

SQLite 関数 直接操作

Windows & PHP5

PHP5 は SQLite が標準で使える。

PHP.ini の編集

Windows 版では PDO も有効にする必要があります。PHP 5.1.0 以降、PDO に依存するようになったからです。

```
extension=php_pdo.dll
extension=php_sqlite.dll
を加える(順序関係あり)
```

実行

```
php test.php
```

Linux & PHP4

SQLite のインストール

```
pear install sqlite
```

PHP.ini,php-cli.ini の編集

```
extension=sqlite.so
を加える
```

実行

```
php test.php
```

Linux & PHP5 & sqlite2

1.yum で php-devel をインストールする

```
# yum install php-devel
```

2.sqlite のソースをダウンロードする

git リポジトリからバージョンに合った PHP のソースから sqlite2 のソースをコピーする。

<http://git.php.net/?p=php-src.git;a=tags>

解凍したディレクトリの

```
ext/sqlite
```

が sqlite2 のソース。

3. コンパイルとインストール

```
# cd ext/sqlite
# phpize
# ./configure
```

```
# make
```

手動で sqlite.so を移動するか

```
# make install
```

これで /usr/lib64/php/modules (32bit は、 /usr/lib/php/modules) に sqlite.so がコピーされる

4.php の設定を編集する

```
# vi /etc/php.d/sqlite2.ini
```

以下の内容を記載する

```
extension = sqlite.so
```

Linux & PHP5 & sqlite2 (古いやり方)

<http://rvrsti.blog.so-net.ne.jp/archive/c10363734-1>

1.yum で php-devel をインストールする

```
# yum install php-devel
```

2.sqlite のソースをダウンロードする

```
# wget http://pecl.php.net/get/SQLite-1.0.3.tgz
```

3. ソースを展開して、環境を設定する

```
# tar xzf SQLite-1.0.3.tgz
# cd SQLite-1.0.3
# phpize
# ./configure
```

4. このままだと make でエラーになるので一部修正

```
# vi sqlite.c
```

```
56 行目
static unsigned char arg3_force_ref[] = {3, BYREF_NONE, BYREF_NONE, BYREF_FORCE };
->行を削除 // でコメントアウトでOK
```

```
125 行目
PHP_FE(sqlite_open, arg3_force_ref)
->arg3_force_ref を third_arg_force_ref に変更
PHP_FE(sqlite_open, third_arg_force_ref)
```

```
126 行目
PHP_FE(sqlite_popen, arg3_force_ref)
->arg3_force_ref を third_arg_force_ref に変更
PHP_FE(sqlite_popen, third_arg_force_ref)
```

5. コンパイル、インストール

```
# make
```

手動で sqlite.so を移動するか

```
# make install
```

これで /usr/lib64/php/modules に sqlite.so がコピーされる

6.php の設定を編集する

```
# vi /etc/php.d/sqlite.ini
```

以下の内容を記載する

```
extension = sqlite.so
```

コード

http://study.rakuto.net/php/sqlite/sqlite_step/

のまま

PDO

5.3.3 以降

一番メジャーな方法。

PHP5.3.3 以上ならば、標準で PDO が使えるはず。

もし使えない場合は、

```
/etc/php.d/pdo.ini  
/etc/php.d/pdo_sqlite.ini
```

等の設定を見直す。

5.3.3 以前

<http://codezine.jp/article/detail/433>

PDO は PHP 5.1 から標準で搭載されるようになりました。使用している PHP のバージョンが 5.1 以降ならば、以下の設定を「php.ini」に追加するだけで PDO を利用することができます。

UNIX 系の場合

```
extension=pdo.so  
extension=pdo_mysql.so  
extension=pdo_pgsql.so  
extension=pdo_sqlite.so  
extension=pdo_firebird.so  
extension=pdo_dblib.so  
extension=pdo_oci.so
```

Windows の場合

```
extension=php_pdo.dll
extension=php_pdo_firebird.dll
extension=php_pdo_mysql.dll
extension=php_pdo_oci.dll
extension=php_pdo_oci8.dll
extension=php_pdo_odbc.dll
extension=php_pdo_pgsql.dll
extension=php_pdo_sqlite.dll
```

もし、PDO のモジュールがインストールされていない場合でも、以下のコマンドで簡単にインストールすることができます。

```
pecl install pdo
```

以下のコマンドを使い、個別の PDO ドライバをインストールすることもできます。

```
pecl install pdo_driver
pecl install pdo_pgsql
```

MDB2

インストール

本体

```
pear install pear/MDB2
```

ドライバ

```
pear/MDB2_Driver_sqlite
```

など。

sqlite.so がインストールされていないとドライバをインストール出来ない。

MDB

本体

```
pear install pear/XML_Parser
pear install pear/MDB
```

使ったことがないのでこれ以上は不明

DB

本体

```
pear install pear/DB
```

ドライバ

php-mysql とか php-pgsql とか sqlite が必要だともう。

使っていないので詳細不明