



TOKYO JOHO UNIVERSITY

東京情報大学

Since 1988

2017年度

# Webシステムプログラミングa

PHPによるDB操作(2)

# 講義内容

- (前回) 提出課題の解答例
- PHPの応用 (例外処理とクラス)
- PHPからMySQLへのアクセス
- PHPからMySQLへのアクセス (+ ユーザからの入力)

# 提出課題

- 課題1 (提出課題): 図書館におけるbook(書籍)テーブルから以下のレコードを検索してみましょう.
  - ◆ レコードの検索条件: 価格が1100(円)以上,かつ1200(円)未満
  - ◆ 取得フィールド: すべて

```
mysql> SELECT * FROM book WHERE price>=1100 AND price<1200;
```

| isbn        | title            | author            | publisher | pub_date   | price |
|-------------|------------------|-------------------|-----------|------------|-------|
| 4000103563  | 岩波講座ソフトウェア科学(16) |                   | 岩波書店      | 1989-02-01 | 1100  |
| 4000260219  | 現代政治分析           | ロバート・アラン・ダール 高島通敏 | 岩波書店      | 1999-03-01 | 1120  |
| 4000268503  | ジェンダ - 化する社会     | 姫岡とし子             | 岩波書店      | 2004-03-01 | 1130  |
| ... (略) ... |                  |                   |           |            |       |
| 4915787214  | マ - ケティング        | 上田隆穂 江原淳          | 新世社(渋谷区)  | 1992-05-01 | 1160  |
| 4938661845  | 社会学者のメチエ         | ピエ - ル・ブルデュ 田原音和  | 藤原書店      | 1994-01-01 | 1100  |

```
40 rows in set (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

# 解答

# 提出課題

- 課題2 (提出課題): 図書館におけるbook(書籍)テーブルから以下のレコードを検索してみましょう。
  - ◆ レコードの検索条件: 書名が"入門"で終わり,かつ発行日が2000年1月1日以降
  - ◆ 取得フィールド: 書名・発行日・価格
  - ◆ レコード並替順: 価格が安い順

## 解答

```
mysql> SELECT title, pub_date, price FROM book WHERE title LIKE "%入門"
-> AND pub_date >="2000/1/1" ORDER BY price ASC;
```

| title                     | pub_date   | price |
|---------------------------|------------|-------|
| ケ - スで学ぶ価格戦略・入門           | 2003-05-01 | 480   |
| 環境統計学入門                   | 2003-04-01 | 530   |
| Mac OS 10 ユ - ザのためのUNIX入門 | 2003-01-01 | 540   |
| ・・・(略)・・・                 |            |       |
| IT革命で変わる新しいマ - ケティング入門    | 2000-10-01 | 4450  |
| 暗号技術入門                    | 2003-09-01 | 4460  |

```
53 rows in set (0.00 sec)
Query OK, 53 rows affected (0.00 sec)
```

# 提出課題

- 課題3 (提出課題) : 図書館における利用者を管理する「user(利用者)」テーブルを作成し、そのテーブル情報を表示してみましょう。
  - ◆ ユーザ j????? でデータベースに接続して作成
  - ◆ データベース j?????\_db の中にテーブルを作成

| テーブル名 | user        |         |         |         |       |          |
|-------|-------------|---------|---------|---------|-------|----------|
| 属性名   | user_id     | name    | address | tel     | birth | reg_date |
| 説明    | ユーザ番号       | 氏名      | 住所      | 電話番号    | 生年月日  | 登録年月日    |
| データ型  | VARCHAR     | VARCHAR | VARCHAR | VARCHAR | DATE  | DATE     |
| データ長  | 15          | 255     | 255     | 15      |       |          |
| 初期値   |             |         |         |         |       |          |
| キー種   | 主キー         |         |         |         |       |          |
| 制約    |             |         |         |         |       |          |
|       | PRIMARY KEY |         |         |         |       |          |

# 提出課題

- 課題3 (提出課題) : 図書館における利用者を管理する「user(利用者)」テーブルを作成し、そのテーブル情報を表示してみましょう。
  - ◆ ユーザ j????? でデータベースに接続して作成
  - ◆ データベース j?????\_db の中にテーブルを作成

解答

```
mysql> CREATE TABLE user (user_id VARCHAR(15) PRIMARY KEY,  
-> name VARCHAR(255), address VARCHAR(255), tel varchar(15),  
-> birth DATE, reg_date DATE);  
Query OK, 0 rows affected (0.00 sec)  
mysql> DESCRIBE user;
```

| Field    | Type         | Null | Key | Default | Extra |
|----------|--------------|------|-----|---------|-------|
| user_id  | varchar(15)  | NO   | PRI | NULL    |       |
| name     | varchar(255) | YES  |     | NULL    |       |
| address  | varchar(255) | YES  |     | NULL    |       |
| tel      | varchar(15)  | YES  |     | NULL    |       |
| birth    | date         | YES  |     | NULL    |       |
| reg_date | date         | YES  |     | NULL    |       |

