

プロジェクト実習
クラウドサービスプロジェクト
前期テキスト

2017年度

東京工科大学

コンピュータサイエンス学部

まえがき

プロジェクト実習とは、一つのテーマを年間を通じて追求することにより、高いスキルを習得できるようにすることばかりでなく、一つの分野について深く考えることを可能にすることを目的として計画されたものです。皆さんが社会に出て、たとえば、ICT 分野のエンジニアとして高い付加価値を生み出してゆくためには、特定の技術分野に関する知識が豊富にあることだけでなく、考える力を備えていることがきわめて重要になります。考える対象は、技術的なものに限りません。利用者がどのような技術を求めているのか、あるいは、どのような技術を提供すれば高い収益が得られるのかといったことも、よく考え、それを技術開発につなげてゆくことが求められます。

そこで、プロジェクト実習は、既存技術の習得を行うステップと、習得した技術を用いて新しい価値を創造することを試みるステップの双方を含んでいることを特徴としています。技術習得のための実習ステップでは、多くの場合、各個人が手を動かしたり、調査したり、考えたりすることを通じて目的を達成します。一方、新しい価値を創造するステップでは、複数の仲間と議論を行い、その内容をまとめて計画を立て、習得した技術にもとづいて実装を行い、結果を評価して改良を加えることによって目的を達成します。

このように、新しい価値創造に向かっていろいろな手段を駆使しながら目的を達成する能力こそ、皆さんが社会で求められる重要な資質になります。これを、プロジェクト実習を通じて涵養することにより、卒業課題や卒業後の活動における大きな助けになることはまちがいなく、皆さんの積極的な参加を切に希望します。

プロジェクト実習 前期

目次

実習の進め方	4
1. 全体説明	6
2. PHP 演習	9
3. Ajax 演習	14
4. SQL 演習	18
5. 先進テーマ	20

実習のすすめかた

1. 一般的注意

- (1) 実習に必要な資料、環境はすべてネットワークから入手できるので、事前に十分内容理解に努めること。
- (2) 13時15分までに必ず実験室に入り、18時15分までに実習を終了すること。

2. クラウド構築実習前期の手順

- (1) 前期実習は、講義実験棟303教室で行う。
- (2) 必ず、ノートPC、ネットワークケーブルとイヤホンを持参すること。
- (3) 欠席は厳禁とする。やむを得ず欠席した場合は、届けを出すとともに、自宅学習で補いレポートを期限までに提出すること。
- (4) 前期実習では、1テーマを1～3週で完成し、前期で少なくとも3テーマを終了すること。自由課題の時間を確保するために、その他の課題はなるべく早く完了させること。遅くとも第3週目の終了時までにはレポートを作成し提出すること。
- (5) 3テーマの終了後、各自がテーマを提案して実装を行う、自由課題の自主開発を行う。何を開発したいかを説明する提案のためのプレゼンテーションと、制作物のデモを主体とした成果発表のプレゼンテーションを行う。後期の実習の準備となるので、積極的に参加すること。

3. 実習レポート

- (1) 実習レポートを各テーマの終了時までには提出できるように、予め準備・調査をしておくこと。
- (2) 各テーマを少なくとも1週取り組み、テーマにあるすべての課題が解けたものは、3週目までに結果を指導員に報告して実習レポートの表紙を受領する。この表紙を添付し、3週目の18時15分までに所定の場所まで提出すること。提出期限を守れない場合は、単位の取得ができないことがある。
- (3) 実習指導の担当者の確認時に、レポートの内容が不備な場合には書き直しを指示することがある。
- (4) 自由課題に関しては、その外部仕様、設計、内部コードの概要、動作例を含んだ報告レポートを提出すること。

4. 実習の評価

- (1) プロジェクト実習の評価は、レポートの内容、および、出席状況などの実験の取り組み方に基づいて行う。
- (2) 少なくとも3テーマの実習レポートの提出が完了しなければ、実習の単位は取得で

きない。

- (3) 自由課題について、先進的なテーマに取り組んだものは、成績を配慮する。

1. 全体説明

1.1. 目的

本実習では、

- 1) Web アプリケーションを作成する標準的な方法について学習する、
- 2) Web アプリケーションを実際に企画、構築することにより、より大きな付加価値を持つアプリケーションを構築する方法について学習する、

