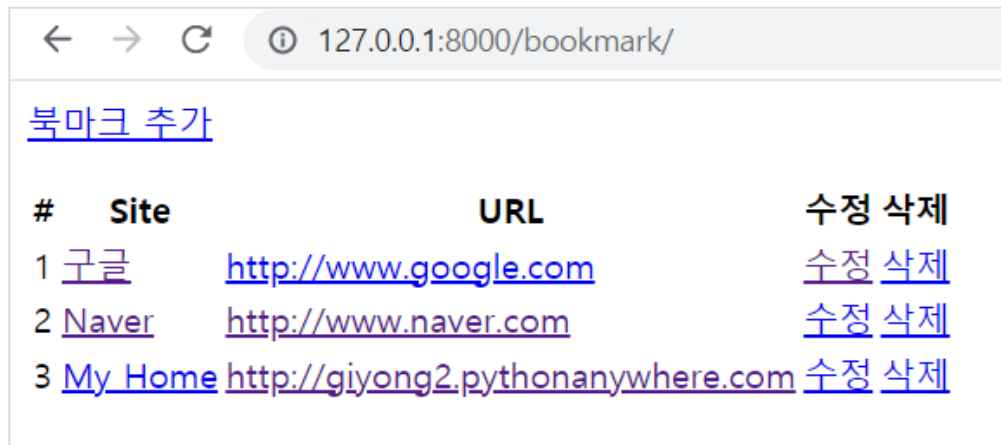


# bookmark 사이트 만들기

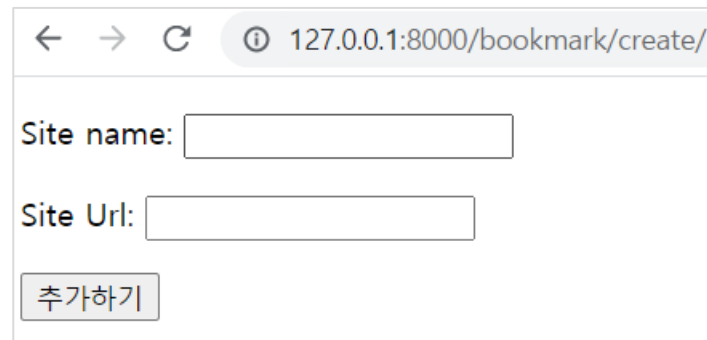


# bookmark 사이트 만들기

## ❖ bookmark 목록 / 추가하기



#	Site	URL	수정	삭제
1	<a href="#">구글</a>	<a href="http://www.google.com">http://www.google.com</a>	<a href="#">수정</a>	<a href="#">삭제</a>
2	<a href="#">Naver</a>	<a href="http://www.naver.com">http://www.naver.com</a>	<a href="#">수정</a>	<a href="#">삭제</a>
3	<a href="#">My_Home</a>	<a href="http://giyong2.pythonanywhere.com">http://giyong2.pythonanywhere.com</a>	<a href="#">수정</a>	<a href="#">삭제</a>



← → ↻ ⓘ 127.0.0.1:8000/bookmark/create/

Site name:

Site Url:

# bookmark 사이트 만들기

## ❖ bookmark 상세페이지 / 수정 / 삭제

← → ↻ ⓘ 127.0.0.1:8000/bookmark/1/

사이트 이름 : 구글

사이트 주소 : <http://www.google.com>

← → ↻ ⓘ 127.0.0.1:8000/bookmark/update/1

Site name:

Site Url:

← → ↻ ⓘ 127.0.0.1:8000/bookmark/delete/2/

이름 : Naver, 주소 : <http://www.naver.com>를 정말로 삭제 하시겠습니까?

# bookmark 사이트 만들기

## ❖ django 프로젝트 process

1. projects 디렉터리 하위에 bookmark 디렉터리에 만들기

```
(mysite) > C:\projects>mkdir bookmark
```

2. bookmark 디렉터리에 새 프로젝트 config 앱 설치하기

```
(mysite) > C:\projects\ bookmark >django-admin startproject config .
```

3. 서버를 구동해서 127.0.0.1:8000 페이지 확인

```
(mysite) > C:\projects\ bookmark >python manage.py runserver
```

