Практическое задание: **Менеджер паролей**  
  
Цель: Создать базовое приложение для безопасного хранения паролей с возможностью шифрования.  
  
Задачи:  
  
1. Базовая структура данных:  
    - Создайте класс **PasswordEntry**, который будет содержать информацию о веб-сайте, логине пользователя и зашифрованном пароле.  
  
2. Ввод и вывод данных:  
    - Пользователь должен иметь возможность вводить данные о веб-сайте, логине и пароле.  
    - После ввода пароля, он должен быть *зашифрован* перед сохранением.  
    - Пользователь должен иметь возможность просматривать все сохраненные веб-сайты и логины, но не пароли.  
  
3. Шифрование:  
    - Используйте простой метод шифрования, например, XOR или другой метод на ваш выбор. (Для реального приложения следует использовать более надежные методы шифрования.)  
    - При просмотре конкретной записи пользователь может запросить дешифровку пароля, введя основной "мастер-пароль" приложения.  
  
4. Поиск:  
    - Предоставьте возможность поиска по сохраненным записям по названию веб-сайта или логину.  
  
5. Мастер-пароль:  
    - При первом запуске приложения пользователь должен установить мастер-пароль, который будет использоваться для дешифровки паролей.  
    - При каждом последующем запуске приложения, пользователь должен вводить мастер-пароль для доступа к данным.  
  
Дополнительные задачи для желающих углубиться:  
  
1. Реализуйте возможность изменения мастер-пароля.  
2. Добавьте функциональность автоматического генерирования случайного пароля при добавлении новой записи.  
3. Реализуйте возможность экспорта и импорта базы данных в зашифрованном виде.  
4. Добавьте функциональность автоматического копирования пароля в буфер обмена при выборе конкретной записи.