```
6 rows in set (0.00 sec)
```

実行例

# 提出課題

- 課題4 (提出課題): 「user(利用者)」テーブルにデータをインポートし, そのデータ(レコード)を検索(表示)してみましょう.
  - ◆ インポートするファイルは以下からダウンロード.  
<http://www.edu.tuis.ac.jp/~fujiwara/2015/user.csv>
  - ◆ 検索はすべてのレコード, すべてのフィールド.

## 解答

```
mysql> LOAD DATA INFILE "Z:¥¥user.csv" INTO TABLE user
-> FIELDS TERMINATED BY ",";
Query OK, 20 rows affected (0.03 sec)
Records: 20 Deleted: 0 Skipped: 0 Warnings: 0
```

```
mysql> SELECT * FROM user;
```

| user_id      | name   | address         | tel          | birth      | reg_date   |
|--------------|--------|-----------------|--------------|------------|------------|
| U000001      | 谷岡 春樹  | 東京都江戸川区一之江X-X   | 03-3652-XXXX | 1980-04-02 | 2014-04-02 |
| U000002      | 塩見 ひとみ | 埼玉県川口市栄町X-X     | 048-251-XXXX | 1975-11-18 | 2014-04-02 |
| U000003      | 植木 仁   | 埼玉県さいたま市桜区中島X-X | 048-851-XXXX | 1990-06-08 | 2014-04-04 |
| ... (省略) ... |        |                 |              |            |            |

20 rows in set (0.00 sec)

# 提出課題

- 課題5 (提出課題): 図書館におけるuser(利用者)テーブルから以下のレコードを検索してみましょう.
  - ◆ レコードの検索条件: 誕生日が1980年1月1日より前の利用者の中で、「千葉県」もしくは「埼玉県」に在住
  - ◆ 取得フィールド: すべて

```
mysql> SELECT * FROM user WHERE birth<"1980/1/1" AND (address LIKE "千葉県%" OR address LIKE "埼玉県%");
```

```
+-----+-----+-----+-----+-----+-----+
| user_id | name      | address                | tel      | birth    | reg_date |
+-----+-----+-----+-----+-----+-----+
| U000002 | 塩見 ひとみ | 埼玉県川口市栄町X-X   | 048-251-XXXX | 1975-11-18 | 2014-04-02 |
| U000004 | 下村 沙耶   | 千葉県船橋市湊町X-X   | 047-436-XXXX | 1977-12-22 | 2014-04-05 |
| U000007 | 三枝 宏     | 千葉縣市川市鬼越X-X   | 047-334-XXXX | 1969-06-13 | 2014-04-07 |
| U000019 | 森久保 明   | 埼玉県戸田市戸田公園X-X | 048-422-XXXX | 1970-04-30 | 2014-04-15 |
| U000020 | 鈴木 和彦   | 千葉県千葉市若葉区御成台X-X | 043-236-XXXX | 1971-09-02 | 2014-04-15 |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

## 解答



# PHPの応用(1)

## ■ 例外処理

➡ 実行時に発生する可能性がある例外(エラー)に対しての処理  
(tryブロックで例外が発生した場合には、以降の処理を止めて、catchブロックに飛ぶ)

```
try {  
    ...  
    例外発生の可能性のある処理  
    ...  
} catch( 例外クラス名 変数 ) {  
    ...  
    例外発生時の処理  
    ...  
}
```

例外発生は以下の2つ.

- 「throw new 例外クラス();」のように自分で例外クラス生成して throw した場合.
- 例外発生の可能性のある関数やメソッド(インスタンス生成も含む)を呼び出し、その処理の中で例外が発生した場合.

例外発生の可能性のある処理は(基本的に必ず) try ~ catch で囲む (= 例外対応) しなければならない(マニュアルをチェック).

# PHPの応用(2)

## ■ クラス

- データ(=プロパティ)と処理(=メソッド)をまとめたもの(他のプログラミング言語と同じ)
- クラスは雛形(=ただの定義)であり,そこから実体であるインスタンス(オブジェクトとも呼ばれる)を作成(**new**)して使用する(ひとつのクラスから複数のインスタンスを作成できる).
- PHP標準で組み込まれているクラスや外部ライブラリのクラスを利用したり,自分でクラスを定義することもできる.
- プロパティおよびメソッドにアクセスするには **->** を使用.
- インスタンスの生成やメソッドで例外発生のある場合は(基本的に必ず) **try ~ catch** で囲む(=例外対応)しなければならない(マニュアルをチェック).

```
$hoge = new Foo(); // クラスFooからインスタンスを生成(引数なし)
```

