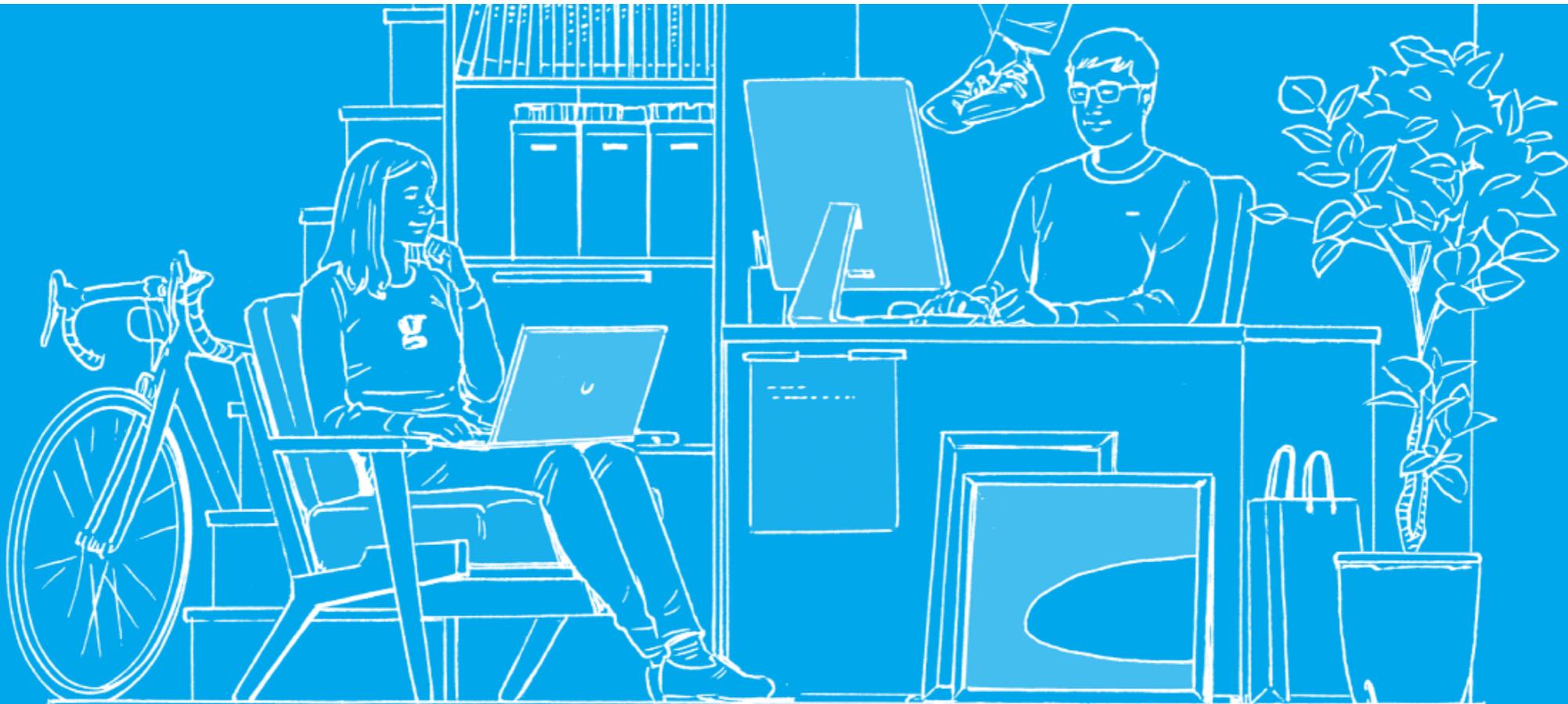


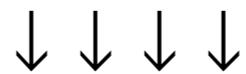


**G's ACADEMY**  
**TOKYO**

# PHP & DB



授業用チャーター呼び出しサービス



<http://www.call.jp/room/gsacademy-tokyo>

# 今日の授業内容

PHP & DB & MySQL

# アジェンダ

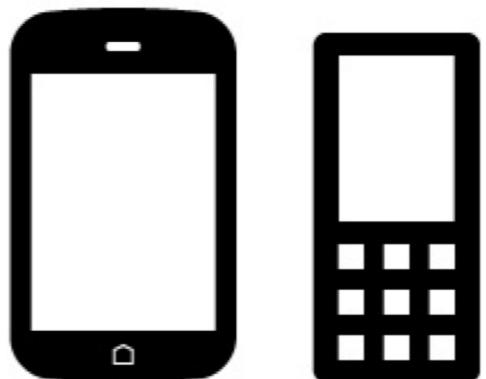
- ・ データベースとは
- ・ データベース作成
- ・ データベース操作
- ・ PHPでデータ登録
- ・ 課題

# PHPとJSの違い

# サーバサイドはリクエストレスポンス

【クライアントサイド】  
HTML, CSS, JavaScript  
ブラウザ(PC)上で動作

JS clickしてもブラウザの中で処理



リクエスト  
`http://***.php/`

PHP  
`<?=$view?>`



【 PHP:サーバサイド】

clickしたらサーバーとリクエスト/レスポンス  
※ページが必ず読み込まれる仕組みです

レスポンスファイルにはPHP文字列は無く、HTMLに置き換わっている状態が届く

`<div>ABC</div>`

# バグの確認

- 追加記述したらすぐに確認！！
- エラーが表示されたら  
ファイル名/行数をすぐに確認！！
- ブラウザ確認は入力画面から！！  
登録処理・更新処理・削除処理だけのファイル  
を表示するとエラーです（これは当然と知る）

# データベースとは

データベースとは？

データを一定のルールで蓄積し、必要に応じて取り出せるようにしたもの

◆データベースはExcelファイルと似ています。

- ・データベース : Excelで言うとファイルです。
- ・テーブル（表） : ExcelではSheetです。
- ・レコード（行・ロウ） : Excelでは横の行です。
- ・フィールド（列・カラム） : Excelでは縦の列です。



# データベースと Excel の比較

## ◇ Excel の場合

ファイル名 : gs\_db

Sheet名 : gs\_an\_table

項目名: id ( ユニーク値 [1…max]重複しない値 )

項目名: name