ことを目的としている。

現代の Web アプリケーション プログラムは、Web ブラウザ上で動作するクライアント プログラム、サーバ側で動作するサーバ プログラム、および、データベース アクセスのためのクエリから構成されることが多い。クライアント プログラムは、JavaScript 言語と Ajax 技術を用いて、HTML に埋め込まれる形式で記述されることが多い。サーバ プログラムは、Java 言語が用いられる以外に、PHP、Python、Ruby 等のスクリプティング言語がよく用いられる。データベースへのアクセスには、SQL 言語が用いられる。したがって、本演習では、JavaScript と Ajax、PHP、SQL についてプログラミング手法を習得できるようにする。

1.2. 習得の手順

(1) Java 言語の習得

本演習では、Java 言語に関するプログラミング技術を十分に習得していることを想定している。たとえば、PHP 言語のオブジェクトの枠組みは、ほとんどそのまま Java 言語の枠組みを利用して作られている。したがって、Java 言語を習得しておけば、PHP 言語については短期間で習得することが可能である。JavaScript 言語についても、その名前から想像されるように、Java 言語の習得が不可欠である。Java 言語に関する理解が不十分である恐れがある場合は、必ず事前に Java 言語の復習を行っておくこと。

(2) 演習環境

本演習では、Web サーバ、PHP 言語処理系、データベースサーバが一体になった、XAMPP パッケージをインストールして使用することを推奨する。利用方法は、実習時に説明するが、以下に例を示す。

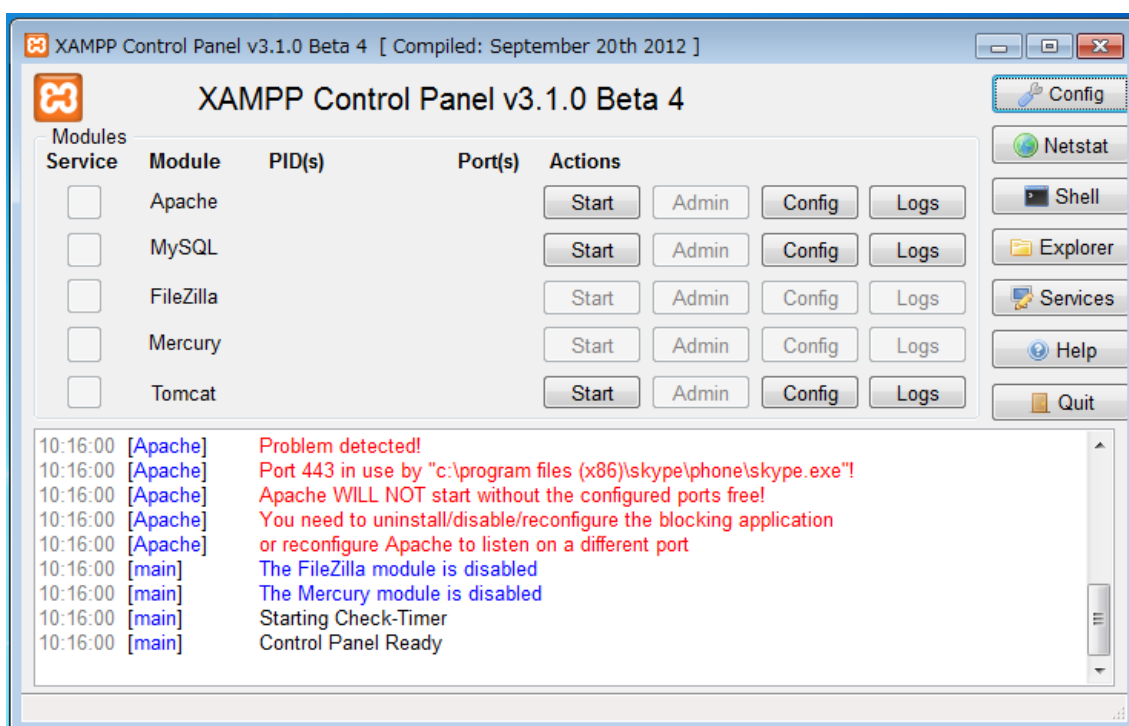
・XAMPP ダウンロード

講義ページを参照のこと。

から、XAMPP USB Lite をダウンロードして展開する。展開したファイルは、“xampp-portable”ディレクトリ下にすべて作成されるので、このディレクトリを、たとえば、USB メモリにコピーしておく、複数の計算機で利用できるのも便利である。

- ・ 設定

xampp-portable ディレクトリ (以下、インストールディレクトリ) 下に、xampp-control とよばれるアプリケーションがあるので、これを起動する。



