



G's ACADEMY
TOKYO

PHP MySQL

関数化／更新／削除

授業前にApache,MySQLの起動確認



アジェンダ

- 演習：関数化（function）
- 演習：更新（UPDATE文）
- 演習：削除（DELETE文）
- 次回課題

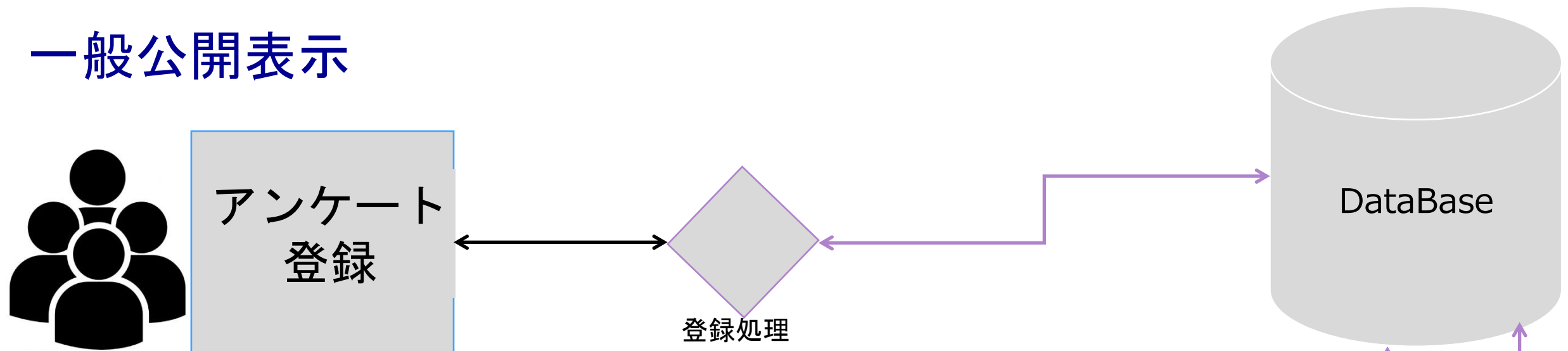
今日の授業準備

- 配布サンプル
htdocs/以下に”php03”フォルダを置く
- ” gs_db3 ”データベースを作成する
配布サンプル” SQL ”フォルダ内の
ファイルを1つを「インポート」
 - gs_an_table.sql
- XAMPP使用の人は → insert.phpのDBパスワード設定

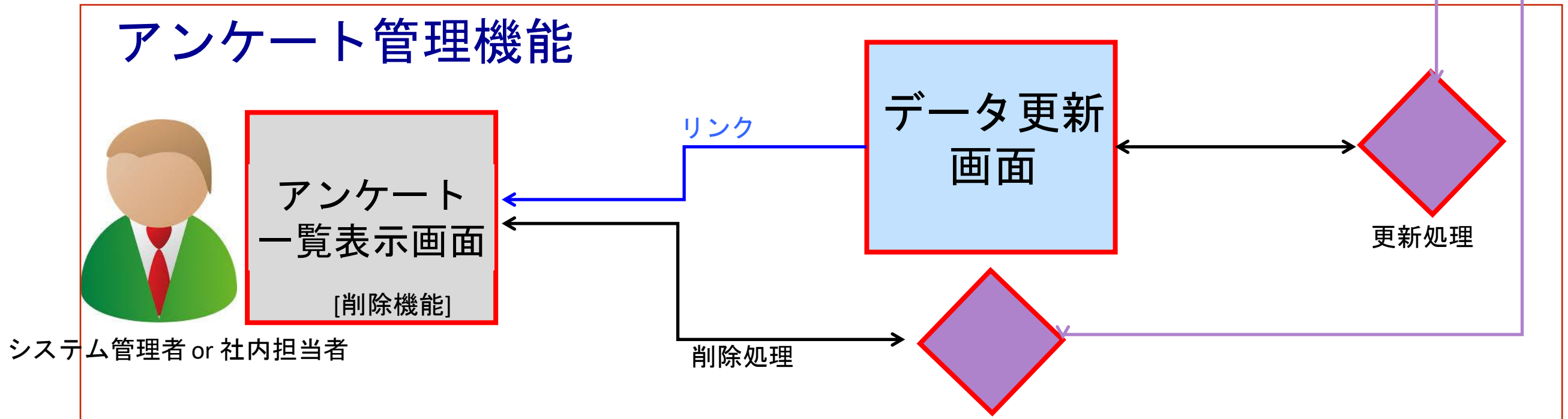
今回作成するシステム解説

アンケートシステム

一般公開表示



アンケート管理機能



関数化

insert.phpから 3 つの処理を関数化



function.php

function.phpへ共通処理を移行

```
1 <?php
2 //XSS対応（echoする場所で使用！それ以外はNG）
3 function h($str){
4     return htmlspecialchars($str, ENT_QUOTES);
5 }
6
7 //DB接続：db_conn()
8
9
10
11
12 //SQLエラー：sql_error($stmt)
13
14
15
16
17 //リダイレクト：redirect($file_name)
18
19
20
```