項目名: email

項目名: naiyou

項目名: indate

## ◇ データベースの場合

データベース名 : gs\_db

テーブル名 : gs\_an\_table

フィールド名: id int (数値) ユニーク値

フィールド名: name varchar(文字列)

フィールド名: email varchar (文字列)

フィールド名: naiyou varchar (文字列)

フィールド名: indate datetime (日付)

# テーブルの作成

## ◇ データベースの場合

データベース名 : gs\_db

テーブル名 : gs\_an\_table

フィールド名 : id int (数値) ユニーク値

フィールド名 : name varchar(文字列)

フィールド名 : email varchar (文字列)

フィールド名 : naiyou text

フィールド名 : indate datetime (日時)

## ◇ 実際にテーブルを設計すると

テーブル名 : gs\_an\_table

フィールド名 :

id int (12) PRIMARY KEY : AUTO\_INCREMENT

name varchar (64)

email varchar (128)

naiyou text

indate datetime

# データベース作成

# Database作成 (XAMPP使用した場合)

- 最初にデータベースを作成します。 <http://localhost/xampp/>
- 画面に左のメニュー “phpMyAdmin” を選択
- 『データベース』タブをクリック
- 『データベースを作成する』に半角英数で任意のデータベース名、「**gs\_db**」を入力し作成。
- 『照合順序』は **utf8\_unicode\_ci** を選択。
- 『作成』ボタンで作成。



# アンケートシステムのDB構築

## ◇テーブル作成

- DB名: **gs\_db**
- Table名: **gs\_an\_table**
- Field名:

id:	int(12) <u>AUTO INCREMENT PRIMARY KEY</u>
name :	var_char(64)
email :	var_char(128)
naiyou :	text
indate:	datetime