Web サーバ (Apache) に関しては、使用するポートが他のアプリケーションと競合するので、はじめに設定が必要である。Apache の行の Config ボタンを押して、http が使用するポートと https が使用するポートを標準の 80、443 から他の値に変更する。

この後、Start ボタンを押すと、Web サーバが起動し、同時に PHP が使用可能となる。この状態では、外部からのアクセスすべてに対応するので、必要に応じてセキュリティ設定を行うことが望ましい。

- ・ 文書ディレクトリの作成

インストールディレクトリ直下に、“htdocs”ディレクトリがある。これが、Web サーバが http プロトコルで公開する、文書ディレクトリとなっている。ここに、たとえば、public ディレクトリを作成して、その下に自分が作成したファイルを置くことによって、Web プ

ブラウザから内容を見ることが可能になる。実習は、このディレクトリ下でファイルを作成することによって行う。

(4) レポート提出

前期は、①PHP 演習 ②JavaScript 演習 ③SQL 演習の3テーマ、および、④自由課題 に関して、それぞれレポートを提出すること。

レポートは、課題番号ごとに、プログラムのリスト、プログラムの説明、実行結果を記すこと。

1.3. 参考図書

以下の図書を参考とすることを強く推奨する

- 1) はじめに読みたい Ajax 入門から実践まで、たにぐちまこと、パーソナルメディア (2006)
- 2) 基礎 Web 技術、松下 温 監修、Ohmsha (2003)

他にも、以下が参考となる。

- 3) 詳細 第4版 HTML & CSS & JavaScript、大藤幹 半場方人、秀和システム (2009)
- 4) 独習 PHP、山田祥寛、SHOEISHA (2006)
- 5) 初めての JavaScript, Shelley Powers, オライリー (2007)
- 6) リレーショナルデータベース教科書、斉藤 孝、ソフト・リサーチ・センタ (2008)
- 7) Web を支える技術、山本 陽平、技術評論社 (2010)

2. PHP 演習

2.1. 目的

現代の Web アプリケーション プログラムは、Web ブラウザ上で動作するクライアントプログラム、サーバ側で動作するサーバ プログラム、および、データベース アクセスのためのクエリから構成されることが多い。ここでは、サーバ プログラムを記述するためのスクリプティング言語である、PHP について習得する。

2.2. 演習方法

(1) 参考書

以下の参考書を購入することを強く推奨する。

たにぐちまこと：はじめに読みたい Ajax 入門から実践まで、パーソナルメディア(2006)

2.3. 課題

2.3.1. 課題 PHP-1 : Hello World の表示

(1) 課題の内容

文字列

Hello World

をブラウザ上に表示するプログラム `php1.php` を作成せよ。

2.3.2. 課題 PHP-2 : 配列の表形式表示

(1) 課題の内容

配列にある住所録を表示する関数 `print_table($table)` を作成せよ。

このとき、`$table` は、住所録を保持する配列であるとする。これは、1 名分のデータを保持する連想配列を、要素として人数分含む 2 重配列構造を持つ。1 名分のデータを保持する連想配列は、インデックスとして、'名前'、'住所'、'電話番号'、'メール'を持つものとする。

(2) 実装の方法

`php2.php` は、呼び出されると、

1) 住所録を配列の形式で初期化する

初期化される住所録の内容は任意でよい。初期化を行う部分を、関数

init_table(\$table)として独立させると、プログラムが読みやすくなる。

2) 初期化された\$table を、関数 print_table を用いて表示する。

ことを行う。

(3) 実行例

実行結果の例は、以下のようなものである。

簡単な住所録の例

名前	住所	電話番号	メール
東京太郎	東京都	012-345-6789	taro@hoge.jp
工科花子	北海道	987-654-3210	hana@hoge.jp

2.3.3.課題 PHP-3： POST メソッドによる通信

(1) 課題の内容

POST メソッドを使って、住所録に項目を追加するプログラム php3.php を実現せよ。
住所録のデータは保存されず、実験のたびに初期状態にもどってよい。

(2) 実行例

初期状態の画面の例は、以下のようなものである。

簡単な住所録の例

名前	住所	電話番号	メール
東京太郎	東京都	012-345-6789	taro@hoge.jp
工科花子	北海道	987-654-3210	hana@hoge.jp

