

# パスワード管理セキュリティプロジェクトドキュメント:

# パスワード管理セキュリティプロジェクトドキュメント:

## はじめに

このドキュメントは、Pythonを使用して基本的なパスワード管理システムを実装する方法を説明します。このシステムでは、ユーザーがアカウントを作成し、ハッシュ化されたパスワードを使用してログインすることができます。コードは、`hashlib` ライブラリを使用してハッシュ化を行い、`getpass` ライブラリを使用して安全なパスワード入力を行います。

## コード概要

### 使用ライブラリ

- `hashlib`: このライブラリは、SHA-256アルゴリズムを使用してパスワードをハッシュ化するために使用されます。
- `getpass`: このライブラリは、コンソールに表示されないようにユーザーにパスワードを入力させるために使用されます。

### データ構造

- `password_manager`: ユーザー名をキーとして、対応するハッシュ化されたパスワードを値として保存する辞書。

### 関数

`create_account()`

この関数は、ユーザーにユーザー名とパスワードを入力させます。パスワードはSHA-256を使用してハッシュ化され、`password_manager` 辞書に保存されます。

```
def create_account():
    username = input("ユーザー名を入力してください: ")
    password = getpass.getpass("パスワードを入力してください: ")
    hashed_password = hashlib.sha256(password.encode()).hexdigest()
    password_manager[username] = hashed_password
    print("アカウントが正常に作成されました")
```

## login()

この関数は、ユーザーにユーザー名とパスワードを入力させます。入力されたパスワードはハッシュ化され、`password_manager` 辞書に保存されているハッシュ化されたパスワードと比較されます。

```
def login():
    username = input("ユーザー名を入力してください: ")
    password = getpass.getpass("パスワードを入力してください: ")
    hashed_password = hashlib.sha256(password.encode()).hexdigest()
    if username in password_manager.keys() and password_manager[username] == hashed_password:
        print("ログインに成功しました")
    else:
        print("ユーザー名またはパスワードが無効です")
```

## main()

この関数は、ユーザーにアカウントを作成するか、ログインするか、または終了するかを選択させます。

```
def main():
    while True:
        choice = input("アカウントを作成するには1、ログインするには2、終了するには0を入力してください: ")
        if choice == '1':
            create_account()
        elif choice == '2':
            login()
        elif choice == '0':
            break
        else:
            print("無効な選択です")
```

# 実行方法

このスクリプトを実行するには、以下のコードを使用します。

```
if __name__ == "__main__":  
    main()
```

---