憂鬱なSQLのためのアレ

PHP-BLT #4

近況報告

- PHPカンファレンス北海道で喋ってきました
  「仕事で使えるComposer」
  http://niconare.nicovideo.jp/watch/kn1371

- WEB+DB Press Vol. 92に
  PHPからHTTPリクエストをする話を書きました
  http://gihyo.jp/magazine/wdpress/archive/2016/vol92

近況報告

- AdventCalendarではPHPマニュアルのバージョン番号をパースする話を書きました
  http://qiita.com/tadsan/items/94e00f6fdf40d96c5072
- スレッドフロート掲示板を作りたかった少年時代の代償行為として、モダンPHP、ただしフレームワーク抜きで書き始めました
  http://qiita.com/tadsan/items/cdbbb5b08591af2b110d

みんな大好きPDO

- PHP Data Objects
- いろんなSQLを抽象化したオブジェクト
- 学生がmysqli関数使ってたけどPDOでいい
- あとなんか、Doctrineとか使った方がいいの?

Real World PHP

- 2015年にもなって徳丸先生がPHPカンファレンスでSQLインジェクションの話をしてるリアル

- O/Rマッパー使っとけｗｗｗと煽ったところで、世界にはまだ文字列結合SQLが眠ってる

みんな事故る

- 文字列結合の温床
- 型を指定するのはめんどくさい
- `$stmt->bindValue(':id', $id, \PDO::PARAM_INT)`
- 可変個数のIN句が地雷

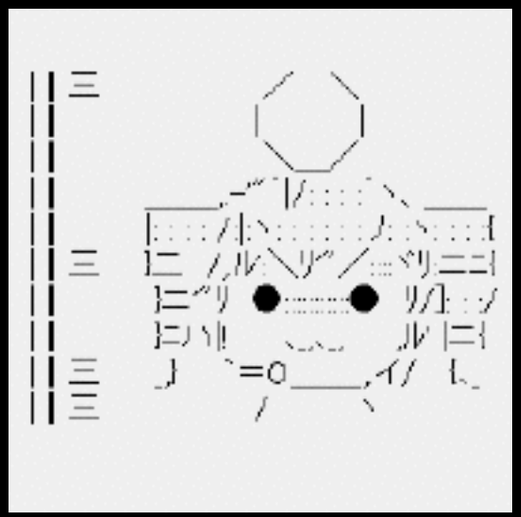
Real World PHP

```
$names = ["taro","jiro","saburo"];
$str = implode(",", array_map($name){
    return sprintf('"%s"', $name);
});
$sql = "SELECT * FROM users WHERE name IN ($str)";

// ↑典型的な漏れのあるロジックの例
```

PDOのラッパー作った

- 警告：ちゃんとしたO/Rマッパーの方がいい
- (会社で同僚が作ったのを)コピペして再実装した
- SQL内に型が書けるDSL
- PDOに乗っかって、余計なことはしない
- SQLインジェクションにある程度の耐性がある?



```
composer require zonuexe/tetosql
```

https://github.com/BaguettePHP/TetoSQL

```php
<?php
namespace InspireBBS\Model;
use Teto\SQL;

/**
 * 板を表現するモデル
 * @copyright 2016 USAMI Kenta
 * @license   WTFPL
 */
final class Board {
    /**
     * @return Board[]
     */
    public static function findAll() {
        $data = SQL\Query::execute(db(), self::findAll_query, [])
            ->fetchAll(\PDO::FETCH_ASSOC) ?: [];

        $boards = [];
        foreach ($data as $b) {
            $boards[] = new Board($b);
        }

        return $boards;
    }
    const findAll_query = 'SELECT `id`, `name`, `text` FROM `boards`';
}
```

```
use Teto\SQL;
public static function find($id) {
    $data = SQL\Query::execute(db(), self::findAll_query, [
        ':id' => $id,
    ])->fetchAll(\PDO::FETCH_ASSOC) ?: [];

    $boards = [];
    foreach ($data as $b) {
        $boards[] = new Board($b);
    }

    return $boards;
}
const find_query =  `
    SELECT `id`, `name`, `text` FROM `boards` WHERE `id` = $id@int
';
```

基本的なAPI

- どうにかしてPDOのコネクションをとっておく

- ```
  $stmt = SQL\Query::execute($con, self::find_query,
   [':id' => $id, ':hoge' => $hoge] );
  ```

- コネクション、SQL、アサインする値を渡す

- 帰ってくる値はPDOStatementなので、
  いつもの感覚でfetch/fetchAllするだけ！
  ```

const

- PHPのconstは(原則)実行時に変更されない
- そこにパラメータを当て嵌めるだけ、なのはなんとなく心理的な安心感がある (本当か?)
- お気付きだろうか…
- このコードがPSR-1違反であることに…