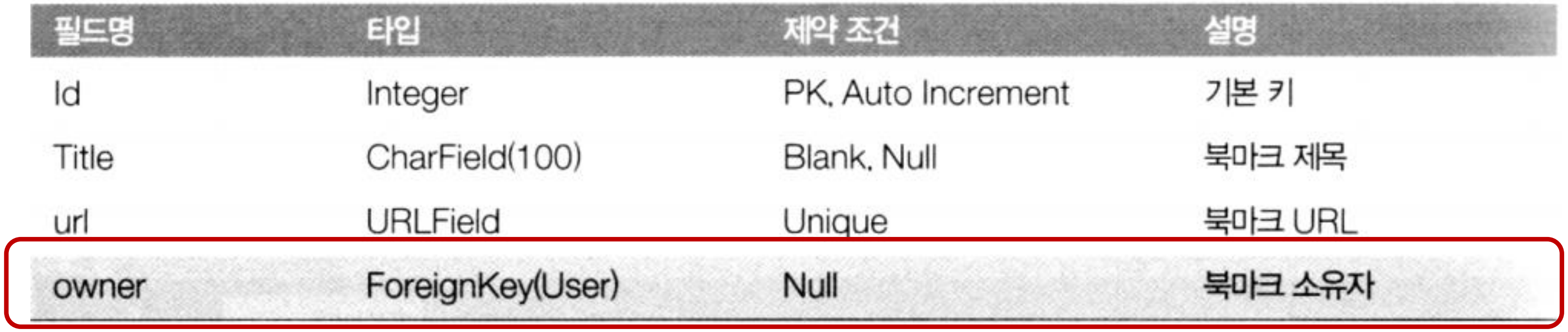
실전 프로그램 개발
- 콘텐츠 편집 기능 -

(Bookmark)

애플리케이션 설계하기

❖ 모델 설계

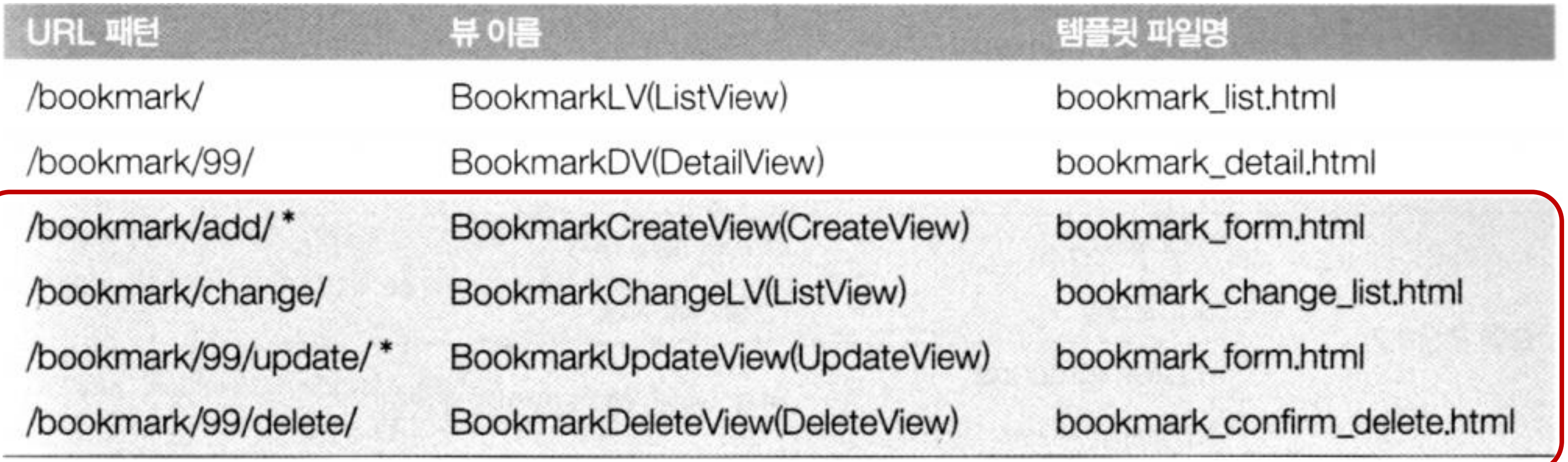
- Bookmark 모델 클래스



필드명	타입	제약 조건	설명
Id	Integer	PK, Auto Increment	기본 키
Title	CharField(100)	Blank, Null	북마크 제목
url	URLField	Unique	북마크 URL
owner	ForeignKey(User)	Null	북마크 소유자

