

Python Contact Mangement System

Project Structure:

Project was organized according to <https://docs.python-guide.org/writing/structure/>

```
/ContactManager-Visakh
  /data
    contacts.sqlite
  /src
    contact_manager.py
    main.py
  /test
    test_contact_manager.py
  README.md
  requirements.txt
```

data/

- This folder contains the database.

/src

- This folder contains the main source files of the application.
 - `contact_manager.py`:
 - This module contains the `Contact` class and `ContactManger` class.
 - `main.py`:
 - This is the main entry point of the application.
 - It contains the user interface to interact with the system.

/tests

- This folder contains some tests files to ensure the methods in `Contact` and `ContactManger` classes working as expected
- `unittest` module is used for testing.

How to run the application

Install dependencies:

```
pip install -r requirements.txt
```

`src/main.py` is the main entry point of the application. it can be run using:

```
python src/main.py
```

Additional Packages used

- re: regular expressions.
- sqlite3: sqlite support.
- prettytable: print data in tabular form.
- unittest: testing functionalities at development time.
- pylint: linting.

Design

App follows a text based UI with a menu driven design.

- ContactManger class initializes the database and creates a table with required fields(Name, Phone, Email, Address)
- contact_id is autogenerated at initialization time by using AUTOINCREMENT.
- main() function displays a menu based UI to user and handles user choice.
- based on user choice, corresponding function will be executed.

regex

phone numbers

```
^(\+91\s?|0)?[6789]\d{9}$
```

It matches most of the indian phone numbers.

- can start with a single +91, optional space or a zero
- the entire above part is optional.(zero or one time)
- first digit of indian numbers are 6789, followed by any 9 digits.

eg:

- 8129667444
- +91 8129667444
- 08129667444
- 91812966744

email address

```
^\w+[\w\.-]+@[ \w]+[\w\.-]+\.\w+$
```

- username(prefix) part can have any word character, or . or -, but must start with a word character.
- prefix and domain are separated by a mandatory @ character.

- domain part starts with a single word character.
- followed by any word characters or . or -
- it will have a mandatory . followed by any word character.

Design choices:

- `with` statements are used instead of explicitly opening and closing connections.
- Some of the error handling has been moved into `main.py`, due to reasons stated below:
 - Some errors should be handled at the main function (eg: if could database creation is failed, then rest of the functions should not work).
 - Raising exceptions from the `contact_manager` module and handling them in main module gives more flexibility on how it should be handled at run time.
- In the main module, code to handle each choice cases (add contact, view contact etc.) have been moved into separate function for modularity, and better linting.

docstrings and comments are written as per the best practices mentioned in <https://realpython.com/documenting-python-code/>