実験実施日 2024 年 10 月 31 日

コンピュータ科学実験b

ソフトウェア実験 第3,4週レポート

学生番号: 102210017
氏名: 安藤 駿
共同実験者:

1　はじめに

情報通信ネットワークは，現代の様々な計算機利用において必要不可欠のものとなっている．本実験では，Python WSGI を用いて, SQL データベースと連携した WEB アプリケーションを作成し, ネットワークプログラミングの基礎を学習する．

2　課題 6 WEB アプリケーションの作成

2.1　目的・概要

WEB サーバ上で SQL データベースと協調して動作する WEB アプリケーションを Python 言語で作成する．

2.2　作成したアプリケーションの概要と機能の説明

作成したアプリケーションの概要と機能の説明をする．

2.2.1　アプリケーションの概要

今回作成したアプリケーションは, WSGI（Web Server Gateway Interface）を使用し, sqlite3 を用いて, 気象庁の公式サイトから入手した愛知県名古屋市の天気と気温をデータベースで管理している．（気象庁, 2024）

2.2.2　アプリケーションの機能

このアプリケーションは, 日付, 気温, 天気の値を持つテーブルを作り, 気象庁の公式サイトから入手した csv ファイルからレコードを作成することで, データベースを管理している．

このアプリケーションには, データの検索, 登録, 削除の機能がある．データの検索は, 条件に適するレコードのみを表示する機能である．登録は, 入力した各フィールドの値を持つレコードを作成する機能である．削除は, 条件に適するレコードをテーブルから削除する機能である．

2.3　サービスの利用方法

作成したアプリケーションの利用方法を説明する．

2.3.1　起動方法

まず, 同じディレクトリ内に「102210017.wsgi」とディレクトリ「data」,「static」を入れる．ディレクトリ「data」内には, 入手した「data.csv」を入れる．ディレクトリ「static」内には,「default.css」を入れる．

ICE 端末に SSH 接続をし,「python3 102210017.wsgi」を実行する．その後 firefox で,「http://localhost:50017」にアクセスすると, 図 1 のようなウェブページが表示される．

図 1　localhost:50017 にアクセスした様子

2.3.2　操作方法 1 検索

検索の機能について説明する．入力欄に「晴」と入力し検索ボタンを押すと，天気が「晴」のレコードの見つかった件数とその一覧が表示される．この時の状況を図 2, 3 に示す．また，日付や気温を入力しても同様に検索できる．