PHPとDB接続

データ更新

更新の流れ

① リンク作成 select.php

localhost/gs_code/php/select.php

データ登録

| | | |
|---------------------|---|---------|
| 2016-08-09 14:31:55 | 1 | やまざき |
| 2016-08-10 14:53:32 | 2 | ジーズ三郎 |
| 2016-08-10 14:54:18 | 4 | ジーズ五郎 |
| 2016-08-17 00:00:00 | 5 | ジーズ六郎 |
| 2016-08-17 00:00:00 | 6 | ジーズ七郎 |
| 2016-08-17 00:00:00 | 7 | ジーズ八郎 |
| 2016-08-17 15:16:58 | 8 | テスト 9 9 |
| 2016-08-17 15:21:47 | 9 | test100 |

[detail.php?id=**](#)

② 更新ページ作成 detail.php

`$_GET["id"];`で1レコード取得

データ一覧

フリーアンケート

名前:

Email:

③ 更新処理作成 update.php

アンケート更新処理

[select.php](#)

データ登録

| | | |
|---------------------|---|--------|
| 2016-08-09 14:31:55 | 1 | ジーズタロウ |
| 2016-08-10 14:53:32 | 2 | ジーズ三郎 |
| 2016-08-10 14:54:18 | 4 | ジーズ五郎 |
| 2016-08-17 00:00:00 | 5 | ジーズ六郎 |
| 2016-08-17 00:00:00 | 6 | ジーズ七郎 |

①リンク作成:「更新ページへのリンク作成」

◇サンプルコード: [select.php](#)

```
$view = "";  
if($status==false) {  
    $error = $stmt->errorInfo(); //Errorがある場合  
    exit("ErrorQuery:".$error[2]); //配列index[2]にエラーコメントあり  
} else {  
    while( $result = $stmt->fetch(PDO::FETCH_ASSOC)){  
        //GETデータ送信リンク作成  
        $view .= '<p>';  
        $view .= '<a href="detail.php?id='. $result["id"].">';  
        $view .= $result["indate"] . " : " . $result["name"] ;  
        $view .= '</a>';  
        $view .= '</p>';  
    }  
}
```

②更新ページ作成: id値をGET→ id値をもとにデータ抽出

◇サンプルコード: [detail.php](#)

```
<?php
```

```
//GET送信されたidを取得 (URLの後ろについてるデータ)
```

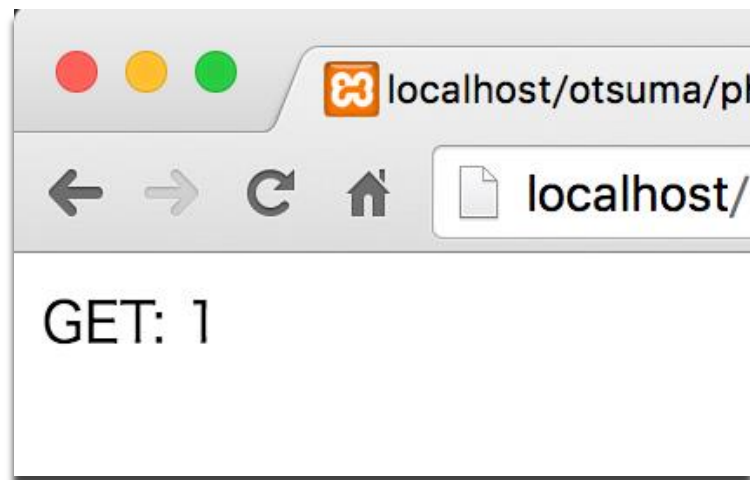
```
$id = $_GET["id"];
```

```
echo "GET: ". $id;
```

```
?>
```

表示結果: detail.php

(注意:必ず [select.php](#) のリンクをクリックして表示!)