# bookmark 사이트 만들기

## ❖ django 프로젝트 process

### 4. DB 사용을 위한 migrate 실행하기

```
(mysite) > C:\projects\ bookmark >python manage.py migrate
```

### 5. 관리자 페이지 사용을 위한 슈퍼유저 생성하기

```
(mysite) > C:\projects\ bookmark >python manage.py createsuperuser
```

### 6. bookmark 앱 생성하기

```
(mysite) > C:\projects\ bookmark >django-admin startapp bookmark
```

# bookmark 사이트 만들기

## ❖ bookmark config/settings.py

```
LANGUAGE_CODE = 'ko-kr'
```

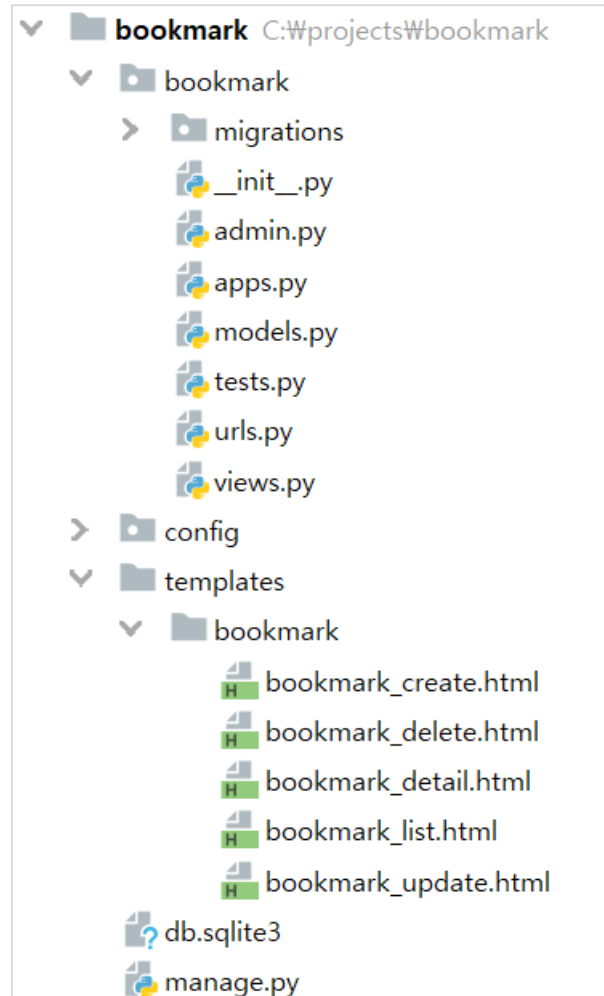
```
TIME_ZONE = 'Asia/Seoul'
```

```
INSTALLED_APPS = [  
    'bookmark',  
    'django.contrib.admin',  
    'django.contrib.auth',
```

```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': [BASE_DIR / 'templates'],  
        'APP_DIRS': True,
```

# bookmark 사이트 만들기

## ❖ bookmark 프로젝트 구성도



# bookmark 사이트 만들기

## ❖ bookmark 모델(model)

```
from django.db import models

class Bookmark(models.Model):
    site_name = models.CharField(max_length=100)
    url = models.URLField('Site Url')

    def __str__(self):
        return "이름 : " + self.site_name + ", 주소 : " + self.url
```



# bookmark 사이트 만들기

## ❖ bookmark config/urls.py

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    # 127.0.0.1/admin
    path('admin/', admin.site.urls),
    # 127.0.0.1/bookmark
    path('bookmark/', include('bookmark.urls'))
]
```

# bookmark 사이트 만들기

## ❖ bookmark URL 매핑 – bookmark/urls.py

```
app_name = 'bookmark'

urlpatterns = [
    # 북마크 목록 - 127.0.0.1/bookmark
    path('', BookmarkList.as_view(), name='list'),

    # 북마크 추가 - 127.0.0.1/create
    path('create/', BookmarkCreate.as_view(), name='create'),

    # 상세 페이지 - 127.0.0.1/2
    path('<int:pk>/', BookmarkDetail.as_view(), name='detail'),

    # 북마크 수정 - 127.0.0.1/update
    path('update/<int:pk>/', BookmarkUpdate.as_view(), name='update'),

    # 북마크 삭제 - 127.0.0.1/delete
    path('delete/<int:pk>/', BookmarkDelete.as_view(), name='delete'),
]
```

# bookmark 사이트 만들기

## ❖ bookmark view 함수

```
class BookmarkList(ListView):
    model = Bookmark

class BookmarkCreate(CreateView):
    model = Bookmark
    fields = ['site_name', 'url']
    # 성공했을때 페이지 이동(redirect)
    success_url = reverse_lazy('bookmark:list')
    template_name_suffix = '_create' # suffix: 접미사

class BookmarkDetail(DetailView):
    model = Bookmark

class BookmarkUpdate(UpdateView):
    model = Bookmark
    fields = ['site_name', 'url']
    success_url = reverse_lazy('bookmark:list')
    template_name_suffix = '_update'

class BookmarkDelete(DeleteView):
    model = Bookmark
    success_url = reverse_lazy('bookmark:list')
```

# bookmark 사이트 만들기

## ❖ bookmark 템플릿 – bookmark\_list.html