※フォームの項目を増やした場合こちらのFieldも増やしましょう。

※Fieldの右にあるのは、データ型(Type)です。

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

# データベース操作

# テーブル基本操作：データベース言語 SQL

MySQLは以下のコマンドを使用してデータ登録・更新・削除が可能です。

## ◇ SQLコマンド（データ操作）

INSERT : データを“登録”する事ができます。

SELECT : データを“表示”する事ができます。

UPDATE : データを“更新”する事ができます。

DELETE : データを“削除”する事ができます。

## ◇SQLのコメント

例 : -- SELECT \* FROM テーブル名;

“–”ハイフンを連続2文字並べることで次の文字からコメントになります。

## ◇SQLで文字列を扱う

シングルクオートで囲む。ダブルクオートはNG。

# テーブル基本操作 SQL

## ■ INSERT (データ登録)

◇書式 :

**INSERT INTO テーブル名(カラム1, カラム2,...)**

**VALUES(値1, 値2,...);**

---

```
INSERT INTO gs_an_table(id, name, email, naiyou, indate)
VALUES(NULL, 'ジーズ太郎', 'test1@test.test', 'テスト1', '2015-06-15
00:00:00');
```

```
INSERT INTO gs_an_table(id, name, email, naiyou, indate)
VALUES(NULL, 'ジーズ次郎', 'test2@test.test', 'テスト2', sysdate() );
```

---

※文字列を登録する場合シングルクオートで囲んで指定します。

# テーブル基本操作 SQL

## ■ SELECT (データ取得)

◇書式：

**SELECT 表示するカラム FROM テーブル名;**

---

SELECT \* FROM gs\_an\_table; --全指定

SELECT name FROM gs\_an\_table; --単体指定

SELECT name, email FROM gs\_an\_table; --複数指定

---

SELECT \* FROM gs\_an\_table WHERE name = 'ジーズ太郎';

※WHERE を使用して特定のデータを抽出することが可能です。

---

## 条件付き検索

### ◇以下条件付きSQLの一例)

- ・演算子を使う

```
SELECT * FROM テーブル名 WHERE id = 1;
```

```
SELECT * FROM テーブル名 WHERE id >= 3;
```

- ・AND, OR で検索条件を複数指定する

```
SELECT * FROM テーブル名 WHERE id = 1 OR id = 2;
```

```
SELECT * FROM テーブル名 WHERE id >= 1 AND id<=3;
```

- ・あいまい検索をする

```
SELECT * FROM テーブル名 WHERE indate LIKE '2015-06%';
```

```
SELECT * FROM テーブル名 WHERE email LIKE '%@gmail.com';
```

```
SELECT * FROM テーブル名 WHERE email LIKE '%@%';
```

## ソートと制限

### ◇表示をソートする

書式：

SELECT \* FROM テーブル名 ORDER BY ソートしたいカラム名 \*\*\*;

SELECT \* FROM テーブル名 ORDER BY id DESC;

SELECT \* FROM テーブル名 ORDER BY email, name DESC;

※ DESCは降順でSORT、ASCは昇順でSORT

### ◇表示件数を制限する

書式：

SELECT 表示するカラム FROM テーブル名 LIMIT \*\*\*;

SELECT \* FROM テーブル名 LIMIT 5;

SELECT \* FROM テーブル名 LIMIT 3, 5;

# PHP & MySQL データ登録処理

# PHPからMySQLを操作する方法「 PDO 」を知りましょう！

## ◆ PDOとは？

PDOは「PHP Data Objects」の略で、PDOを使うことで  
「MySQL・SQLite・PostgreSQL」など違うデータベースを利用する  
場合でも、同じ関数で使うことができます。とても便利なもので  
す。（ PDOはPHP5.1以降に標準で装備されました。）

## ◆ プリペアードステートメント

prepare()に”プレースホルダ（挿入位置を決めるための文字列）”を  
用いた SQL を渡し、実際の値は [bindValue\(\)](#) や bindParam() など  
で指定します。SQL文内にある”プレースホルダ”に変数をバインド  
(代入・関連付ける) する際に「値は適切にエスケープ」されます。

※ エスケープ = 無効化する (SQLインジェクション対策)