ポイント: **\$row** = \$stmt->fetch();

③更新処理作成: データ更新処理 → 処理完了後にデーター一覧ページへ遷移

◇サンプルコード: 更新処理 (update.php)

//データ登録SQL作成

```
$update = $pdo->prepare("UPDATE テーブル名 SET name=:name,  
email=:email, naiyou=:naiyou, age=:age WHERE id=:id");
```

//bindValueを通して危険な文字を無効化!

```
$update ->bindValue(':name', $name, PDO::PARAM_STR);
```

```
$update ->bindValue(':email', $email, PDO::PARAM_STR);
```

```
$update ->bindValue(':naiyou', $naiyou, PDO::PARAM_STR);
```

```
$update ->bindValue(':age', $age, PDO::PARAM_INT);
```

```
$update ->bindValue(':id', $id, PDO::PARAM_INT);
```

//SQL実行

```
$status = $update->execute();
```

//次のページへリダイレクト

```
header("Location: select.php");
```

PHPとDB接続

データ削除

削除の流れ

select.php

← → ↻ 🏠 ⓘ localhost/gs_code/php/select.php

データ登録

| | | | |
|---------------------|---|---------|------|
| 2016-08-09 14:31:55 | 1 | やまざき | [削除] |
| 2016-08-10 14:53:32 | 2 | ジーズ三郎 | [削除] |
| 2016-08-10 14:54:18 | 4 | ジーズ五郎 | [削除] |
| 2016-08-17 00:00:00 | 5 | ジーズ六郎 | [削除] |
| 2016-08-17 00:00:00 | 6 | ジーズ七郎 | [削除] |
| 2016-08-17 00:00:00 | 7 | ジーズ八郎 | [削除] |
| 2016-08-17 15:16:58 | 8 | テスト 9 9 | [削除] |
| 2016-08-17 15:21:47 | 9 | test100 | [削除] |

[delete.php?id=**](#)

`$_GET["id"];`でデータ取得

delete.php

アンケート削除処理

[select.php](#)

データ登録

| | | | |
|---------------------|---|-------|--|
| | | | |
| 2016-08-10 14:53:32 | 2 | ジーズ三郎 | |
| 2016-08-10 14:54:18 | 4 | ジーズ五郎 | |
| 2016-08-17 00:00:00 | 5 | ジーズ六郎 | |
| 2016-08-17 00:00:00 | 6 | ジーズ七郎 | |

PHPとデータベースの接続 (リンク作成＞データ削除)

◇サンプルコード：[select.php](#)

// 赤文字のリンクを追加する！！

```
while( $result = $stmt->fetch(PDO::FETCH_ASSOC)){  
    //GETデータ送信リンク作成  
    $view .= '<p>';  
    $view .= '<a href="detail.php?id=' . $result["id"].'">';  
    $view .= $result["indate"] . " : " . $result["name"] ;  
    $view .= '</a>';  
    $view .= '  ';  
    $view .= '<a href="delete.php?id='.$result["id"].'">';  
    $view .= ' [削除] ';  
    $view .= '</a>';  
    $view .= '</p>';  
}
```

PHPとデータベースの接続 (データ削除)

◇サンプルコード： 削除 (delete.php)

//GET値取得

```
$id = $_GET["id"];
```

//データ登録SQL作成

```
$update = $pdo->prepare("DELETE FROM テーブル名 WHERE id=:id");
```

//WHERE id=変更するidを指定

```
$update->bindValue(':id', $id, PDO::PARAM_INT);
```

//SQL実行

```
$status = $update->execute();
```


課題 1

課題テーブル仕様

本をブックマークするDBを造

- **DB名:** **前回課題と同DB名**
- **Table名:** **gs_bm_table**
- **項目名:**
 1. ユニーク値 (int 12 , PRIMARY, AutoIncrement)
 2. 書籍名 (varChar 64)
 3. 書籍URL (text)
 4. 書籍コメント (text)
 5. 登録日時 (datetime)

前回課題

index.php(登録)

select.php(更新)

授業で作成したアンケートを模倣して作ること。

フィールド名は
自分で考えて実際にテーブルを作成しましょう！

【今回課題】ブックマークアプリの更新画面 & 処理作成

◇仕様：

- アンケート画面を参考に「ブックマーク”更新”画面」
bm_update_view.phpを作成
(アンケートのHTMLをコピーして変更使用してもOK)
- bm_update.php ファイルを作成し「更新処理」を作成。
- 「更新処理」完了後に、自動で bm_list_view.php に戻る
※header関数を利用
- 「削除」も実装。