애플리케이션 설계하기

❖ Bookmark URL 설계



URL 패턴	뷰 이름	템플릿 파일명
/bookmark/	BookmarkLV(ListView)	bookmark_list.html
/bookmark/99/	BookmarkDV(DetailView)	bookmark_detail.html
/bookmark/add/ *	BookmarkCreateView(CreateView)	bookmark_form.html
/bookmark/change/	BookmarkChangeLV(ListView)	bookmark_change_list.html
/bookmark/99/update/ *	BookmarkUpdateView(UpdateView)	bookmark_form.html
/bookmark/99/delete/	BookmarkDeleteView(DeleteView)	bookmark_confirm_delete.html

개발 코딩하기

❖ **bookmark/models.py**

```
from django.db import models
from django.contrib.auth.models import User


class Bookmark(models.Model):
  title = models.CharField('TITLE', max_length=100, blank=True)
  url = models.URLField('URL', unique=True)

  owner = models.ForeignKey(User, on_delete=models.CASCADE, blank=True,
                            null=True)

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

개발 코딩하기

❖ 데이터베이스 반영

```
$ python manage.py makemigrations bookmark
$ python manage.py migrate
```

개발 코딩하기

❖ 접근 권한 제어 뷰 클래스

- django.contrib.auth.mixins.LoginRequiredMixin
  - 로그인 사용자만 뷰가 작동하도록 함
  - 로그인 하지 않고 해당 url로 직접 접근하면 로그인 페이지로 리다이렉트 됨

- OwnerOnlyMixin
  - django.contrib.auth.mixins.AccessMixin를 상속
  - 로그인 사용자가 모델 인스턴스의 소유자인 경우에만 뷰가 작동하도록 함

개발 코딩하기

❖ **mysite/views.py**

```
:
from django.contrib.auth.mixins import AccessMixin
from django.views.defaults import permission_denied

:

class OwnerOnlyMixin(AccessMixin):
  raise_exception = True
  permission_denied_message = "Owner only can update/delete the object"

  def get(self, request, *args, **kwargs):
    self.object = self.get_object()    # 모델 인스턴스 얻기
    if self.request.user != self.object.owner:
      self.handle_no_permission()
    return super().get(request, *args, **kwargs)
```

개발 코딩하기

❖ bookmark/views.py

```
from django.views.generic import ListView, DetailView
from bookmark.models import Bookmark
from django.views.generic import CreateView, UpdateView, DeleteView
from django.contrib.auth.mixins import LoginRequiredMixin
from django.urls import reverse_lazy
from mysite.views import OwnerOnlyMixin

:

class BookmarkCreateView(LoginRequiredMixin, CreateView):
    model = Bookmark
    fields = ['title', 'url'] # 폼 모델에 사용할 필드 → 폼 모델 자동 생성
    success_url = reverse_lazy('bookmark:index')

    def form_valid(self, form):
        form.instance.owner = self.request.user
        return super().form_valid(form)
```

개발 코딩하기

❖ **bookmark/views.py**