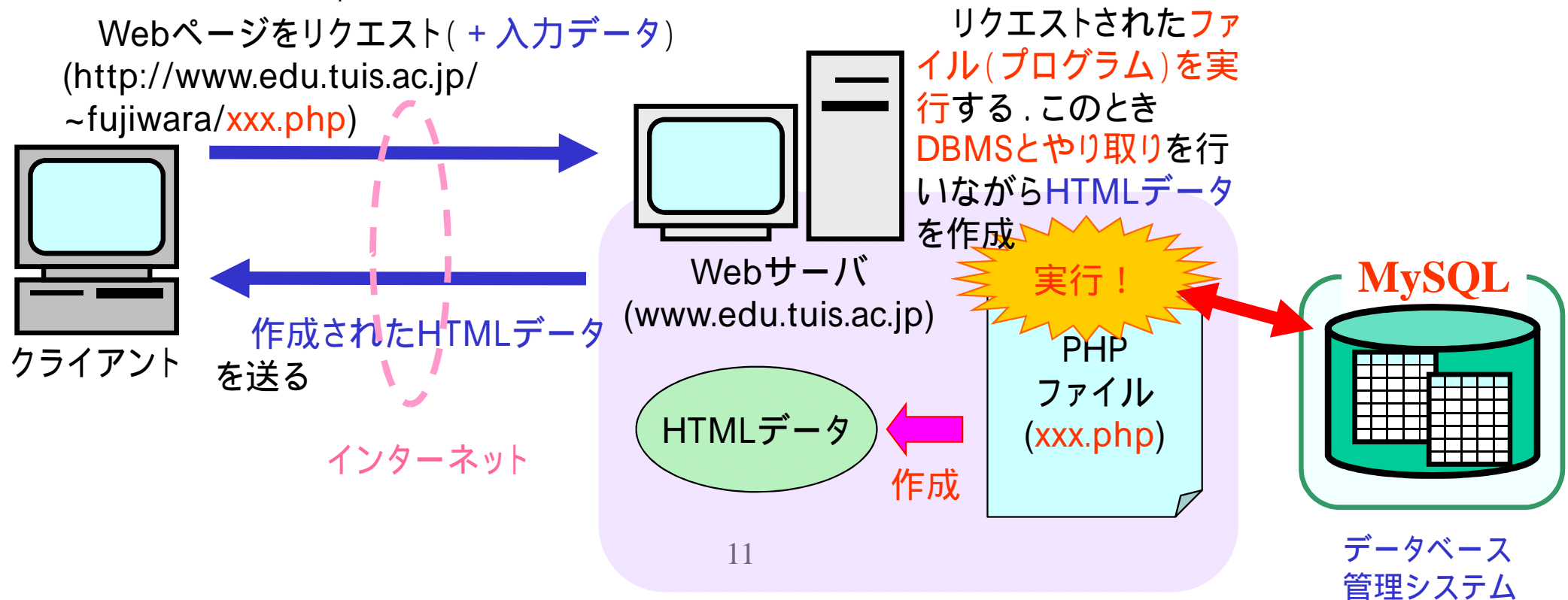
```
$hoge->print(); // メソッドprint()の呼び出し(引数なし)
```

```
$hoge->val = 123; // プロパティvalへの代入
```

# 動的なWebページ + DB

## ■ 動的なWebページ( + データベース)

- ➡ クライアント(Webブラウザ)のリクエスト(+ 入力データ)に対して, その指定されたプログラムを実行する. そのときにデータベース管理システムとやりとりし, 必要なデータの取得や更新を行いながらHTMLデータを作成, そのデータを送る



# PHPからMySQLへのアクセス(1)

## ■ データベース管理システムへの接続

```
$db = new PDO("mysql:host=IPアドレスまたはホスト名:ポート番号;dbname=選択するデータベース名;charset=文字コード",  
             "DBMSでのユーザ名", "パスワード");
```

PHPでDBへアクセスするには PDO (PHP Data Object) クラスを使用する (PDOはDBへのさまざまな処理をまとめたクラス. MySQL以外にも対応).

以降変数\$db (=PDOオブジェクト) を通してデータベースにアクセスできる.

例外 PDOException クラスが発生する可能性あり (以降のPDOに対する各メソッドも).

もちろん変数\$dbは違う名前でもよい.

```
<?php  
...  
try {  
    $db = new PDO('mysql:host=localhost:63306;dbname=j01234_db;charset=utf8', 'j01234', 'hoge');  
} catch (PDOException $e) {  
    exit('データベース接続失敗: '.$e->getMessage());  
}  
...  
?>
```

関数exit()はそれ以降のスキ립トの処理を停止する関数 (引数は表示メッセージ).

