

# 情報科学プロジェクト実験レポート課題

S142063 佐藤涼亮

平成 28 年 12 月 15 日

## Ajax(Asynchronous JavaScript + XML)

### 1 課題の内容

#### XMLHttpRequest による JavaScript からの HTTP の制御

#### 1.1 要点

- Ajax の技術を用いて、ページリロードをしない Web プログラムの実装
- C++ プログラムを CGI プログラムとして実装
- 入力されたデータをデータベースに格納

### 2 プログラムの説明

入力画面では、郵便番号、住所、氏名、年齢、性別の入力欄を設け、データの入力を求める。Ajax により郵便番号 7 桁から住所入力の補助を行う。郵便番号の入力欄は、7 桁を最大とし、7 桁入力された時点で CGI プログラムにより、入力された郵便番号の住所を検索し、住所の欄に結果を入れる。消去ボタンと送るボタンを設け、消去ボタンは form のリセット、送るボタンは、CGI プログラムに form を送信し、確認画面の表示をする。未入力の欄があれば入力を促すダイアログを表示する。

確認画面では、入力画面で入力されたデータをテーブル方式で表示し、確認を求める。編集ボタンと送るボタンを設け、編集ボタンはデータをそのままに入力画面に戻り、送るボタンはデータを CGI プログラムに送り、データをデータベースに格納する。その後、最終画面を表示する。

CGI プログラムは一つにまとめ、行う動作の判別のために、シリアルライズされた form の末尾に flag を追加し渡すことで判別を可能にした。

Web サーバ上のディレクトリ情報は以下のとおりである。

```
html/  
  cgi/  
    data.cgi  
    db/  
      data.db  
      tokyo.db  
    src/  
      data.cpp  
    script.js  
    data.html
```

データを格納するデータベース (data.db) のテーブル、スキーマは以下の通りである。

```
sqlite> .table  
data  
sqlite> .schema  
CREATE TABLE data  
(num text,addr text,name text,age int,gender text);
```

## 2.1 目的

XMLHttpRequest による JavaScript からの HTTP の制御

## 2.2 方法

Ajax の技術を用いて、ページのリロードを行わない Web プログラムの実装

## 2.3 結果

初期状態

### 入力画面

---

郵便番号:

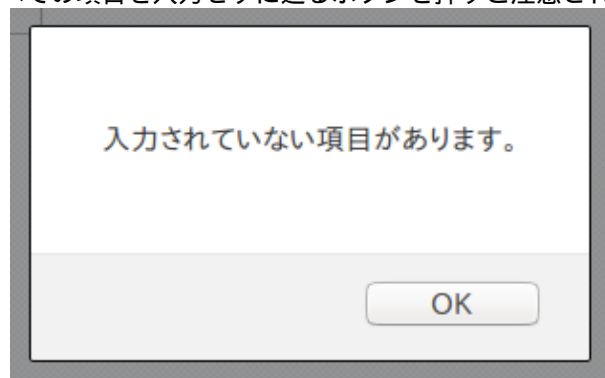
住所 :

氏名 :

年齢 :

性別 : 男: ☐ 女: ☐

すべての項目を入力せずに送るボタンを押すと注意される。



すべての項目を入力する

### 入力画面

---

郵便番号:

住所 :

氏名 :

年齢 :

性別 : 男: ☒ 女: ☐

入力画面から送るボタンを押すと確認画面が表示される

## 確認画面

	入力内容
郵便番号	1790084
住所	東京都練馬区氷川台
氏名	佐藤涼亮
年齢	21
性別	男

編集

送る

(ここで編集ボタンを押すと入力画面に戻る)

確認画面から送るボタンを押すと最終画面が表示され終了する

**ありがとうございました。**

## 2.4 考察

実行前のデータベースの状態は以下のとおりである。

```
sqlite> select * from data;  
sqlite>
```

実行後、データベースは以下のとおりになる。

```
sqlite> select * from data;  
1790084|東京都練馬区氷川台|佐藤涼亮|21|男  
sqlite>
```

これらより、データの格納は成功し、ページのリロードも行われなかったことから、期待通りの結果が得られた。

## 3 感想

前回に引き続き C++ プログラムによる CGI プログラムと HTML に加え、今回は JavaScript プログラムを加えた、XMLHttpRequest を用いた Ajax プログラムについて学習した。非同期通信により、サーバの処理待ちを必要とせず、ページ遷移することなく高速で画面の切り替えが可能であり、ユーザの動作を止めることなくスムーズな簡単なページが実現できるようになった。簡単な Web ページではあるが、普段自分が閲覧、利用しているような Web ページが、自分の手で制作することができて感動した。さらに学習をすることで、複雑な Web ページを理解できるようになりたい。