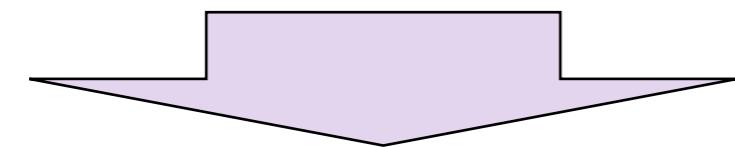
### 【補足】

「->」の記号は…の中の…という意味

-----  
echo \$stmt->queryString;

-----  
実行したSQLを表示出来ます！

\$db = new PDO('DB種類名:オプション属性');



## \$dbオブジェクト->prepare()->○○を使う！

- ① SQL文字列をqueryStringに保持

\$dbオブジェクト->prepare()->queryString;

- ② bind変数を使いSQLに値を渡す

\$dbオブジェクト->prepare()->bindValue()

- ③ SQL実行

\$dbオブジェクト->prepare()->execute()

- ④ データ取得

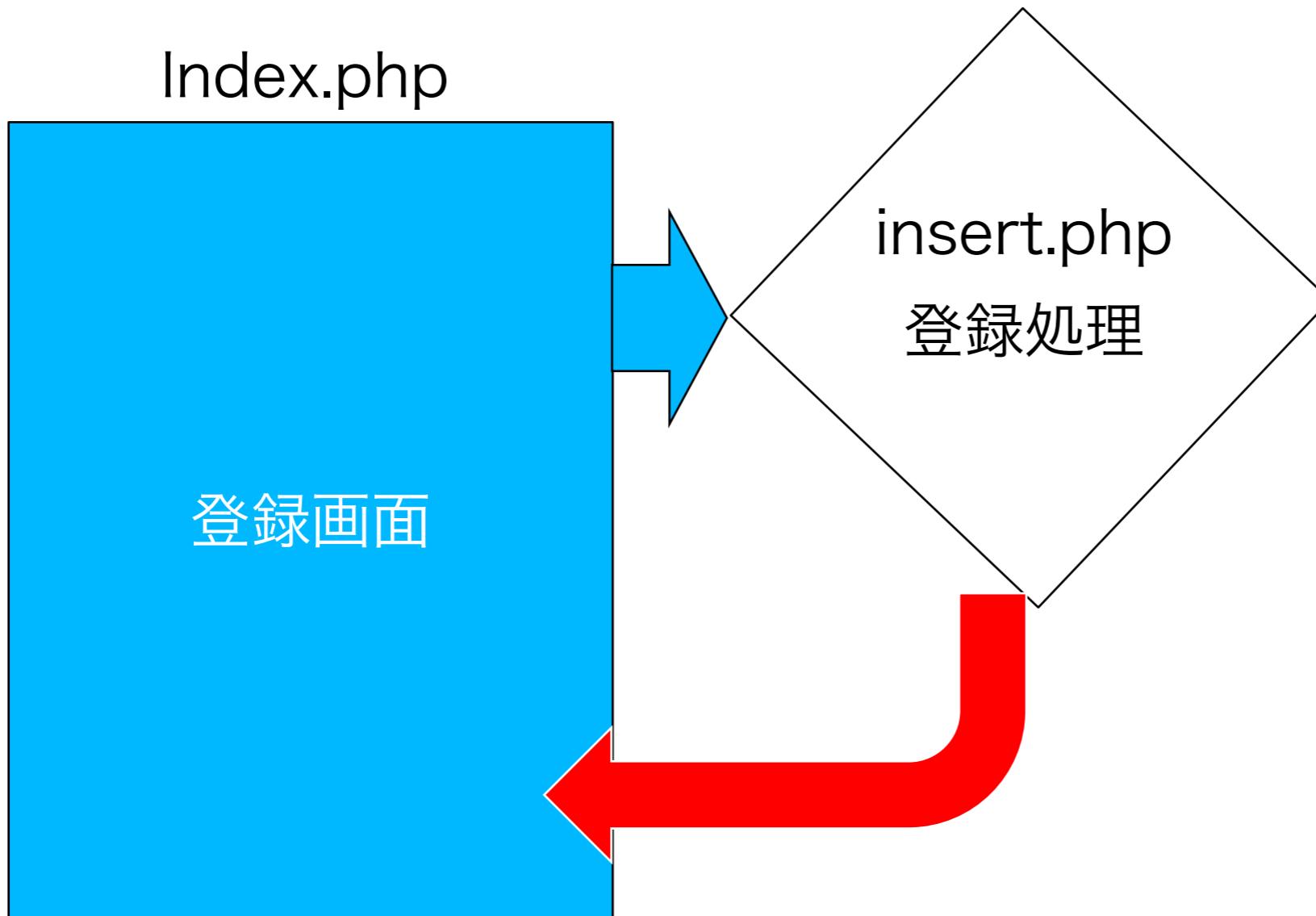
\$dbオブジェクト->prepare()->fetch()