# PHPからMySQLへのアクセス(2)

## ■ SQLオブジェクトの準備

```
$sql = $db->prepare("SQL文");
```

変数\$dbはPDOオブジェクト  
変数\$sqlは違う名前でもよい

## ■ SQLの実行

```
$sql->execute();
```

変数\$sqlはSQLオブジェクト

```
<?php
. . .
$sql = $db->prepare("SELECT * FROM book;");
$sql->execute();
. . .
?>
```

# PHPからMySQLへのアクセス(3)

## ■ SQLの実行結果からレコードを取得

```
$row = $sql->fetch();
```

- 変数`$sql`はSQLオブジェクト。  
`fetch()`メソッドを呼び出すたびに実行結果から1レコードずつ取得(もうレコードが無い場合は`FALSE`を返す)。
- 変数`$row`に1行分のレコードが連想配列として代入される(キーはフィールド名)。  
変数`$row`は違う名前でもよい。

## ■ レコードからフィールド値の取得

```
$row['フィールド名']
```

`$row`は `fetch()` メソッドの戻り値

### 注意: 条件式が代入文

- ・条件式が代入文である場合は、先に代入文が実行される(= 変数に値が代入される)。条件式のチェックはその変数の中身について行われる。
- ・この例の場合はレコードが取得できた場合に繰り返す(メソッド`fetch()`が`FALSE`を返すまで)。
- ・他のプログラミング言語でもよく使われる。

```
<?php
...
while( $row = $sql->fetch() ) {
    ...
    echo $row['name'];
    echo $row['address'];
    ...
}
...
?>
```

# 演習の準備

- 実習0: 本日の演習用作業フォルダを作成しましょう.
  - ◆ 手順1: 以下のフォルダの中に, 新規フォルダ「php04」を作成する.

Z:\xampp\htdocs\

以下, 本日の演習でのファイルはすべてこの「php04」フォルダに作成すること.

# PHPからMySQLへのアクセス(4)

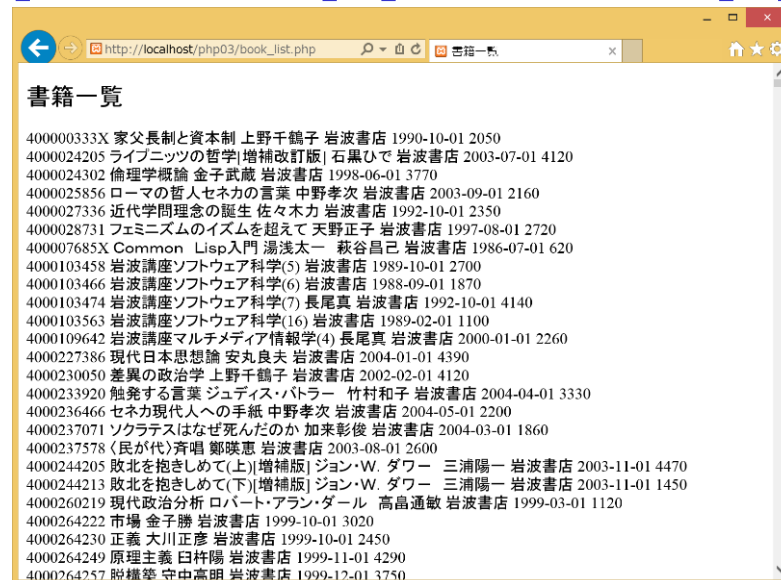
- 実習1: データベースからデータを読み込み, そのデータを表示するWebページ作成してみましょう
  - 「book」テーブルから書籍のデータを一覧表示する.

◆ 手順1: 以下 (book\_list.php) のPHPのファイルを「book\_list.php」という名前で作業フォルダに作成する.

- ????????? の部分は自分で考える.

◆ 手順2: Webブラウザで手順1で作成したページにアクセスし確認する.

「[http://localhost/php04/book\\_list.php](http://localhost/php04/book_list.php)」





```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<title>実習 1 </title>
</head>
<body>

<h2>実習 1 </h2>

<?php
try { // データベース処理の例外範囲始まり
    // データベースへの接続
    $db = new PDO('mysql:host=localhost:63306;dbname=?????;charset=utf8','?????','?????');
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION); // PDOエラー時に例外発生させるように設定

    // SQL文の準備と実行
    $sql = $db->prepare("????????????????????");
    $sql->execute();

    // レコードごとの処理
    while( $row = $sql->fetch() ) {
        echo $row['?????????????'] . " ";
        echo $row['title'] . " ";
        echo $row['author'] . " ";
        echo $row['publisher'] . " ";
        echo $row['pub_date'] . " ";
        echo $row['price'] . "<br>";
    }
} catch( PDOException $e ){ // データベース処理の例外キャッチ
    exit('データベース処理失敗:'.$e->getMessage());
}
?>
</body>
</html>
```

# PHPからMySQLへのアクセス(5)

## ■ 実習2:【実習1の改良】データベースからデータを読み込み, そのデータを表示するWebページ作成してみましょう

- 「book」テーブルから書籍のデータを一覧表示する.
- TABLEタグを使って見やすくする.

◆ 手順1: 実習1で作成した「book\_list.php」をコピーして, 名前を「book\_list2.php」する.

◆ 手順2: 適宜プログラムを変更する.