```
<p><a href="{% url 'bookmark:create' %}">북마크 추가</a></p>
<table>
  <thead>
    <tr>
      <th>#</th><th>Site</th><th>URL</th><th>수정</th><th>삭제</th>
    </tr>
  </thead>
  <tbody>
    {% for bookmark in object_list %}
      <tr>
        <td>{{ forloop.counter }}</td>
        <td>
          <a href="{% url 'bookmark:detail' pk=bookmark.id %}">{{ bookmark.site_name }}</a>
        </td>
        <td><a href="{% bookmark.url %}" target="_blank">{{ bookmark.url }}</a></td>
        <td><a href="{% url 'bookmark:update' pk=bookmark.id %}">수정</a> </td>
        <td><a href="{% url 'bookmark:delete' pk=bookmark.id %}">삭제</a> </td>
      </tr>
    {% endfor %}
  </tbody>
</table>
```

# bookmark 사이트 만들기

bookmark\_detail.html

```
<h3> 사이트 이름 : {{ object.site_name }} </h3>  
<h3> 사이트 주소 : {{ object.url }}</h3>
```

bookmark\_create.html

```
<form action="" method="post">  
    {% csrf_token %}  
    {{ form.as_p }}  
    <button type="submit">추가하기</button>  
</form>
```

# bookmark 사이트 만들기

bookmark\_update.html

```
<form action="" method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">수정하기</button>
</form>
```

bookmark\_confirm\_delete.html

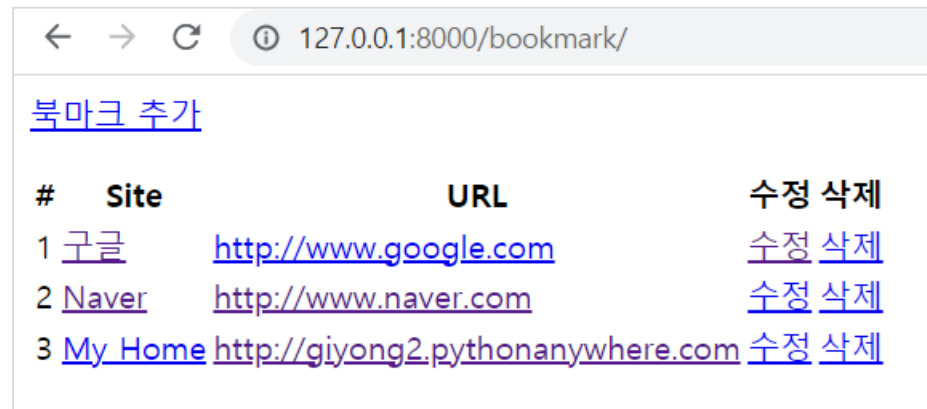
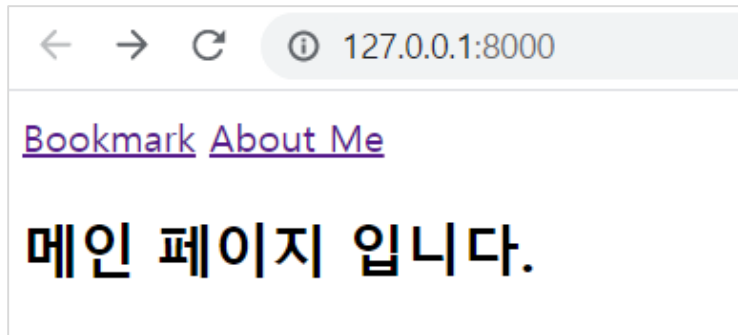
```
<form action="" method="post">
    {% csrf_token %}
    <p>{{ object }}를 정말로 삭제 하시겠습니까?</p>
    <button type="submit">삭제하기</button>
</form>
```

반드시 이름은  
bookmark\_confirm\_delete  
로 해야함

TemplateDoesNotExist at /bookmark/delete/4/  
bookmark/bookmark\_confirm\_delete.html

Request Method: GET  
Request URL: http://127.0.0.1:8000/bookmark/delete/4/  
Django Version: 3.2  
Exception Type: TemplateDoesNotExist

# index 페이지 만들기



# index 페이지 만들기

## 1. front 앱 생성하기

```
(mysite) > C:\projects\bookmark > django-admin startapp front
```

## 2. config/urls.py 에 등록하기

```
urlpatterns = [  
    # 127.0.0.1:8000/admin  
    path('admin/', admin.site.urls),  
    # 127.0.0.1:8000/bookmark  
    path('bookmark/', include('bookmark.urls')),  
  
    # 127.0.0.1:8000/  
    path('', include('front.urls'))  
]
```



# index 페이지 만들기

## 3. Index페이지 URL 매핑하기 - front/urls.py

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

app_name = 'front'

urlpatterns = [
    path('', views.index, name='index')
]
```

# index 페이지 만들기

## 4. index 함수 정의하기 - front/views.py

```
from django.shortcuts import render

def index(request):
    return render(request, 'front/index.html')
```

## 5. 템플릿 만들기 – templates/front/index.html

```
<body>
<nav>
    <a href="/bookmark/">Bookmark</a>
    <a href="#">About Me</a>
</nav>
    <h2>메인 페이지 입니다.</h2>
</body>
```