## 4 プログラム

ソースコード 1: data.html

```
1 <!doctype html>
2 <html>
3 <head><meta charset="UTF-8"><title>AJax sample</title>
4 <script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
5 <script src="/script.js"></script>
6 </head>
7 <body>
8 <div id='input' >
9 <h1 style="border-bottom: 1px solid #000000;">入力画面</h1>
10 <form id='form'>
11 <p>
12 郵便番号 :<input required class='entry' id='post' name='post' type='text' maxlength='7'
13           placeholder='ex.123-4567 1234567' pattern='^[0-9]+$/><br><br>
14 住所 :<input required class='entry' id='addr' name='addr' type='text'
15       placeholder='郵便番号から検索されます' readonly/><br><br>
16 氏名 :<input required class='entry' name='name' type='text'
17       placeholder='ex.成蹊太郎' /><br><br>
18 年齢 :<input required class='entry' name='age' type='text'
19       placeholder='半角数字で入力してください' pattern='^[0-9]+$/><br><br>
20 性別 :
21 男:<input name='gender' type='radio' value='男' />
22 女:<input name='gender' type='radio' value='女' /></p>
23 <input type="button" id="submit" value="送る" />
24 <input type="reset" id="delete" value="消去" />
25 </form>
26 </div>
27 <div id='confirm' style='display: none;'>
28 <h1 style="background-color: #F2F2E5;">確認画面</h1>
29 <table border="1" cellspacing="0" style="font-size: 17px;">
30 <tr style="background-color: #F0F7F9; font-weight: bold;">
31 <td style="padding: 5px; text-align: center;"></td>
32 <td style="padding: 5px; text-align: center;">入力内容</td>
33 </tr>
34 <tr>
35 <td style="padding: 5px; text-align: center;">郵便番号</td>
36 <td class='conf' style="padding: 5px; text-align: center;"></td>
37 </tr>
38 <tr style="background-color: #E8E3D2;">
39 <td style="padding: 5px; text-align: center;">住所</td>
40 <td class='conf' style="padding: 5px; text-align: center;"></td>
41 </tr>
42 <tr>
43 <td style="padding: 5px; text-align: center;">氏名</td>
44 <td class='conf' style="padding: 5px; text-align: center;"></td>
45 </tr>
46 <tr style="background-color: #E8E3D2;">
47 <td style="padding: 5px; text-align: center;">年齢</td>
48 <td class='conf' style="padding: 5px; text-align: center;"></td>
49 </tr>
50 <tr>
51 <td style="padding: 5px; text-align: center;">性別</td>
52 <td class='conf' style="padding: 5px; text-align: center;"></td>
53 </tr>
54 </table><br>
55 <input type="button" id="edit" value="編集" />
56 <input type="button" id="send" value="送る" />
57 </div>
58 <div id='fin' style='display: none;'>
59 <h3>ありがとうございました。</h3>
60 </div>
61 </body>
62 </html>
```

## ソースコード 2: script.js

```

1 $(function() {
2     $('#post').keyup(function(){
3         if($(this).val().length == 7){
4             $.post("/cgi/data.cgi", $('#post').serialize()+"&flag=post", function(result){
5                 $('#addr').val(result);
6             });
7         }
8     });
9
10    $('#submit').click(function(){
11        var flag=true;
12        $('.entry').each(function(){
13            if($(this).val()==""){
14                alert("入力されていない項目があります");
15                flag=false;
16                return false;
17            }
18        });
19        if(flag && $('input[name=gender]:checked').val() == null){
20            alert("入力されていない項目があります");
21            flag=false;
22        }
23        if(flag && !$('#input[name=age]').val().match(/^[0-9]+$/)){
24            alert("年齢は半角数字で入力してください");
25            flag=false;
26        }
27        if(flag){
28            var tmp = [];
29            $('.entry').each(function(){
30                tmp.push($(this).val());
31            });
32            tmp.push($('input[name=gender]:checked').val());
33
34            $('.conf').each(function(){
35                $(this).text(tmp.shift());
36            });
37            $('#input').hide();
38            $('#confirm').show();
39        }
40    });
41
42    $('#edit').click(function(){
43        $('#confirm').hide();
44        $('#input').show();
45    });
46
47    $('#send').click(function(){
48        $.post("/cgi/data.cgi", $('#form').serialize()+"&flag=fin", function(result){
49            $('#confirm').hide();
50            $('#fin').show();
51        });
52    });
53 });

```

## ソースコード 3: data.cpp

```

1 #include <iostream>
2 #include <string>
3 #include <MyDBS3.hpp>
4 #include <CGInput.hpp>
5 using namespace std;
6
7 int main()
8 {
9     CGInput fmd; // 宣言とともに入力を終える
10 }

```

```

11 if(fmd["flag"] == "post"){
12     // 結果を出力
13     cout << "Content-type:_text/plain;_charset=UTF-8\r\n\r\n";
14
15     // データベースファイルを開く
16     MyDBS d("db/tokyo.db");
17     if (!d){
18         cout << "error";
19         return 1;
20     }
21
22     string sql;
23
24     // 郵便番号での検索
25     sql = "select_kanji1,kanji2,kanji3_from_post_where_num=?";
26     if(d.prepare(sql,fmd["post"]) != SQLITE_OK){
27         cout << "error";
28         return 1;
29     }
30     string kanji1,kanji2,kanji3;
31     if(d.step(&kanji1,&kanji2,&kanji3) == SQLITE_ROW){
32         cout << kanji1 + kanji2 + kanji3;
33     }
34 }
35
36 if(fmd["flag"] == "fin"){
37
38     cout << "Content-type:_text/plain;_charset=UTF-8\r\n\r\n";
39
40     // データベースファイルを開く
41     MyDBS d("db/data.db");
42     if (!d){
43         cout << "open_db_error";
44         return 1;
45     }
46
47     string sql;
48
49     // データの挿入
50     sql = "insert_into_data_values(?,?,?,?);";
51     if(d.exec(sql,fmd["post"],fmd["addr"],fmd["name"],fmd["age"],fmd["gender"]) !=
        SQLITE_OK){
52         cout << d.error();
53         return 1;
54     }
55 }
56
57 return 0;
58 }

```