

1. id integer primary key,

вот тут при создании таблицы у лайта автоинкремент автоматически для ПК стоит?
Если нет, то надо всё-же дописать

2. приорити думаю тоже лучше сделать интежером, там же будет циферка,
обозначающая важность, думаю

3. посмотри плез вот этот материал. Было бы отлично, если бы ты смог реализовать такую тему с хранением в блобе фото и файлов

кстати, можно не искать вар со сторонним хранилищем никаким.

в SQLite можно хранить все файлы в виде битовых массивов

<https://pythonru.com/biblioteki/rabota-s-izobrazhenijami-i-fajlami-v-sqlite>

4. @valery_kharkov и вот такие функции для общения с базой надо оформить, они в любом случае будут нужны

и функция поиска объекта

примерно я так накидал общую структуру:

```
def search_object(text, search_type):  
    object_info = [] # готовим выходное представление  
    sql = " " # инициализируем базу данных  
  
    # здесь поиск по базе  
    if search_type == 'номер':  
        pass # алгоритм поиска по кадастровому номеру в БД  
    elif search_type == 'адрес':  
        pass # алгоритм поиска по адресу в БД  
  
    if text in sql: # готовим нужный вариант вывода, если text в базе  
        object_info.append(0)  
        object_info.append('данные из БД')  
    else: # если в БД не находится нужный объект, выдаём код ошибки  
        object_info.append(1)  
    return object_info
```

```

1 def search_object(text, search_type):
2     object_info = [] # готовим выходное представление
3     sql = '' # инициализируем базу данных
4
5     # здесь поиск по базе
6     if search_type == 'номер':
7         pass # алгоритм поиска по кадастровому номеру в БД
8     elif search_type == 'адрес':
9         pass # алгоритм поиска по адресу в БД
10
11     if text in sql: # готовим нужный вариант вывода, если text в базе
12         object_info.append(0)
13         object_info.append('данные из БД')
14     else: # если в БД не находится нужный объект, выдаём код ошибки
15         object_info.append(1)
16     return object_info
17

```

но в общем всё равно что там внутри будет

главное, что я даю на вход некоторый текст для поиска и тип: либо по номеру искать, либо по адресу

они уже к этому моменту проверены, что не белиберда

5. а на выходе надо получить список из кода: 0 - объект найден, 1 - не найден и собственного в каком-то виде записей по объекту из БД

например в виде словаря, скажем

6. а далее нужны функции такие:

для выбранного объекта внесение изменений по полям:

изменить поле отслеживать / не отслеживать

изменить поле адреса

добавить фото

добавить документ

изменить поле статуса

изменить поле приоритета

для добавления нового объекта:

собственно функция, принимающая список или словарь всех нужных атрибутов и вносящая в БД новый объект

для сортировки:

выбрать только отслеживаемые объекты

выбрать приоритетные объекты среди отслеживаемых

выбрать недавно добавленные объекты среди отслеживаемых (пока за последний месяц, чтобы был какой-то ориентир)

отсортировать по дедлайну по возрастанию

отсортировать по номеру по возрастанию

отсортировать по адресу в алфавитном порядке

7. ах да, у нас есть фильтр по дате добавления объекта, а дата в базе только дедлайна

надо ещё дату добавления добавить

и, получается, в функции, которая будет добавлять данные в БД, прописать, чтобы туда возвращалась куррентдэйт

8. @valery_kharkov

кстати, структура такая, что база данных будет лежать по пути storage/data/(название файла БД)

можешь так же сделать и обращения сразу писать к этому адресу, чтобы потом меньше переписывать для интеграции, плз