1. App user keyword with sqlite using django ORM

App name:

app\_user\_keyword\_db

1. Create project
2. Create “app\_user\_keyword\_db”

Create a new APP named “app\_user\_keyword\_db” in your project.

python manage.py startapp app\_user\_keyword\_db

1. website\_configs/settings.py

|  |  |
| --- | --- |
| |  | | --- | | INSTALLED\_APPS = [    ' app\_user\_keyword\_db',  ] | |

1. website\_configs/urls.py

website\_configs/urls.py

|  |  |
| --- | --- |
| |  | | --- | | from django.contrib import admin  from django.urls import path  from django.urls import include  urlpatterns = [  # full text search and associated keyword display using db  path('userkeyword\_db/', include('app\_user\_keyword\_db.urls')),  # admin  path('admin/', admin.site.urls),    ] | |

1. app\_user\_keyword\_db/urls.py

Create a Python file named “urls.py”.

app\_user\_keyword\_db/urls.py

|  |  |
| --- | --- |
| |  | | --- | | from django.urls import path  from . import views  app\_name="app\_user\_keyword\_db"  urlpatterns = [  path('', views.home, name='home'),  path('api\_get\_userkey\_data/', views.api\_get\_userkey\_data, name='api\_get\_userkey\_data'),  ] | |

1. model.py

資料表

|  |
| --- |
| from django.db import models  # Create your models here.  class NewsData(models.Model):  item\_id = models.CharField(max\_length=255, primary\_key=True)  date = models.DateField()  category = models.CharField(max\_length=255)  title = models.TextField()  content = models.TextField()  sentiment = models.FloatField(null=True, blank=True)  # summary = models.TextField(null=True, blank=True)  top\_key\_freq = models.TextField(null=True, blank=True) # Storing as string representation  tokens = models.TextField(null=True, blank=True)  tokens\_v2 = models.TextField(null=True, blank=True)  entities = models.TextField(null=True, blank=True)  token\_pos = models.TextField(null=True, blank=True)  link = models.URLField(null=True, blank=True)  photo\_link = models.URLField(null=True, blank=True)    class Meta:  db\_table = 'news\_data'    def \_\_str\_\_(self):  return f"{self.date}: {self.title}" |

1. admin.py

後台管理資料庫

|  |
| --- |
| from django.contrib import admin  from .models import NewsData  # Register your models here.  @admin.register(NewsData)  class NewsDataAdmin(admin.ModelAdmin):  list\_display = ('category', 'title', 'date') |

1. views.py

參看檔案

|  |
| --- |
|  |

1. home.html

參看檔案

|  |
| --- |
|  |

1. base.html

|  |
| --- |
| <a class="dropdown-item" href="{% url 'app\_user\_keyword\_db:home' %}">自訂全文檢索與關聯分析db</a> |

1. Database migration資料庫遷移

執行以下命令

python manage.py makemigrations

python mange.py migrate

or

python manage.py makemigrations app\_user\_keyword\_db

1. Create Superuser

python manage.py createsuperuser

1. Run server

python mange.py runserver

1. import data into dataset

在專案根目錄下執行以下import\_news\_data.ipynb檔案

|  |
| --- |
| import os  import sys  import pandas as pd  import argparse  from datetime import datetime  import pathlib  # 新增：將上一層目錄加入 sys.path  parent\_path = pathlib.Path().absolute().parent  sys.path.insert(0, str(parent\_path))  # Setup Django environment  os.environ.setdefault('DJANGO\_SETTINGS\_MODULE', 'website\_configs.settings')  import django  django.setup()  # 重要：設定環境變數以允許在 Jupyter 的異步環境中執行同步操作  os.environ["DJANGO\_ALLOW\_ASYNC\_UNSAFE"] = "true"  # Now we can import Django models  from app\_user\_keyword\_association.models import NewsData  # Read CSV file  csv\_file\_path = '../app\_user\_keyword/dataset/cna\_news\_200\_preprocessed.csv'  # csv\_file\_path = '../app\_user\_keyword/dataset/cna\_news\_preprocessed\_12weeks.csv'  df = pd.read\_csv(csv\_file\_path, sep='|')  # Process each row and create a NewsData object  for idx, row in df.iterrows():  try:  # Convert date string to datetime object  date\_obj = datetime.strptime(row['date'], '%Y-%m-%d').date()  # Create or update NewsData object  news\_data, created = NewsData.objects.update\_or\_create(  item\_id=row['item\_id'],  defaults={  'date': date\_obj,  'category': row['category'],  'title': row['title'],  'content': row['content'],  'sentiment': row['sentiment'],  #'summary': row['summary'],  'top\_key\_freq': row['top\_key\_freq'],  'tokens': row['tokens'],  'tokens\_v2': row['tokens\_v2'],  'entities': row['entities'],  'token\_pos': row['token\_pos'],  'link': row['link'],  'photo\_link': row['photo\_link'] if row['photo\_link'] != "" and not pd.isna(row['photo\_link']) else None,  }  )  if created:  print(f"Created new NewsData object with item\_id: {row['item\_id']}")  else:  print(f"Updated existing NewsData object with item\_id: {row['item\_id']}")  except Exception as e:  print(f"Error at row {idx}: {e}")  print(row)  # photo\_link 欄位的值可能為以下幾種情況：  # 實際有值的 URL 字串  # 空字串 ("")  # Pandas NaN 值（當 CSV 檔案中該欄位為空時）  # None 值 |

1. Administration