気温の欄に float 型に変換できない文字列を入力すると警告文が表示される．気温の欄に「あああ」と入力し，検索ボタンを押したときの状況を図 4 に示す．

図 2　検索条件の入力

図 3　検索結果

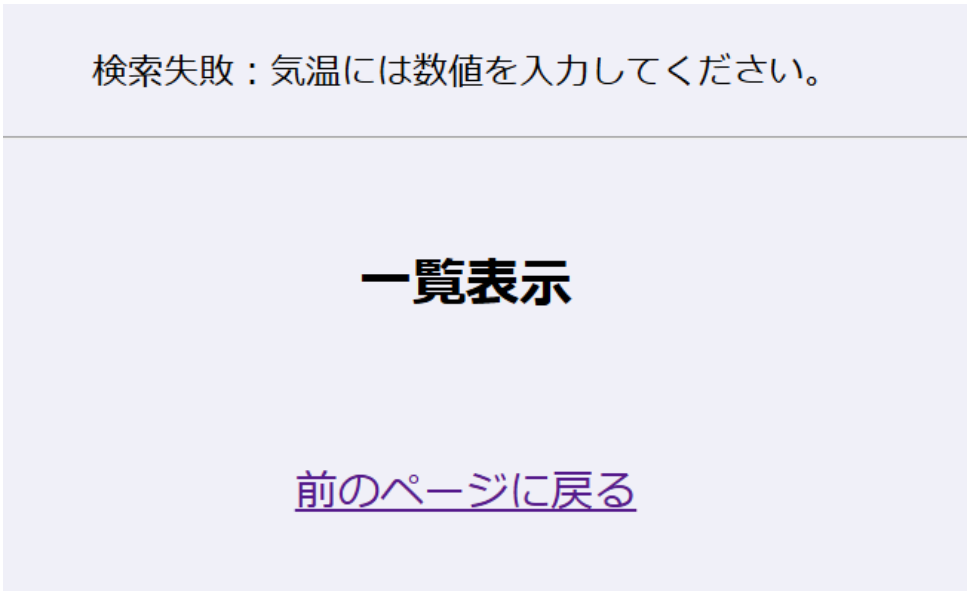

図 4　気温の入力に対する警告

2.3.3　操作方法 2 登録

登録の機能について説明する．入力欄に「2024/12/5」,「23.5」,「晴」と入力し登録ボタンを押すと，レコードの一覧が表示され，その最後尾に新しく作成したレコードが追加される．この時の状況を図 5, 6 に示す．

日付，気温，天気の入力欄のどれか一つに値を入れていないと，警告文が表示される．また，検索のときと同様に気温の欄に float 型に変換できない文字列を入力すると，警告文が表示される．

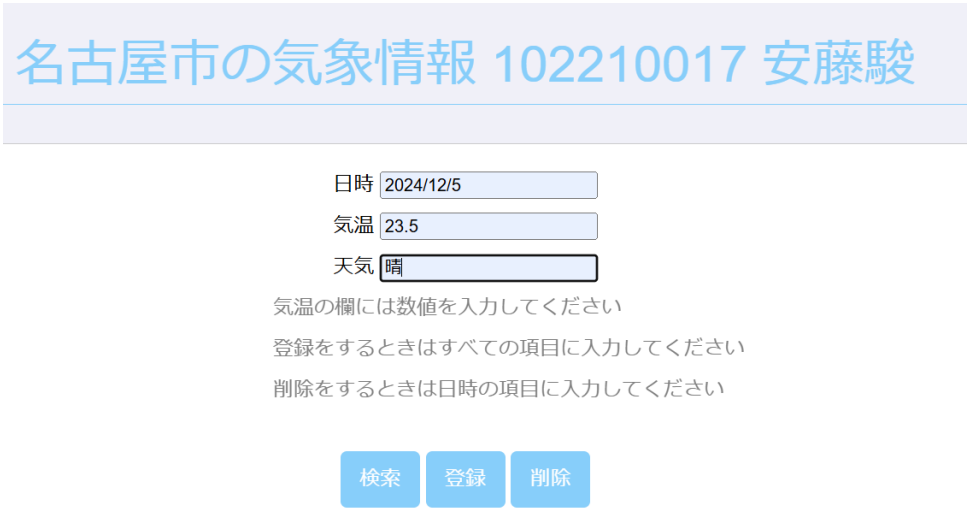

図 5　登録条件の入力

図 6　登録結果

2.3.4　操作方法 3 削除

削除の機能について説明する．日付の欄にのみ値を入力し，削除ボタンを押すとその日付のレコードがすべて削除される．検索や登録とは違い，気温，天気による削除はできない．

気温の欄に「2024/10/5」と入力し，削除ボタンを押した時の状況を図 7 に示す．

図 7　削除結果

2.4　工夫した点

気温の欄に float に変換できない値を入力したときにエラーになってしまったので, これに対応するために「try except」を用いて, 警告文を表示できるようにした.

2.5　未解決のバグ

二回目以降に「python3 102210017.wsgi 50017」を実行したときにデータベースに再び csv ファイルからデータが読み込まれ, レコードが作成されてしまうことがあった. 毎回このバグが起こるわけではく, 発生する条件がよくわからなかった.

3　調査課題 6

Web アプリケーションのセキュリティを脅かす代表的な脆弱性について調査した.