◆ 手順3: Webブラウザで手順1で作成したページにアクセスし確認する.  
「[http://localhost/php04/book\\_list2.php](http://localhost/php04/book_list2.php)」



The screenshot shows a web browser window with the URL [http://localhost/php03/book\\_list2.php](http://localhost/php03/book_list2.php). The page title is "書籍一覧" (Book List). The browser displays a table with the following data:

| ISBN       | タイトル              | 著者        | 出版社  | 発行日        | 価格   |
|------------|-------------------|-----------|------|------------|------|
| 400000333X | 家長制と資本制           | 上野千鶴子     | 岩波書店 | 1990-10-01 | 2050 |
| 4000024205 | ライプニッツの哲学[増補改訂版]  | 石黒ひで      | 岩波書店 | 2003-07-01 | 4120 |
| 4000024302 | 倫理学概論             | 金子武蔵      | 岩波書店 | 1998-06-01 | 3770 |
| 4000025856 | ローマの哲人セネカの言葉      | 中野孝次      | 岩波書店 | 2003-09-01 | 2160 |
| 4000027336 | 近代学問理念の誕生         | 佐々木力      | 岩波書店 | 1992-10-01 | 2350 |
| 4000028731 | フェミニズムのイズムを超えて    | 天野正子      | 岩波書店 | 1997-08-01 | 2720 |
| 400007685X | Common Lisp入門     | 湯浅太一 萩谷昌己 | 岩波書店 | 1986-07-01 | 620  |
| 4000103458 | 岩波講座ソフトウェア科学(5)   |           | 岩波書店 | 1989-10-01 | 2700 |
| 4000103466 | 岩波講座ソフトウェア科学(6)   |           | 岩波書店 | 1988-09-01 | 1870 |
| 4000103474 | 岩波講座ソフトウェア科学(7)   | 長尾真       | 岩波書店 | 1992-10-01 | 4140 |
| 4000103563 | 岩波講座ソフトウェア科学(16)  |           | 岩波書店 | 1989-02-01 | 1100 |
| 4000109642 | 岩波講座マルチメディア情報学(4) | 長尾真       | 岩波書店 | 2000-01-01 | 2260 |

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<title>書籍一覧</title>
</head>
<body>

<h2>書籍一覧</h2>

<?php
try { // データベース処理の例外範囲始まり

    // データベースへの接続
    $db = new PDO('mysql:host=localhost:63306;dbname=??????;charset=utf8','??????','??????');
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION); // PDOエラー時に例外発生させるように設定

    // SQL文の準備と実行
    $sql = $db->prepare("????????????????????");
    $sql->execute();
```

次につづく

```
echo "<table>";
echo "<tr bgcolor='#99FF99'><th>ISBN</th><th>タイトル</th><th>著者</th><th>出版社</th><th>発行日</th><th>価格</th></tr>";

// レコードごとの処理
while( $row = $sql->fetch() ) {
    echo "<?? bgcolor='#EEEEEE'>";
    echo " <??>" . $row['????????????????'] . "</??>";
    echo " <??>" . $row['title'] . "</??>";
    echo " <??>" . $row['author'] . "</??>";
    echo " <??>" . $row['publisher'] . "</??>";
    echo " <??>" . $row['pub_date'] . "</??>";
    echo " <??>" . $row['price'] . "</??>";
    echo "</tr>";
}

echo "</?????>";

} catch( PDOException $e ){ // データベース処理の例外キャッチ
    exit('データベース処理失敗:'.$e->getMessage());
}
?>

</body>
</html>
```

# PHPからMySQLへのアクセス ( + ユーザからの入力 ) ( 1 )

## ■ SQLインジェクション

➡ ユーザからの入力データをSQL文の条件値などに直接利用した場合に、入力データによって意図しない不正な実行が行われるこ

```
<html>
...
<form action="search.php" method="GET">
...
検索する書籍タイトル: <br>
<input type="text" name="title">
...
?>
```

book\_search.html

```
<?php
$val = $_GET['title'];
...
$sql = $db->prepare("SELECT * FROM book WHERE title='" . $val . "'");
$sql->execute();
...
?>
```

search.php

http://www.hoge.jp/book\_search.html

検索する書籍タイトル:



SELECT \* FROM book WHERE title='データベース入門'

http://www.hoge.jp/book\_search.html

検索する書籍タイトル:



SELECT \* FROM book WHERE title='';DELETE FROM book;#'

# PHPからMySQLへのアクセス ( + ユーザからの入力 ) ( 2 )

## ■ SQLオブジェクトの準備 ( プレースホルダ付き )

```
$sql = $db->prepare("プレースホルダを含むSQL文");
```

インジェクション対策として、ユーザの入力データをSQL文の値として使用する場合には**プレースホルダ**を使う。  
プレースホルダとは後で(以下の**bindValue**実行時)具体的な値に置換される部分。「?」で記述する。  
SQL文の中でプレースホルダは複数記述可能。  
\$dbはPDOオブジェクト。\$sqlは他の名前でもよい。

## ■ プレースホルダへの値の設定 ( = バインド )

```
$sql->bindValue(プレースホルダ番号, 設定したい値);
```

プレースホルダ番号はSQL文の中での「どのプレースホルダか」を指定する番号 (SQL文内での記述順に1, 2, 3, …)。

