

## ■セッションとクッキー関連の関数と変数

作成: 2015/8/25 yoshi

関数	機能	返り値	PHP対応バージョン	セッション開始前実行	備考
session_start	新しいセッションを開始、又は既存のセッションを再開する セッションデータを初期化又は、リクエスト上で GET, POST またはクッキーにより渡されたセッション ID に基づき現在のセッションを復帰します。	true/false PHP5.3以降falseも返す			自作のセッション open ハンドラ、read ハンドラを session_set_save_handler() で定義することもできます。
session_name	現在のセッション名を取得または設定する。 名前付きのセッションを使用したい場合、session_name() をコールする。	以前の session_name		○	\$bef_name = session_name('new_name'); \$bef_name に、元のSession Nameがセットされ、'new_name' に設定される。
session_id	現在のセッション ID を取得または設定する セッション ID として使える文字は、「a-zA-Z0-9,-」のみ(アンダースコア)は不可)	以前のsession_id		○	セッションが開始された時点でSIDにセットされる。セッションIDの通知を行うためには次の二つの方法「Cookies、URLパラメータ(セッションIDを直接URLに埋め込み)」があります。
session_cache_expire	現在のキャッシュの有効期限を返す、分単位(デフォルト: 180) 現在のキャッシュ有効期限は、引数のnew_cache_expire で置換する。	現在の設定値		○	session_cache_expire(30); \$cache_expire = session_cache_expire();
session_cache_limiter	現在のキャッシュリミッタを取得または設定する クライアントに送信されるキャッシュ制御用の HTTPヘッダ(キャッシュする規則の定義)を制御 *nocache ...クライアント/プロキシのキャッシュは無効 *public ...クライアントとプロキシのキャッシュ許可 *private ...クライアントのみキャッシュ許可 *private_no_expire ...クライアントのみキャッシュ許可 (但し、Expireヘッダを指定しない)	現在のキャッシュリミッタの名前		○	session_cache_limiter('private'); \$cache_limiter = session_cache_limiter();
session_write_close	セッションデータを書き込んでセッションを終了する。セッションデータは、スクリプト終了時に保存されますが、同時書き込みを防ぐためにロックされるため、処理をスムーズに進行できます。	なし			session_write_close();
session_commit	session_write_close のエイリアス	なし			
session_encode	現在のセッションデータ(\$_SESSION)を、セッションエンコードされた文字列に変換する	セッション内容をエンコードしたも			\$cvt_string = session_encode();
session_decode	セッションエンコードされた文字列からセッションデータ(\$_SESSION)に結果を格納(デコード)する	true/false			session_decode(\$cvt_string); print_r(\$_SESSION);
session_destroy	セッションに登録されたデータを全て破棄する。 セッションに関するグローバル変数、セッションクッキーを破棄しないので、\$_SESSIONを初期化して、セッション変数の破棄すると良い。	true/false			セッション変数の利用を再開するには session_start() をコールする必要があります。ユーザがログアウトするときのように、セッションを切断するには、セッション ID の割り当ても解除する必要があります。
session_module_name	現在のセッションモジュールを取得または設定する。 引数のmodule が指定された場合、そのモジュールを代わりに使用する。	現在のセッションモジュールの名前			
session_regenerate_id	現在のセッションIDを新しく生成したものと置き換える。その際、現在のセッション情報は維持されます。 「delete_old_session」のパラメータを付加すると関連付けられた古いセッションを削除する	true/false			
session_save_path	現在のセッションデータ保存パスを取得または設定する。 引数のpath パスが指定されると、セッションデータを保存するパスが変更されます。	現在のデータ保存先ディレクトリのパス		○	

session_set_save_handler	セッションに関連するデータを保存および取得するために使用されるユーザ定義のセッション保存関数を設定する。 セッションデータをローカルデータベースに保存する場合のように、PHPセッションにより提供されるもの以外の保存方法を使用したい場合に有用。	true/false			PHP5.4以前で、オブジェクトをセッション保存ハンドラとして使うときには、register_shutdown_function() を使って session_write_close() を追加登録する方法がある。 PHP5.4以降で、SessionHandlerInterface でセッションハンドラを実装したり、SessionHandler でPHPの内部セッションハンドラを公開したりできる。
session_unset	全てのセッション変数を開放する	なし			\$SESSION スーパーグローバルを使用したセッション変数の登録が不可能になってしまうため、unset(\$SESSION) を使って \$SESSION を完全に unset しないでください。
session_is_registered	変数がセッションに登録されているかどうかを調べる	true/false	5.4で削除		
session_register	現在のセッションに1つ以上の変数を登録する	true/false	5.4で削除	○	
session_unregister	現在のセッションから変数の登録を削除する	true/false	5.4で削除		
session_status	現在のセッションの状態を返す	備考の状態参照	5.4以降		<ul style="list-style-type: none"> <li>•PHP_SESSION_DISABLED: セッションが無効</li> <li>•PHP_SESSION_NONE: ツションが有効だ、セッションなし</li> <li>•PHP_SESSION_ACTIVE: ツションが有効、セッションあり</li> </ul>
session_register_shutdown	session_write_close() をシャットダウン関数として登録する。	なし	5.4以降		
session_abort	Discard session array changes and finish session	なし	5.6以降		
session_reset	Re-initialize session array with original values	なし	5.6以降		
session_set_cookie_params	セッションクッキーパラメータを設定する。 (有効期間 [URLパス [ドメイン [セキュア [HTTP直接アクセスのフラグ]]]]) PHP の ini 設定の値を実行時に上書きします。値の取得には ini_get() が使えます。	なし		○	<ul style="list-style-type: none"> <li>"lifetime":クッキーの有効期限(秒数)</li> <li>"path":情報が保存されている場所のパス(/のみ指定でドメイン上のすべてのパスで動作)</li> <li>"domain":クッキーのドメイン(先頭にドットで、全てのサブドメインでクッキー有効:例 'sample.com')</li> <li>"secure":クッキーはセキュアな接続(true)でのみ送信</li> <li>"httponly":クッキーは HTTP を通してのみアクセス可能</li> </ul>
session_get_cookie_params	セッションクッキーのパラメータを得る。現在のセッションクッキーの情報を連想配列として返す。	備考の連想配列を参照			
setcookie	クッキーを送信する。 setcookie() はブラウザに何らかの出力を行う前にコールする必要がある (クッキーは HTTP ヘッダの一部のため、制約はheader() と同じ) setcookie ( \$name [, \$value [, \$expire= 0(整数) [, \$path [, \$domain [, \$secure = false(ブール) [, \$httponly = false(ブール) ]]]]] )	true/false			PHP5.2以降、httponly パラメータ追加 PHP5.5以降、クライアントに送出する Set-Cookie ヘッダに、Max-Age 属性も含めるようになる。
setrawcookie	値を URL エンコードせずにクッキーを送信する	true/false			
\$_SESSION['xxx']	セッション変数の追加・変更・削除と読み出し。 「xxx」のキーは、アンダースコアか英数字のみ				すべてのセッションデータの破棄は、「\$_SESSION=array();」を利用する
\$_COOKIE['xxx']	クッキー変数の読み出し				現在のスクリプトに HTTP クッキーから渡された変数の連想配列

※SessionHandler は特殊なクラスで、これを継承したクラスを作れば PHP が内部的に使っているセッション保存ハンドラを拡張できます。(PHP 5.4以降)