# PHPとDB接続

データ登録

# データ登録の流れ

---



## PHPとデータベースの接続 (データ登録)

◇サンプルコード : [insert.php](#)

```
<?php
try {
    $pdo=new PDO('mysql:dbname=gs_db;charset=utf8;host=localhost','root','');
} catch (PDOException $e) {
    exit( 'DbConnectError:' . $e->getMessage() );
}

//データ登録SQL作成
$sql="INSERT INTO gs_an_table ( id, name, email, naiyou, indate )
VALUES( NULL, :a1, :a2, :a3, sysdate() )";

$stmt = $pdo->prepare($sql);
$stmt->bindValue(':a1', 'ジーズ三郎', PDO::PARAM_STR);
$stmt->bindValue(':a2', 'test3@test.test', PDO::PARAM_STR);
$stmt->bindValue(':a3', 'テスト3', PDO::PARAM_STR);

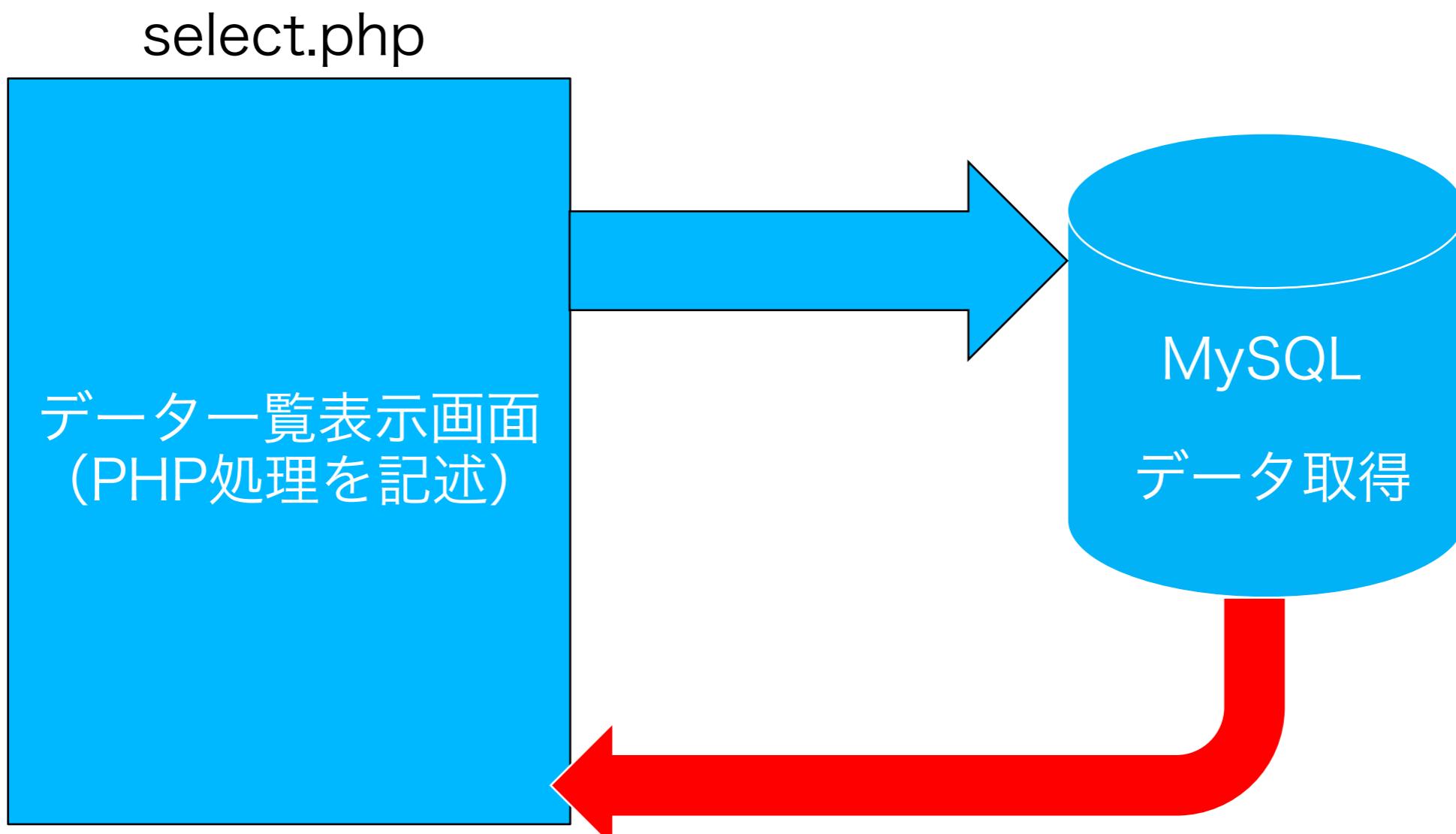
//SQL実行
$flag = $stmt->execute();
?>
```