```
<?php
$val1 = $_GET['title'];
$val2 = $_GET['author'];
...
$sql = $db->prepare("SELECT * FROM book WHERE title=? OR author=?");
$sql->bindValue( 1, $val1 ); // 1番目のプレースホルダに$val1
$sql->bindValue( 2, $val2 ); // 2番目のプレースホルダに$val2
$$sql->execute();
...
?>
```

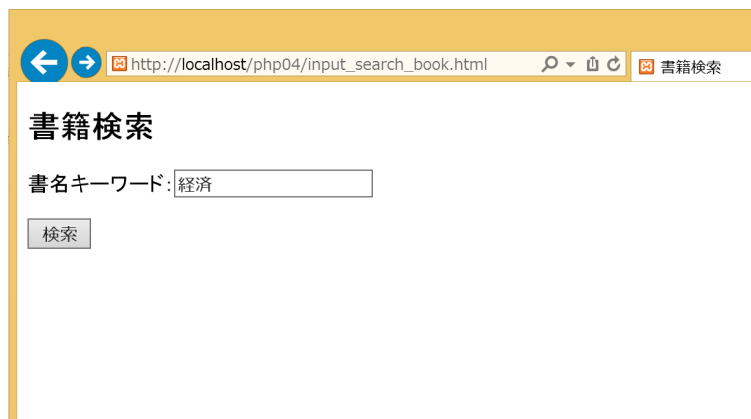
```
$sql = $db->prepare("SELECT * FROM book WHERE title=" . $val1 . " OR author=" . $val2 . "');
```

と書いてはいけない ( SQLインジェクションの危険性 )

# PHPからMySQLへのアクセス (+ ユーザからの入力) (3)

- 実習3: 検索キーワードを入力し「検索」ボタンを押すと、入力されたキーワードを**書名(タイトル)**に含む書籍の一覧を表示するWebページを作りましょう。

- ◆ 手順1: HTMLのファイル(「キーワード」入力Webページ)を「input\_keyword\_book.html」という名前で作業フォルダに作成する。
- ◆ 手順2: PHPプログラムのファイルを「search\_book.php」という名前で作業フォルダに作成する(前回の資料を参考にする)。
- ◆ 手順3: Webブラウザで手順1で作成したページにアクセスし確認する。  
「http://localhost/php04/input\_keyword\_book.html」



input\_keyword\_book.html

入門  
keyword

書籍一覧

検索キーワード: 経済

| ISBN       | タイトル                   | 著者                                | 出版社      | 発行日        | 価格   |
|------------|------------------------|-----------------------------------|----------|------------|------|
| 4130421018 | 経済学のための数学入門            | 神谷和也 浦井憲                          | 東京大学出版会  | 1996-01-01 | 1350 |
| 4478373590 | 競争戦略の経済学               | ペンシルバニア大学ウォートン・スクール ロンドン・ビジネススクール | ダイヤモンド社  | 2001-04-01 | 3870 |
| 4492313222 | マクロ経済学(1[入門篇])[第2版]    | N. グレゴリ・マンキュー 足立英之                | 東洋経済新報社  | 2003-04-01 | 430  |
| 4492313346 | マクロ経済学(2[応用篇])[第2版]    | N. グレゴリ・マンキュー 足立英之                | 東洋経済新報社  | 2004-04-01 | 2600 |
| 4641121559 | 国際化時代の地域経済学[改訂版]       | 岡田知弘                              | 有斐閣      | 2002-04-01 | 1910 |
| 4657033069 | マルチメディアと経済・社会          | 佐竹元一郎                             | 早稲田大学出版部 | 2003-03-01 | 4030 |
| 4750313823 | 叢書現代の経済・社会とジェンダー(第2巻)  | 竹中恵美子 久場嬉子                        | 明石書店     | 2001-03-01 | 1650 |
| 4750314722 | 叢書現代の経済・社会とジェンダー(第5巻)  | 竹中恵美子 久場嬉子                        | 明石書店     | 2001-10-01 | 2970 |
| 4750315060 | 叢書 現代の経済・社会とジェンダー(第3巻) | 竹中恵美子 久場嬉子                        | 明石書店     | 2001-12-01 | 1240 |
| 4750315540 | 叢書現代の経済・社会とジェンダー(第1巻)  | 竹中恵美子 久場嬉子                        | 明石書店     | 2002-03-01 | 2100 |
| 4792341817 | 経済学へのアプローチ[第2版]        | 石橋春男                              | 成文堂      | 2002-05-01 | 830  |
| 482224394X | MBAミクロ経済学              |                                   | 日経BP社    | 2004-04-01 | 1480 |

search\_book.php

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<title>書籍検索</title>
</head>
<body>

<h2>書籍検索</h2>

<form action="search_book.php" method="??????">

<p>
書名キーワード: <input type="text" name="??????">
</p>

<p>
<input type="submit" value="検索">
</p>

</form>

</body>
</html>
```

