■PHP5_4移行上の留意点と新機能について
-- 作成：2014.6.6 yoshi --

■PHP 5.3.x から PHP 5.4.x への移行
http://se2.php.net/manual/ja/migration54.php

■初期設定ファイル：php.ini ディレクティブ
http://se2.php.net/manual/ja/ini.core.php

◇short_open_tag -> off
<?= は常に有効となるため、offで良くなった。 ... PHP 5.4.0 以降
<?= が常に有効になりました。 short_open_tag の設定にかかわらずいつでも使える。

・register_globals と register_long_arrays が削除されました。

・session.upload_progress.enabled を有効にすると、
アップロード中の個々のファイルの進捗状況を PHP で追えるようになります。

■変更内容または削除

◇参照渡し
呼び出し時の参照渡しをサポートしなくなりました。
foo(&$a); のような形式で & を利用するとfatal エラーが発生

◇breakとcontinueの引数
break と continue への引数として、変数は使えなくなりました。
つまり、 break 1 + foo() * $bar; などとは書けなくなったということです。
静的な引数を使うのはもちろん可能で、break 2; などは使えます。
この変更の副作用として、 break 0; や continue 0; が使えなくなりました。

◇配列と文字列の返り値
$a が文字列の場合に非数値のオフセット (たとえば $a['foo']) を指定したときの isset() の返り値は false、
そして empty() の返り値は true となりました。 そして同時に E_WARNING が発生します。
配列を文字列に変換しようとすると E_NOTICE が発生するようになりました。

◇予約語の追加
trait
callable
insteadof

◇関数の削除
session_is_registered(), session_register(), session_unregister()

◇スーパーグローバルをパラメータ名として使用不可
スーパーグローバルをパラメータ名として使うと fatal error となります。
つまり function foo($_GET, $_POST) {} などのコードは使えなくなります。

◇マジッククォートが削除
マジッククオートは、PHPスクリプトに入力されるデータを 自動的にエスケープする機能です。

■新機能内容
◇トレイトのサポート
http://se2.php.net/manual/ja/language.oop5.traits.php
トレイトは、単一継承の制約を減らすために作られたもので、 いくつかのメソッド群を異なるクラス階層にある独立した
クラスで再利用できるようにします。
トレイトはクラスと似ていますが、トレイトは単にいくつかの機能をまとめるためだけのものです。
トレイト自身のインスタンスを作成することはできません。
昔ながらの継承に機能を加えて、振る舞いを水平方向で構成できるようになります。
つまり、継承しなくてもクラスのメンバーに追加できるようになります。
・優先順位
基底クラスから継承したメンバーよりも、トレイトで追加したメンバーのほうが優先されます。
優先順位は現在のクラスのメンバーが最高で、その次がトレイトのメソッド、 そしてその次にくるのが継承したメソッドとなります。
・複数のトレイトをひとつのクラスに追加するには、use 文でカンマ区切りで指定します。
・衝突の解決
同一クラス内での複数のトレイト間の名前の衝突を解決するには、 insteadof 演算子を使って そのうちのひとつを選ばなければなりません。
この方法はひとつのメソッドだけしか使えませんが、 as演算子を使うと、 衝突するメソッドのいずれかを別の名前で
含めることができます。（エイリアスを指定）
・メソッドの可視性の変更
as 構文を使うと、 クラス内でのメソッドの可視性も変更することができます。
・トレイトを組み合わせたトレイト
クラスからトレイトを使えるのと同様に、トレイトからもトレイトを使えます。
トレイトの定義の中でトレイトを使うと、 定義したトレイトのメンバーの全体あるいは一部を組み合わせることができます。
・抽象メソッドを使ってクラスの要件を指定できます。
・静的なメンバーやメソッドを定義できます。
・プロパティも定義できます。

◇配列の短縮系（無名配列、無名連想配列）
$a = [1, 2, 3, 4]; や $a = ['one' => 1, 'two' => 2, 'three' => 3, 'four' => 4];
foo()[0] ... 関数の返り値を配列として扱えるようになりました。

◇二進数フォーマットが追加
0b001001101 のように使えます。

◇ファイルのアップロード状況を追跡
セッションモジュールで、ファイルのアップロード状況を追跡できるようになりました。

◇WEBサーバの組み込み
CLI モードで動く、開発用のWEBサーバが組み込まれました。

◇クラス
・クロージャ（無名関数）が $this をサポート
（PHP5.3から、無名関数が使える）
1. コールバック パラメータとして使う ...
```
preg_replace_callback('~-([a-z])~', function ($match) { return strtoupper($match[1]);}, 'hello-world');
```
2.変数の値として使用 ...
```
$greet = function($name){ printf("Hello %s¥r¥n", $name);};
$greet('World');
```
3.変数を親のスコープから引き継ぐ、引き継ぐ変数は、use で渡さなければなりません。
```
public function getTotal($tax){
   $total = 0.00;
   $callback = function ($quantity, $product) use ($tax, &$total){
        $pricePerItem = constant(__CLASS__ . "::PRICE_" .strtoupper($product));
        $total += ($pricePerItem * $quantity) * ($tax + 1.0);
        };
   array_walk($this->products, $callback);
   return round($total, 2);
}
```
・クラスのインスタンスを生成するときに、そのメンバーにアクセスできる（例: (new Foo)->bar()）
・Class::{expr}() 構文をサポート
クラスのコンストラクタ __construct の引数を、 基底クラスの abstract コンストラクタで強制できるようになりました。

◇新関数とクラスと新しいメソッド
・PHP コア:
```
hex2bin()
http_response_code()
get_declared_traits()
getimagesizefromstring()
stream_set_chunk_size()
socket_import_stream()
trait_exists()
header_register_callback()
```

is_link() が Windows Vista 以降でのシンボリックリンクに対しても適切に機能するようになりました。
parse_url() が、 スキームが省略されていてコンポーネント区切り文字から始まるホスト名を認識するようになりました。（PHP 5.4.7 以降）
・OpenSSL: AES に対応しました。
no padding オプションが openssl_encrypt() および openssl_decrypt() に追加されました。
・Standard PHP Library (SPL):
```
class_uses()

CallbackFilterIterator
RecursiveCallbackFilterIterator

RegexIterator::getRegex()
SplObjectStorage::getHash()
DirectoryIterator::getExtension()
SplDoublyLinkedList::serialize()
SplDoublyLinkedList::unserialize()
SplFileInfo::getExtension()
SplFileObject::fputcsv()
SplQueue::serialize()
SplQueue::unserialize()
SplStack::serialize()
SplStack::unserialize()
SplTempFileObject::fputcsv()
```

・Closure:

```
Closure::bind()
Closure::bindTo()
```

・Json:
JsonSerializable インターフェイス
・Session:

    session_status()
    session_register_shutdown()

    SessionHandler クラス
    SessionHandlerInterface
・Snmp:
    SNMP
・PDO_dblib:
    PDO::newRowset()

・Mysqli:
    mysqli_error_list()
    mysqli_stmt_error_list()
・Zlib:
    zlib_decode()
    zlib_encode()

MySQL 用の拡張モジュールである mysql や mysqli そして PDO_mysql が、デフォルトのライブラリとして mysqlnd を使うようになりました。
libmysqlclient を使うことも可能ですが、その場合は configure のオプションで libmysqlclient のパスを指定します。
mysqlnd - 名前付きパイプをサポートするようになりました。

htmlspecialchars() および htmlentities() のデフォルトの文字セットが ISO-8859-1 から UTF-8 に変わりました。

$_SERVER['REQUEST_TIME_FLOAT'] が追加され、マイクロ秒単位の精度も取得できるようになりました。