```
class BookmarkChangeLV(LoginRequiredMixin, ListView):
    template_name = 'bookmark/bookmark_change_list.html'

    def get_queryset(self):
        return Bookmark.objects.filter(owner=self.request.user)


class BookmarkUpdateView(OwnerOnlyMixin, UpdateView):
    model = Bookmark
    fields = ['title', 'url'] # 폼 모델에 사용할 필드 → 폼 모델 자동 생성
    success_url = reverse_lazy('bookmark:index')


class BookmarkDeleteView(OwnerOnlyMixin, DeleteView):
    model = Bookmark
    success_url = reverse_lazy('bookmark:index')
```

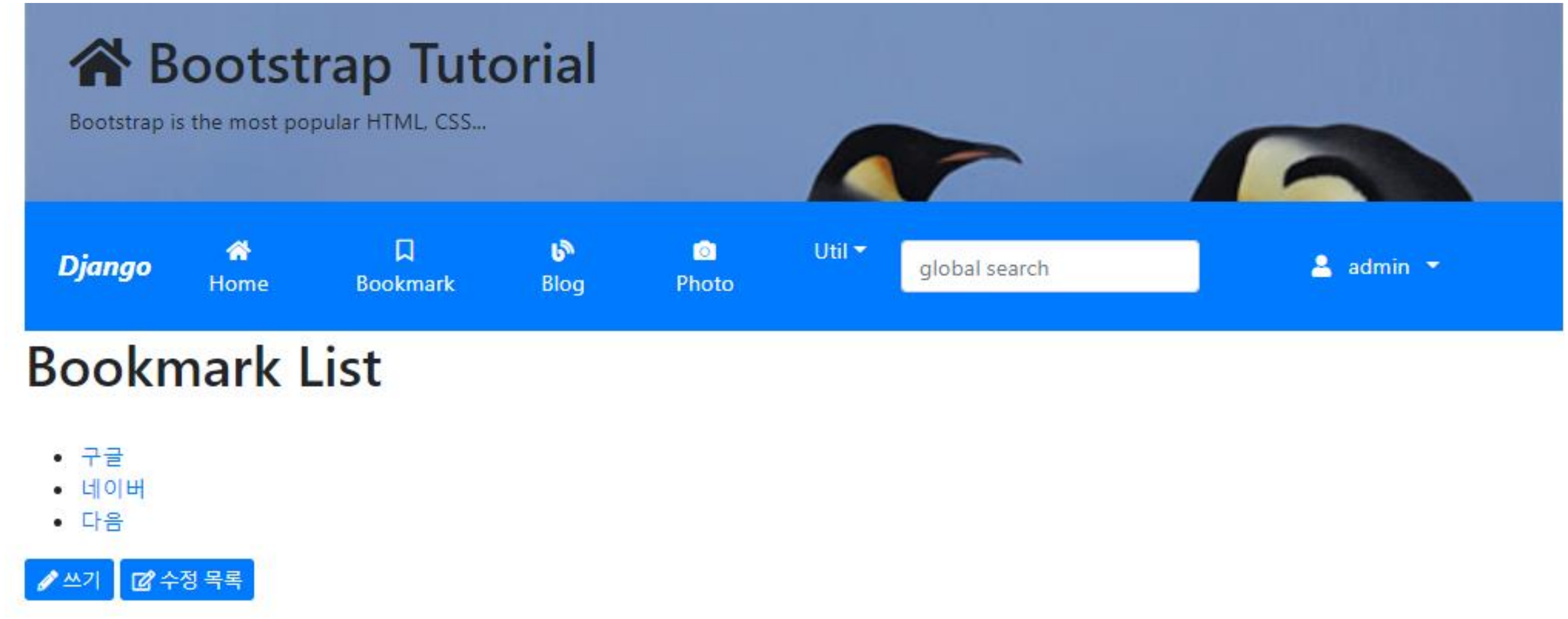
개발 코딩하기

❖ bookmark/urls.py

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

app_name = 'bookmark'
urlpatterns = [
     :
  # Example: /bookmark/add/
  path('add/', BookmarkCreateView.as_view(), name="add"),

  # Example: /bookmark/change/
  path('change/', BookmarkChangeLV.as_view(), name="change"),

  # Example: /bookmark/99/update/
  path('<int:pk>/update/', BookmarkUpdateView.as_view(), name="update"),

  # Example: /bookmark/99/delete/
  path('<int:pk>/delete/', BookmarkDeleteView.as_view(), name="delete"),

]
```