input\_keyword\_book.html





# 提出課題

## ■ 課題1 (提出課題): データベースからデータを読み込み, そのデータを表示するWebページ作成してみましょう

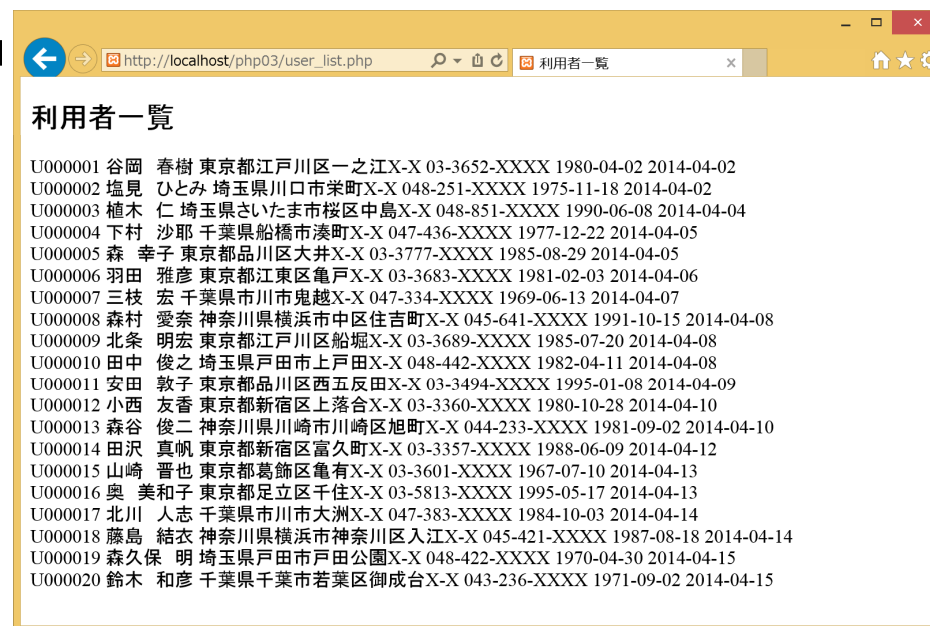
- 「user」テーブルから書籍のデータを一覧表示する.

### ◆ 手順1: PHPのファイルを「user\_list.php」という名前で作業フォルダに作成する.

- プログラムは自分で考える.

### ◆ 手順2: Webブラウザで手順1で作成したページにアクセスし確認する.

「[http://localhost/php04/user\\_list.php](http://localhost/php04/user_list.php)」



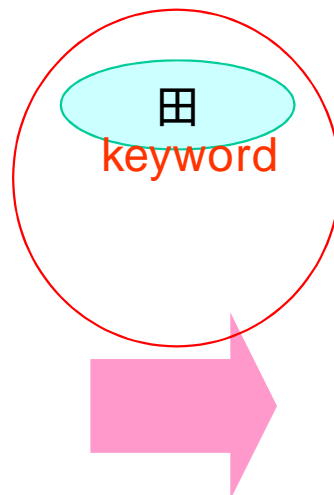
# 提出課題

- 課題2 (提出課題): 検索キーワードを入力し「検索」ボタンを押すと、入力されたキーワードを氏名に含む利用者の一覧を表示するWebページを作りましょう。
  - ◆ 手順1: HTMLのファイル(「キーワード」入力Webページ)を「input\_keyword\_user.html」という名前で作業フォルダに作成する。
  - ◆ 手順2: PHPプログラムのファイルを「search\_user.php」という名前で作業フォルダに作成する(前回の資料を参考にすること)。
  - ◆ 手順3: Webブラウザで手順1で作成したページにアクセスし確認する。  
「http://localhost/php04/input\_keyword\_user.html」

ユーザー検索

氏名キーワード:

input\_keyword\_user.html



利用者一覧

検索キーワード: 田

| ユーザ番号   | 氏名    | 住所            | 電話番号         | 生年月日       | 登録年月日      |
|---------|-------|---------------|--------------|------------|------------|
| U000006 | 羽田 雅彦 | 東京都江東区亀戸X-X   | 03-3683-XXXX | 1981-02-03 | 2014-04-06 |
| U000010 | 田中 俊之 | 埼玉県戸田市上戸田X-X  | 048-442-XXXX | 1982-04-11 | 2014-04-08 |
| U000011 | 安田 敦子 | 東京都品川区西五反田X-X | 03-3494-XXXX | 1995-01-08 | 2014-04-09 |
| U000014 | 田沢 真帆 | 東京都新宿区富久町X-X  | 03-3357-XXXX | 1988-06-09 | 2014-04-12 |

