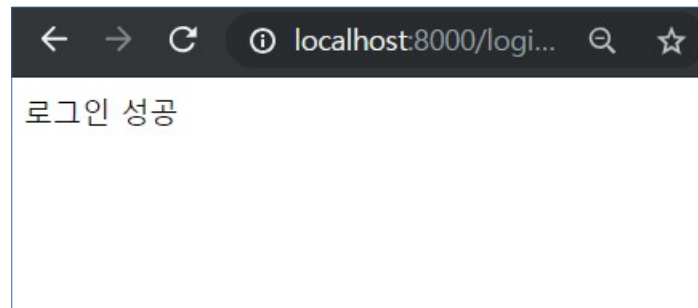
■ 로그인

● 로그인 페이지

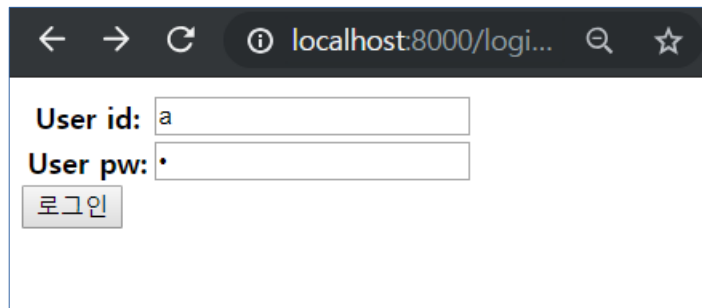
– 입력 정보가 맞는 경우



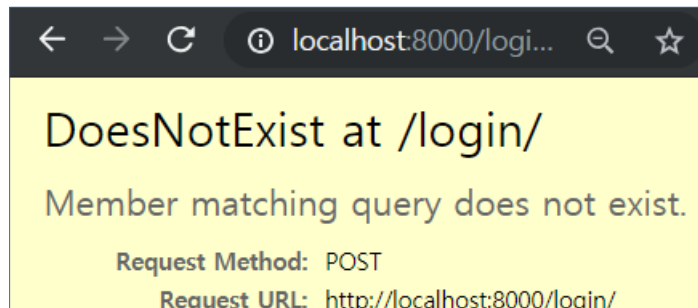
A screenshot of a web browser window with the address bar showing 'localhost:8000/logi...'. The page contains a login form with two input fields: 'User id:' with the value 'a' and 'User pw:' with a masked password '•'. Below the fields is a button labeled '로그인' (Login).



– 입력 정보가 맞지 않는 경우



A screenshot of a web browser window with the address bar showing 'localhost:8000/logi...'. The page contains a login form with two input fields: 'User id:' with the value 'a' and 'User pw:' with a masked password '•'. Below the fields is a button labeled '로그인' (Login).



■ 로그인

● 로그인 페이지

– member/views.py

```
def login(request):
    if request.method == 'GET':
        form = LoginForm()
        return render(request, 'member/login.html', {'form': form})
    else:
        user_id = request.POST['user_id']
        user_pw = request.POST['user_pw']

        try:
            Member.objects.get(user_id=user_id, user_pw=user_pw)
        except Member.DoesNotExist:
            return HttpResponse('로그인 실패')
        else:
            return HttpResponse('로그인 성공')
```

■ 로그인

● 로그인 후 메인페이지 이동

- member/templates/member/main.html

```
<h1>Main</h1>
```

- member/views.py

```
def main(request):  
    return render(request, 'member/main.html', {})
```

- project2/urls.py

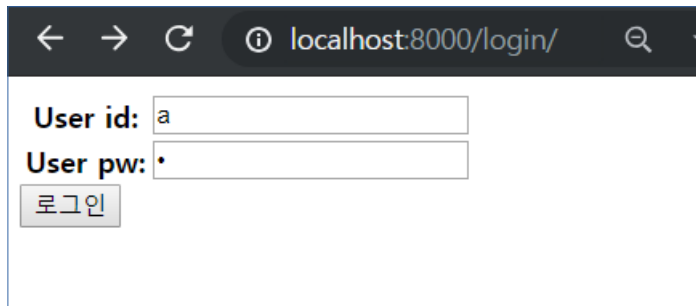
```
urlpatterns = [  
    ...  
    path('main/', main, name='main'),  
]
```

■ 로그인

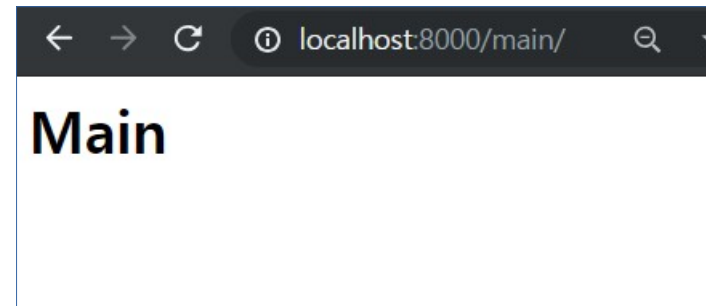
● 로그인 후 메인페이지 이동

– member/views.py

```
from django.shortcuts import render, redirect
...
def login(request):
    ...
    try:
        member = Member.objects.get(user_id=user_id, user_pw=user_pw)
    except Member.DoesNotExist:
        return HttpResponse('로그인 실패')
    else:
        return redirect('main')
```



A screenshot of a web browser showing the login page at localhost:8000/login/. The page has a dark header with navigation icons and the URL. Below the header, there are two input fields: "User id:" with the value "a" and "User pw:" with a masked password "•". A "로그인" (Login) button is located below the password field.



■ 로그인

● 세션 활용

– member/views.py

```
from django.shortcuts import render, redirect
...
def login(request):
    ...
    try:
        member = Member.objects.get(user_id=user_id, user_pw=user_pw)
        request.session['user_name'] = member.user_name
        request.session['user_id'] = member.user_id
    except Member.DoesNotExist:
        return HttpResponse('로그인 실패')
    else:
        return redirect('main')
```

■ 로그인

● 세션 활용

– member/templates/member/main.html

```
{% if request.session.user_name %}
    {{ request.session.user_name }}님 반갑습니다.
{% endif %}

<h1>Main</h1>
```