개발 코딩하기

❖ 쓰기 / 수정목록 인터페이스



Bookmark List

- 구글
- 네이버
- 다음

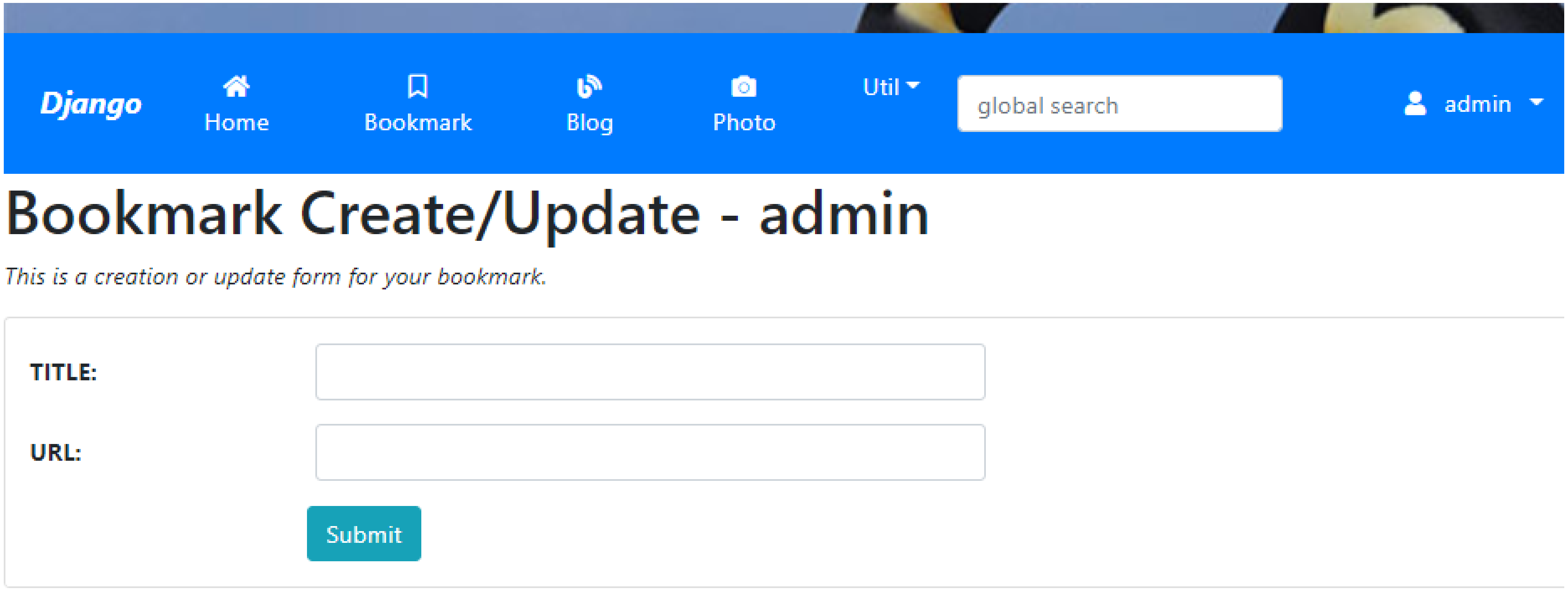
개발 코딩하기

❖ **bookmark/templastes/bookmark/bookmark_list.html**

```
   :

 <div>
    {% if user.is_active %}
     <a href="{%url 'bookmark:add' %}" class="btn btn-primary btn-sm">
        <i class="fas fa-pencil-alt"></i> 쓰기</a>

     <a href="{% url 'bookmark:change'%}"
       class="btn btn-primary btn-sm"><i class="far fa-edit"></i> 수정 목록</a>
    {% endif %}
  </div>
{% endblock %}
```

개발 코딩하기(쓰기/수정)

❖ 북마크 쓰기/수정



Bookmark Create/Update - admin

This is a creation or update form for your bookmark.