search\_user.php

# 提出課題

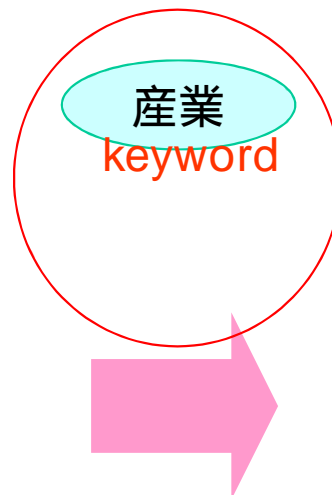
- 課題3 (提出課題): 検索キーワードを入力し「検索」ボタンを押すと、入力されたキーワードを**書名(タイトル)**または**著者**に含む書籍の一覧を表示するWebページを作しましょう。
  - ◆ 手順1: 実習3で作成した「input\_keyword\_book.php」をコピーして、名前を「input\_keyword\_book2.php」に変えるする。
  - ◆ 手順2: 「input\_keyword\_book2.php」について適宜プログラムを変更する。
  - ◆ 手順3: 実習3で作成した「search\_book.php」をコピーして、名前を「search\_book2.php」する。
  - ◆ 手順4: 「search\_book2.php」について適宜プログラムを変更する。
  - ◆ 手順5: Webブラウザで手順1で作成したページにアクセスし確認する。  
「[http://localhost/php04/input\\_keyword\\_book2.html](http://localhost/php04/input_keyword_book2.html)」

書籍検索

書名・著者名キーワード:

検索

input\_keyword\_book2.html



書籍一覧

検索キーワード: 産業

| ISBN       | タイトル                | 著者                        | 出版社     | 発行日        | 価格   |
|------------|---------------------|---------------------------|---------|------------|------|
| 4478320896 | 「新設」産業人の未来          | ピーター・ファーディナンド・ドラッカー 上田 惇生 | ダイヤモンド社 | 1998-06-01 | 1760 |
| 4496028905 | レジャー産業のマーケティング戦略    | 宇田川 荘二                    | 同友館     | 1999-08-01 | 1050 |
| 4621070983 | 映像コンテンツ産業論          | 菅谷実 中村清                   | 丸善      | 2002-09-01 | 3810 |
| 4806119806 | 中国の自動車産業がニッポンを追い抜く日 | 遠藤 誉                      | 中経出版    | 2004-03-01 | 2790 |
| 4876496021 | 循環ビジネス戦略            | 経産省                       | ケイブ出版   | 2004-04-01 | 1970 |

search\_book2.php

# 提出課題

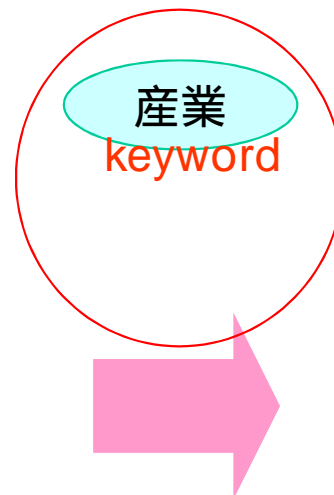
ヒントは次ページ

- 課題4 (オプション課題): 検索キーワードを入力し「検索」ボタンを押すと、入力されたキーワードを**書名(タイトル)**または**著者**に含む書籍の一覧を表示し、キーワードを**黄色でハイライト**するWebページを作りましょう。
  - ◆ 手順1: 課題3で作成した「input\_keyword\_book2.php」をコピーして、名前を「input\_keyword\_book3.php」に変えるする。
  - ◆ 手順2: 「input\_keyword\_book3.php」について適宜プログラムを変更する。
  - ◆ 手順3: 課題3で作成した「search\_book2.php」をコピーして、名前を「search\_book3.php」する。
  - ◆ 手順4: 「search\_book3.php」について適宜プログラムを変更する。
  - ◆ 手順5: Webブラウザで手順1で作成したページにアクセスし確認する。  
「[http://localhost/php04/input\\_keyword\\_book3.html](http://localhost/php04/input_keyword_book3.html)」

書籍検索

書名・著者名キーワード:

input\_keyword\_book3.html



| ISBN       | タイトル                         | 著者                 |
|------------|------------------------------|--------------------|
| 4478320896 | 「新訳」 <b>産業</b> 人の未来          | ピーター・ファーディナン<br>惇生 |
| 4496028905 | レジャー <b>産業</b> のマーケティング戦略    | 宇田川 荘二             |
| 4621070983 | 映像コンテンツ <b>産業</b> 論          | 菅谷 実 中村 清          |
| 4806119806 | 中国の自動車 <b>産業</b> がニッポンを追い抜く日 | 遠藤 誉               |
| 4876496021 | 循環ビジネス戦略                     | 経済 <b>産業</b> 省     |

# 提出課題

## ■ 課題4 (オプション課題) のヒント

### ◆ ハイライトするHTMLタグ

```
<mark>ハイライトしたい文字列</mark>
```

経済<mark>産業</mark>省



経済産業省

### ◆ PHPでの文字列の置換

- 対象文字列の中から、検索文字列を探し、当該部分を指定した置換後文字列に置き換える。

```
str_replace(検索文字列, 置換後文字列, 対象文字列)
```

```
$val = "山田さんと卓球をした。山田さんは卓球がうまい。";  
echo $val;  
echo str_replace("山田", "城之内", $val);
```



```
山田さんと卓球をした。山田さんは卓球がうまい。  
城之内さんと卓球をした。城之内さんは卓球がうまい。
```