氏名	<input type="text" value="工科太郎"/>
住所	<input type="text" value="東京都"/>
電話番号	<input type="text" value="00-000-0000"/>
メールアドレス	<input type="text" value="hoge@hoge.jp"/>
<input type="button" value="追加"/>	

初期状態の画面が表示された後に、“追加” ボタンを押すと、ボタンの上のテキストボックスに入力されている氏名、住所、電話番号、メールアドレスを持つ項目が追加され、

以下のような画面が表示される。

簡単な住所録の例

名前	住所	電話番号	メール
東京太郎	東京都	012-345-6789	taro@hoge.jp
工科花子	北海道	987-654-3210	hana@hoge.jp
工科太郎	東京都	00-000-0000	hoge@hoge.jp

氏名

住所

電話番号

メールアドレス

(3) 実装方法

“追加”ボタンを押すと、POST メソッドによってデータがサーバに転送されるようにする。
php3.php プログラムは、画面を表示するための HTML の役割と、転送されたデータをサーバ側で処理する役割の双方を担っている。画面表示のための HTML のヘッダ部分とボディ部分に PHP プログラムを埋め込むことによって全体を構成する。

1) 画面の作成

“追加”ボタンを押すと、POST メソッドを用いてデータが転送されるように、HTML を作成する。そのためには、ボディ部分の最後に<form>タグを埋め込むようにすればよい。<form>タグの使用方法については、各自調べること。

2) サーバ側の処理

php3.php プログラムを作成するうえでのポイントは、POST メソッドによって送られたデータの処理方法である。php3.php プログラムは、初期画面表示の際と、“追加”ボタンが押された際の2回起動される。両者の違いは、スーパーグローバル変数\$_POST が null であるか否かで判定できる。すなわち、プログラム中で、

```
if($_POST != null) {      // POST の処理
    else {                 // 初期画面の処理
```

とすることによって区別できる。

\$_POST 変数は、連想配列で、<form>タグ中の<input>タグに付けた id と同じインデックスを持っている。詳細は、ビデオ資料を参照のこと。)

2.3.4.課題 PHP-4 : json 形式の利用

(1) 課題の内容

php3.php において、データが追加された際、住所録の内容を json 形式で表示するようにせよ。

(json 形式による配列の表示は、デバッグの手段としても有効である)

(2) 実行例

```
[{"名前":"東京太郎","住所":"東京都","電話番号":"012-345-6789","メール":"taro@hoge.jp"}, {"名前":"工科花子","住所":"北海道","電話番号":"987-654-3210","メール":"hana@hoge.jp"}]
```

簡単な住所録の例

名前	住所	電話番号	メール
東京太郎	東京都	012-345-6789	taro@hoge.jp
工科花子	北海道	987-654-3210	hana@hoge.jp

氏名

住所

電話番号

メールアドレス

(3) 実装の方法

配列から json 形式への変換には、json_encode 関数を用いよ。

2.3.5.課題 PHP-5 : ファイル入出力

(1) 課題の内容

php3.php で作成した住所録の内容を、ファイルに永続的に保存できるようにせよ。新しいデータを“追加”ボタンで追加した内容はファイルに保存され、後に画面を参照した場合には更新された画面が見えるようにせよ。

(2) 実装の方法

住所録の内容を、json 形式に変換してファイルに保存せよ。また、最初に画面表示を行う際には、保存したファイルを読み込むことを試みるようにせよ。もし、ファイルが読み込めなかったら、初期状態として、空の住所録を表示すればよい。

3. JavaScript 演習

3.1. 目的

現代の Web アプリケーションでは、高度な画面表示を可能にするために、JavaScript 言語が多く用いられている。また、サーバと柔軟に情報交換を行うために、JavaScript 言語を用いた Ajax 技術が用いられる。本演習は、JavaScript 言語、および、Ajax 技術の習得を目的とする。

3.2. 演習方法

(1) 前提知識

本演習は、HTML に関する十分な知識を必要とするが、HTML に関する体系的な説明はここでは行わない。各自図書、Web ページ等で確認すること。

例：

<http://www.tohoho-web.com/www.htm>

<http://www.tagindex.com/>

詳細 第4版 HTML & CSS & JavaScript、大藤幹 半場方人、秀和システム
(2009)

3.3. 課題

3.3.1. 課題 js-1: Hello World の表示

(1) 課題の内容

HTML を用いてブラウザ画面上にボタンを表示し、ボタンが押された際に alert() メソッドによってメッセージを表示するプログラム `js1.html` を作成せよ。