TITLE:	
URL:	

개발 코딩하기(쓰기/수정)

❖ **bookmark/templastes/bookmark/bookmark_form.html**

```
{% extends "base.html" %}
{% load widget_tweaks %}

{% block title %}bookmark_form.html{% endblock %}

{% block content %}

  <h1>Bookmark Create/Update - {{user}}</h1>
  <p class="font-italic">This is a creation or update form for your
bookmark.</p>

  {% if form.errors %}
  <div class="alert alert-danger">
    <div class="font-weight-bold">
        Wrong! Please correct the error(s) below.</div>
    {{ form.errors }}
  </div>
  {% endif %}
```

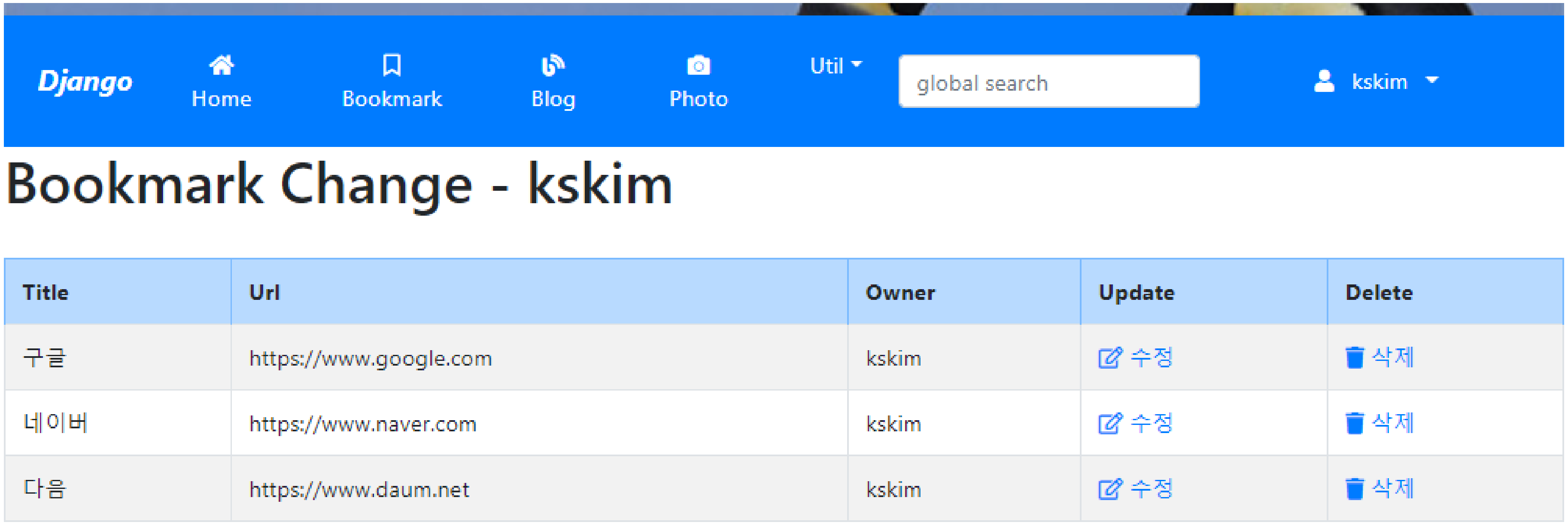
개발 코딩하기(쓰기/수정)

❖ **bookmark/templastes/bookmark/bookmark_form.html**

```
  <form action="." method="post" class="card pt-3">{% csrf_token %}
    <div class="form-group row">
      {{ form.title|
         add_label_class:"col-form-label col-sm-2 ml-3 font-weight-bold" }}
      <div class="col-sm-5">
        {{ form.title|add_class:"form-control"|attr:"autofocus" }}
      </div>
    </div>
    <div class="form-group row">
      {{ form.url|
         add_label_class:"col-form-label col-sm-2 ml-3 font-weight-bold" }}
      <div class="col-sm-5">
        {{ form.url|add_class:"form-control" }}
      </div>
    </div>
    <div class="form-group">
      <div class="offset-sm-2 col-sm-5">
        <input type="submit" value="Submit" class="btn btn-info"/>
      </div>
    </div>
  </form>
{% endblock %}
```

개발 코딩하기(쓰기/수정)