# PHPとDB接続

データ抽出&表示

# データ取得の流れ

---



## PHPとデータベースの接続 (データ取得表示)

◇サンプルコード : [select.php](#)

```
$stmt  = $pdo->prepare("SELECT * FROM gs_an_table");
$status = $stmt->execute();

$view  = "";
if($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 {
    //Selectデータの数だけ自動でループしてくれる
    while( $result = $stmt->fetch(PDO::FETCH_ASSOC)){
        $view .= '<p>';
        $view .= $result["indate"] . " : ". $result["name"] ;
        $view .= '</p>';
    }
} //※$view を表示したい場所でechoしましょう！
```

# JS・PHPの連携 (中級レベル)

## JS/PHPデータ連携例（PHPとJSは同じFile内の記述です）

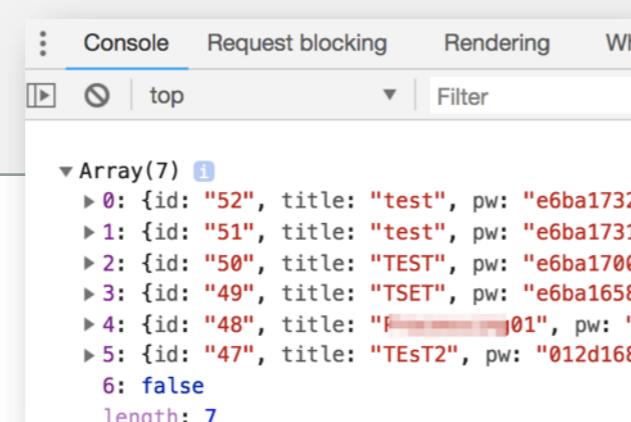
```
//PHP処理  
// "配列$result"に全てのデータを代入できます。
```

```
while( $result[] = $stmt->fetch(PDO::FETCH_ASSOC));  
$json = json_encode($result);  
?>
```

```
//JavaScript処理  
<script>
```

```
const data = JSON.parse('<?=json?>'); //JSON文字列→配列に変換  
console.log(data); //配列値の確認
```

```
</script>
```



# 課題

# 課題テーブル仕様

本をブックマークするDBを作りましょう！

- DB名: **好きなDB名で新しく作成**
- 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)

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！

# 課題登録ページ作成

本をブックマークするPHP処理を作りましょう！

- フォーム項目名:

1. ユニーク値 (int 12, PRIMARY, AutoIncrement)
2. 書籍名 (varChar 64)
3. 書籍URL (text) Index.php
4. 書籍コメント(text)
5. 登録日時 (date)