3.3.2. 課題 js-2: イベントの処理

(1) 課題の内容

HTML を用いてブラウザ画面上にボタンを表示し、ボタンが押された際にボタン表面の文字を“ボタンが押されました”に変更するプログラム `js2.html` を作成せよ。

3.3.3. 課題 js-3: DOM へのアクセス

(1) 課題の内容

HTML を用いてブラウザ画面上にテキストボックスとボタンを表示し、ボタンが押された際に `alert()` メソッドによってテキストボックスの内容を表示するプログラム `js3.html` を作成せよ。

3.3.4. 課題 js-4: DOM を用いた画面変更

(1) 課題の内容

HTML を用いてブラウザ画面上にチェックボックスとボタンを表示し、ボタンが押されるたびにチェックボックスのチェックの状態が切り替わるプログラム `js4.html` を作成せよ。

3.3.5. 課題 js-5: レイヤーの利用

(1) 課題の内容

HTML を用いてブラウザ画面上に文字列とボタンを表示し、レイヤー機能を利用してボタンが押されると文字列の表示位置が移動するプログラム `js5.html` を作成せよ。

3.3.6. 課題 js-6: GET メソッドによる情報取得

(1) 課題の内容

XMLHttpRequest 関数を用いて、ブラウザとサーバの間で GET メソッドによって通信を行うプログラム `js6_client.html` と `js6_server.php` プログラムを作成せよ。

1) `js6_client.html` プログラムは、画面上にボタンを表示し、そのボタンが押されると `js6_server.php` に向けて GET メソッドによる送信を行う。

2) `js6_server.php` プログラムは、GET メソッドのレスポンスとして文字列（任意でよい）を返す。

3) `js_client.html` は、レスポンスを受信し、その内容を `alert()` メソッドによって表示する。

3.3.7. 課題 js-7: GET メソッドによる情報送信

(1) 課題の内容

XMLHttpRequest 関数を用いて、ブラウザとサーバの間でパラメータ付きの GET メソッドによって通信を行うプログラム `js7_client.html` と `js7_server.php` プログラムを作成せよ。

- 1) js7_client.html プログラムは、画面上にボタンとテキストボックスを表示し、そのボタンが押されると js7_server.php に向けて GET メソッドによる送信を行う。このとき、URL にクエリ文字列を付けることによって、テキストボックスの内容も GET メソッドのパラメータとして送信できるようにせよ。
- 2) js7_server.php プログラムは、GET メソッドのパラメータとして送られてきた文字列に、文字列 “- - が送られました” を付け加えて、GET メソッドのレスポンスとして返す。
- 3) js7_client.html は、レスポンスを受信し、その内容を alert()メソッドによって表示する。

3.3.8. 課題 js-8: POST メソッドによる情報送信

(1) 課題の内容

XMLHttpRequest 関数を用いて、ブラウザとサーバの間でパラメータ付きの POST メソッドによって通信を行うプログラム js8_client.html と js8_server.php プログラムを作成せよ。

- 1) js8_client.html プログラムは、画面上にボタンとテキストボックスを表示し、そのボタンが押されると js8_server.php に向けて POST メソッドによる送信を行う。このとき、send メソッドの引数指定することによって、テキストボックスの内容も POST メソッドのパラメータとして送信できるようにせよ。
- 2) js8_server.php プログラムは、POST メソッドのパラメータとして送られてきた文字列に、文字列 “- - が送られました” を付け加えて、POST メソッドのレスポンスとして返す。
POST メソッドのレスポンスとして送られてきた文字列を返す。
- 3) js8_client.html は、レスポンスを受信し、その内容を alert()メソッドによって表示する。

3.3.9. 課題 js-9: テキストボックスへの表示

(1) 課題の内容

3. 3. 7 課題 js- 7 を、以下の点について変更し、js9_client.html js9_server.php を作成せよ。

変更点)

課題 js- 7 js7_client.html において、GET メソッドのレスポンスとして送られて来た文字列を、alert()メソッドで表示するのではなく、テキストボックスを書き換えて表示できるようにせよ。

3.3.10. 課題 js-10: レイヤーによる HTML の書き換え

(1) 課題の内容

3. 3. 7 課題7を、以下の点について変更し、js10_client.html js10_server.php を作成せよ。

変更点)

課題 js-7 js7_client.html において、GET メソッドのレスポンスとして送られて来た文字列を、alert()メソッドで表示するのではなく、テキストボックスの下にレイヤーを設け、それを書きかえることによって表示できるようにせよ。