❖ 수정목록



Bookmark Change - kskim

Title	Url	Owner	Update	Delete
구글	https://www.google.com	kskim	수정	삭제
네이버	https://www.naver.com	kskim	수정	삭제
다음	https://www.daum.net	kskim	수정	삭제

개발 코딩하기(쓰기/수정)

❖ **bookmark/templastes/bookmark/bookmark_change_list.html**

```
{% extends "base.html" %}

{% block title %}bookmark_change_list.html{% endblock %}

{% block content %}

  <h1>Bookmark Change - {{user}}</h1>
  <br>

  <table class="table table-bordered table-condensed table-striped">

    <thead>
      <tr class="table-primary">
        <th>Title</th>
        <th>Url</th>
        <th>Owner</th>
        <th>Update</th>
        <th>Delete</th>
      </tr>
    </thead>
```

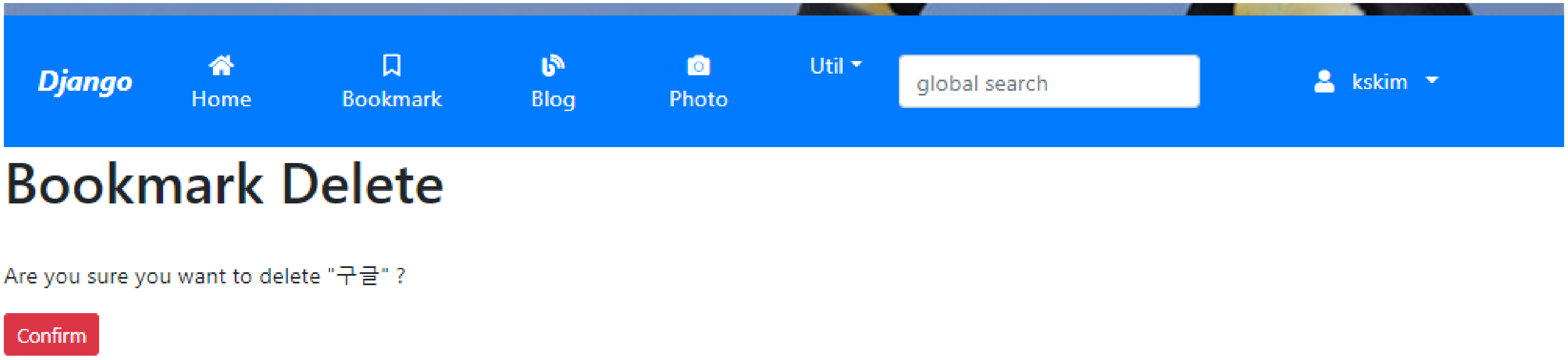
개발 코딩하기(쓰기/수정)

❖ **bookmark/templastes/bookmark/bookmark_change_list.html**

```
 <tbody>
      {% for item in object_list %}
      <tr>
        <td>{{ item.title }}</td>
        <td>{{ item.url }}</td>
        <td>{{ item.owner }}</td>
        <td><a href="{% url 'bookmark:update' item.id %}">
            <i class="far fa-edit"></i> 수정</a>
        </a></td>
        <td><a href="{% url 'bookmark:delete' item.id %}" >
              <i class="fas fa-trash"></i> 삭제</a>
        </a></td>
      </tr>
      {% endfor %}
    </tbody>
  </table>

{% endblock %}
```

개발 코딩하기(삭제)

❖ 삭제 확인 인터페이스



Bookmark Delete

Are you sure you want to delete "구글" ?

개발 코딩하기(삭제)

❖ **bookmark/templastes/bookmark/bookmark_confirm_delete.html**

```
{% extends "base.html" %}

{% block title %}bookmark_confirm_delete.html{% endblock %}

{% block content %}

  <h1>Bookmark Delete</h1>
  <br>

  <form action="." method="post">{% csrf_token %}
    <p>Are you sure you want to delete "{{ object }}" ?</p>
    <input type="submit" value="Confirm" class="btn btn-danger btn-sm" />
  </form>

</div>
{% endblock %}
```