3.1　SQL インジェクション

SQL インジェクションとは, 入力フォームや URL パラメータなどから入力した「SQL 文を含む文字列」で, アプリケーションが想定していない SQL 文を実行させ, データベースの情報を不正操作するサイバー攻撃

の一種である.

SQL インジェクションの主な危険性には, 攻撃者が通常アクセスできないデータベースの内容を取得可能になること, データベース内のデータを不正に変更されるリスクがあること, 特定の SQL 文を使用することで, データベースやアプリケーション全体が破壊される可能性があること, 正しいユーザー名やパスワードを入力せずにシステムにアクセスできるようになる可能性があることなどがある.

SQL インジェクションを防ぐための手段として, プリペアードステートメントの使用がある. ユーザー入力をパラメータ化することで、不正な SQL 文の実行を防ぐことができる.

また, データベースユーザーの権限を最小限に制限し, 不必要な操作権限を与えないようにすることも重要である. アプリケーション用のデータベースユーザーには, 必要最低限の参照権限の SELECT 権限のみを与えることで, それ以外の INSERT, DELETE, UPDATE を行うことを制限させことができる.

(e-tak, 2024)

3.2 クロスサイトスクリプティング

クロスサイトスクリプティング（Cross-Site Scripting）とは Web サイトや Web アプリケーションに悪意のあるスクリプトを挿入し, ユーザーのブラウザで実行させる攻撃手法である.

クロスサイトスクリプティングの攻撃者は Web サイトのフォームやコメント欄などを利用して, 悪意あるスクリプトを埋め込む. そして, このスクリプトがユーザーのブラウザで実行されると, 何らかの情報を取得し, 攻撃者のサイトに送信される. このように不正にデータを入手することなどができる.

クロスサイトスクリプティングが発生する主な原因は入力されたデータを無処理で Web ページに表示することである.

入力データに対するエスケープ処理が適切に行われていない場合, スクリプトがそのままページに表示されユーザーのブラウザ上で実行されるリスクが発生する. ユーザーが直接入力するフィードバックやコメント欄などの項目で HTML や JavaScript のタグが無処理で表示されていると XSS 攻撃を誘発しやすくなる.

また, JavaScript や DOM 操作による動的なページが多用される場合には, クライアントサイドでの不適切な出力処理も原因となることがある. innerHTML を使用して動的にページの内容を更新する際に, 入力データをそのまま設定してしまうといったことでも XSS 攻撃のリスクが発生する.

クロスサイトスクリプティングの対策に, ユーザーからの入力データをサニタイジング（無害化）しスクリプトが実行されないようにする方法がある. <や>などの記号を HTML エンティティに変換することで, ブラウザがそれをスクリプトとして認識しないようにすることが可能である.

また, 入力バリデーションを徹底することで不正なデータが挿入されることを未然に防ぐ方法もある. 例えば英数字のみ入力をさせたいテキストに対しては特定のフォーマット（アルファベットや数値のみ）に制限するなどの対策が有効である.

Web アプリケーションファイアウォール（WAF）を使用することで、XSS 攻撃を含む不正アクセスを防ぐことも可能である. WAF はリクエスト内容を解析し特定のルールに基づいて悪意あるアクセスをブロックする機能を持っており, サーバーへの攻撃を未然に防ぐために非常に効果的である. 多くの WAF は XSS 検出の

ためのプリセットが予め設定されており特別な設定をせずとも一定の防御が可能である.
（e-tak, 2024）

参考文献

[1] 気象庁: **過去の気象データ・ダウンロード**.
https://www.data.jma.go.jp/risk/obsdl/index.php 2024.

[2] e-tak: *SQL* **インジェクションとは？ その脅威と防止策の例を解説**.
https://qiita.com/e-tak/items/702166f57e96591da5eb 2024.

[3] e-tak: **クロスサイトスクリプティング（*XSS*）とは？ わかりやすく解説**.
https://qiita.com/e-tak/items/36617b52ba7f4c922a66 2024.