3.3.11. 課題 js-11: JSON の利用

(1) 課題の内容

- 1) js11_client.html プログラムに、プッシュボタンを表示せよ。このプッシュボタンを押すと GET メソッドによって js11_server.php からレスポンスを取得できるようにせよ。
- 2) js11_server.php プログラムは、アクセスされると、json 形式の住所録をレスポンスとして返すようにせよ。住所録の内容は、空でなければ任意でよい。PHP 演習で作成したプログラム、たとえば、php5.php を参考にして、PHP で直接 HTML を作成するのではなく、json 形式の内容のみをレスポンスとして返すように作り直すこと。
- 3) js_client.html プログラムは、js11_server.php からのレスポンスを受け取ると、json 形式の文字列をオブジェクトに変換し、内容を住所録の形式で表示するようにせよ。

4. SQL 演習

4.1. 目的

本演習では、SQL 言語、および、データベース利用技術の習得を目的とする。

4.2. 演習方法

(1) 参考資料

図書

- 1) リレーショナルデータベース教科書、斉藤 孝、ソフト・リサーチ・センター (2008)

URL

- 1) http://www.techscore.com/sitemap_sql.html

4.3. 課題

4.3.1. 課題 sql-1 : 表の作成と削除

(1) 課題の内容

各自のデータベースに、表 ‘my_table’を作成するプログラム sql-create.php と、削除するプログラム sql-del.php を作成せよ。

my_table は、住所録に使用するものとする。列は、id(一次キー、INTEGER 型), name(名前、VARCHAR(32)), address(住所、VARCHAR(50)), email(メールアドレス、VARCHAR(50)) から構成されるものとする。

表が作成されたか否かは、mysql コマンドによって確認できる。mysql コマンドについても習熟すること。

4.3.2. 課題 sql-2 : データ挿入

(1) 課題の内容

my_table に、住所録の 1 名分のデータを入力し、サーバに転送する画面 sql2.html と、転送されたデータを my_table に登録するプログラム sql2.php を作成せよ。

4.3.3. 課題 sql-3 : データ一覧

(1) 課題の内容

GET メソッドに対して、`my_table` の内容を転送するプログラム `sql3.php` と、転送された内容を表示する画面 `sql3.html` を作成せよ。内容を転送するデータの形式に関しては、各自工夫せよ。

4.3.4. 課題 sql-4 : データ選択

(1) 課題の内容

`my_table` の内容を、名前によって検索するプログラム `sql4.php` と、検索結果を表形式で表示するプログラム `sql4.html` を作成せよ。検索結果を転送するデータの形式に関しては、各自工夫せよ。

4.3.5. 課題 sql-5 : データ変更

(1) 課題の内容

`sql-4` で作成した検索機能に加えて、検索結果の画面を書き換えて、“変更” ボタンを押した場合には、データベースの変更が行えるプログラムを作成せよ。画面表示機能に関しては、`sql4.html` に機能を追加し、`sql5.html` を作成せよ。データベースの変更を行う PHP プログラムとして、新たに、`sql5.php` を作成せよ。

4.3.6. 課題 sql-6 : データ削除

(1) 課題の内容

`sql-4` で作成した検索機能に加えて、“削除” ボタンを押した場合には、`my_table` の内容から対応する行を削除する機能を持つプログラムを作成せよ。画面表示機能に関しては、`sql5.html` にさらに機能を追加し、`sql6.html` を作成せよ。データベースの変更を行う PHP プログラムとして、あらたに、`sql6.php` を作成せよ。

4.3.7. 課題 sql-7 : 統合されたシステムの構築

(1) 課題の内容

以上の課題で作成した機能を統合して、住所録を作成せよ。画面から、住所録全体の作成、削除、特定行の検索、検索結果の行の変更、削除等が行えるように、住所録画面 `sql7.html` を作成せよ。

(2) 追加課題 (実施は任意)

名前以外の情報で検索が行えるようにする等、各自機能を工夫して追加せよ。

自由課題

4.4. 目的

これまでに、PHP、Ajax、SQL について実習を行ってきた。ここでは、これら基礎知識を用いて、実際に利用可能な、より大規模なアプリケーションの作成を試みる。

4.5. 課題

自分で、作成したいプログラムを計画してそれを作成してよい。どのようなプログラムを作成するか、事前に 2，3 ページの企画書を作成し、プレゼンテーションを行うこと。終了時に、デモ発表会を企画する。