授業で作成したアンケート
を模倣して作ること。
処理ほぼ一緒。

スポーツと一緒に「トレーニング」手と頭を動かす！
「作る、動作確認」を繰り返すことが習得の早道！

課題 2

～ブックマークアプリのUSER管理画面を作る～

- ・ 管理ユーザーを追加 (登録画面、登録処理)
- ・ 管理ユーザーの一覧表示 (一覧画面、[更新/削除リンク](#))
- ・ 管理ユーザーを変更 ([更新画面](#)、[更新処理](#))
- ・ 管理ユーザーを削除 ([削除処理](#))

ユーザーテーブルの仕様は20ページを参照。

※認証機能は来週の授業でやります。

管理ユーザー：管理画面

◇管理ユーザーテーブルを作成

- **DB名:** `gs_db`
- **Table名:** `gs_user_table`
- **Field名:**
 - `id:` `int(12)` `AUTO INCREMENT PRIMARY KEY`
 - `name :` `var_char(64)`
 - `lid:` `var_char(128)`
 - `lpw:` `var_char(64)`
 - `kanri_flg:` `int(1)` ※0=管理者, 1=スーパー管理者
 - `life_flg:` `int(1)` ※0=退社, 1=入社

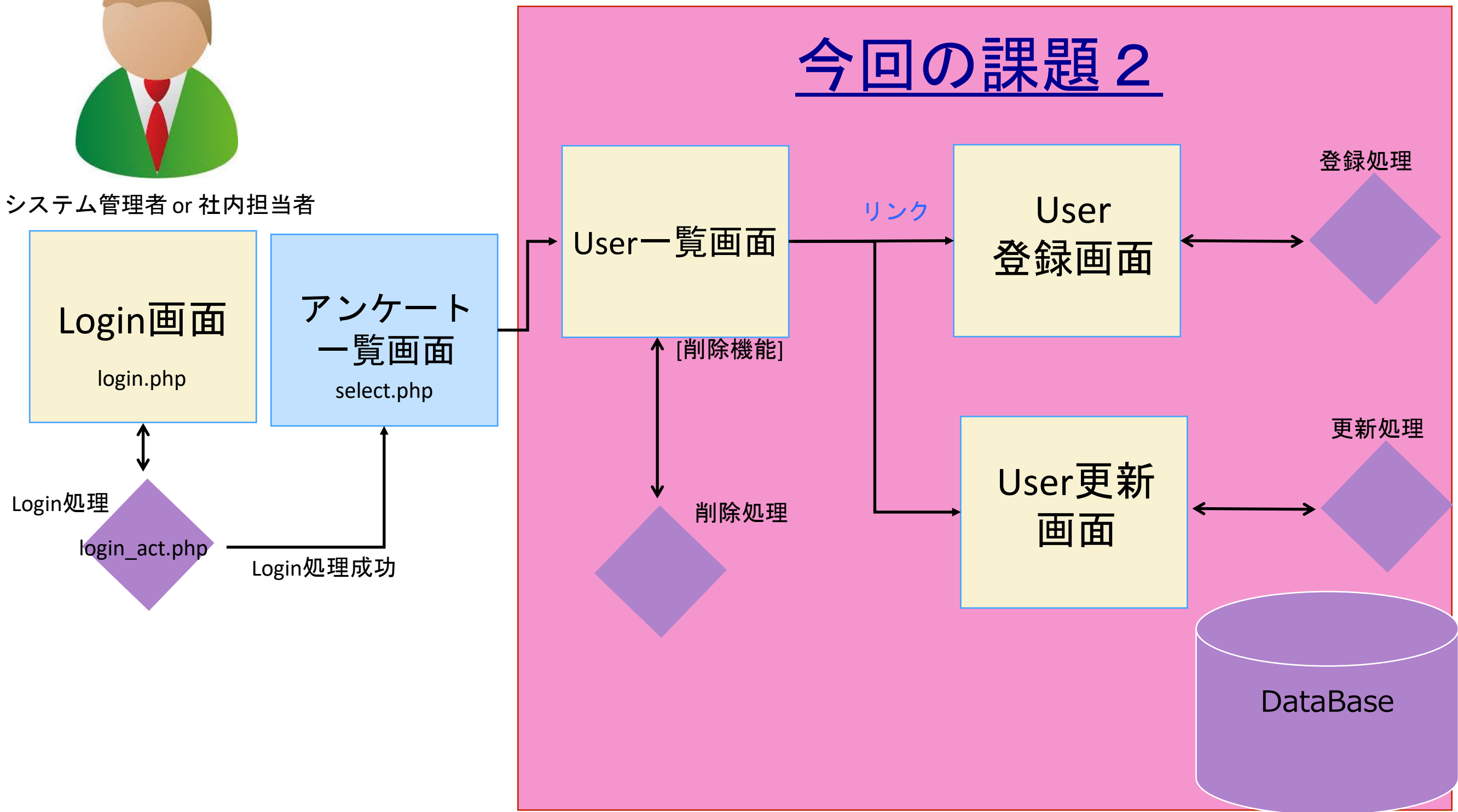
※Fieldの右にあるのは、データ型(Type)です。

<http://mysql.akarukutanoshiku.com/category5/entry21.html>

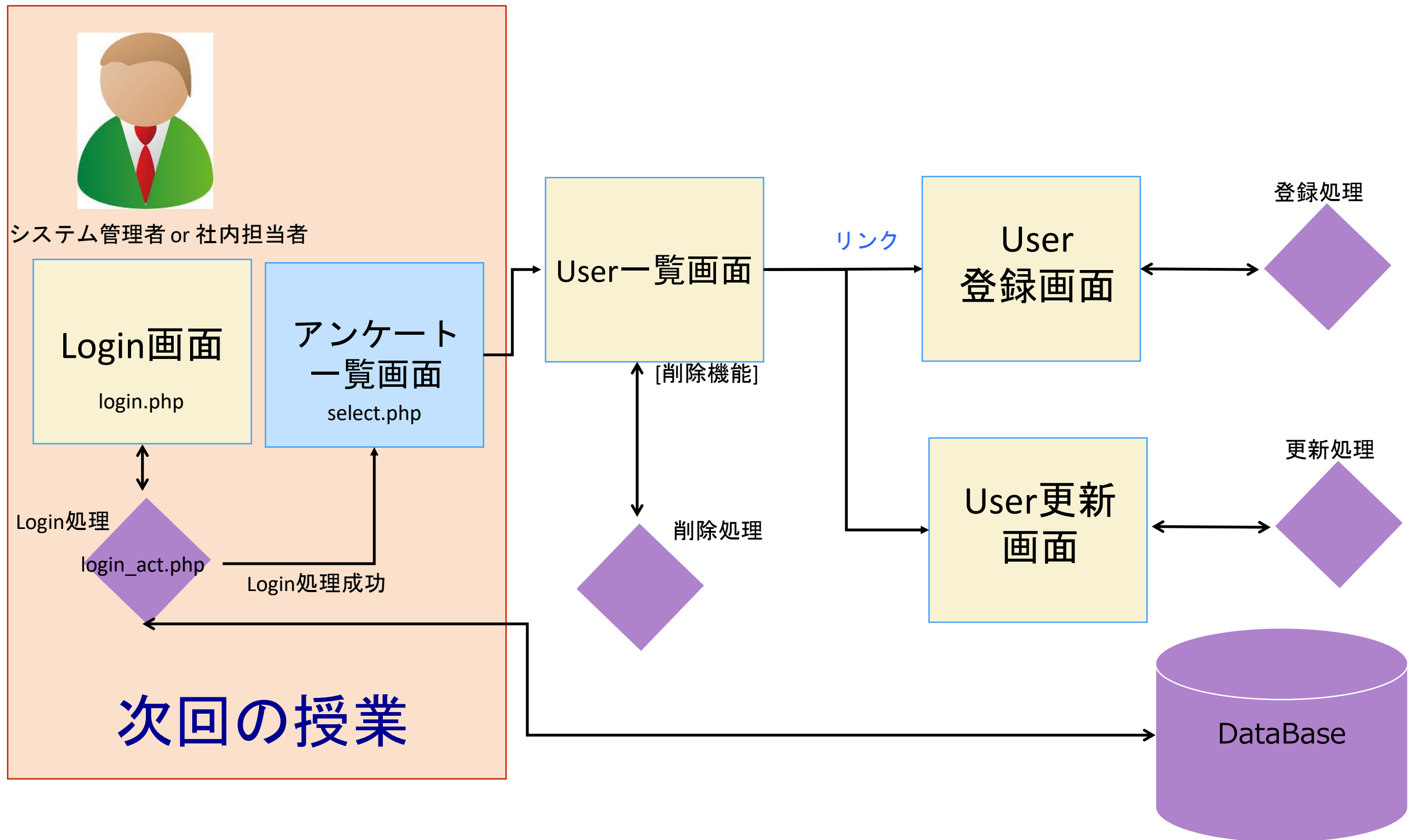
2. システム管理者機能 (課題2)



システム管理者 or 社内担当者



3. システム管理者機能 (次回の授業)



管理機能完成図

管理システム完成図

