

第10回 データベース接続 (INSERT)

10. 新規登録

10-1. トランザクション処理

トランザクション管理については、データベースの授業で既に学んでいますので、ここでは PHP のコードの記述方法を記載します。

■ トランザクション管理メソッド

PDO のメソッド	意味
PDO::beginTransaction	トランザクションを開始する
PDO::commit	トランザクションを完了する
PDO::rollback	トランザクション開始前の状態に戻す

トランザクションの開始は、PDO::beginTransaction メソッドで行います。トランザクションは、PDO::commit メソッドもしくは PDO::rollback メソッドが呼ばれるまで継続します。

一連のデータベース処理を完了したいときは、PDO::commit メソッドでトランザクションの確定を行い、処理を取り消したいときは、PDO::rollback でトランザクションのロールバックを行ってください。



コミットもロールバックも行わなかったときは

トランザクションを開始したら、コミットするまではデータベース処理が確定されません。コミットもロールバックも行わなかった状態で、プログラムを終了すると、「ロールバックと同じ意味になります。」

また、トランザクションの状態を知るには、PDO::inTransaction メソッドで確認することができます。(トランザクションがアクティブであれば True、アクティブでなければ False が返ってきます。)

* 自動コミットモードの指定方法は、「8 回目の資料」で確認してください。

★Sample コード (INSERT)

Sample のデータベース情報 **(皆さんが使用する DB とは違うものを使用しています)**

- ・ ホスト名 : localhost
- ・ DB 名 : malldb
- ・ DB ユーザー : malluser
- ・ DB パスワード : mall
- ・ テーブル : products

【概要】

ひとつ前のページから POST 形式で「product_id」と「product_name」が送信された状態で、POST データを受け取り、処理をする一例です。

コードが長くなるため、次のページに記載します。

```

$dsn = "mysql:host=".DB_HOST.";dbname=malldb;charset=".DB_CHARSET;
try {
    $db = new PDO($dsn, "malluser", "mall");
    $db->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $db->setAttribute(PDO::ATTR_AUTOCOMMIT, false);

    // トランザクション開始

    $db->beginTransaction();
    $sql = "INSERT INTO PRODUCTS(PRODUCT_ID, PRODUCT_NAME)
VALUES(:product_id, :product_name)";

    $stmt = $db->prepare($sql);
    $stmt->bindParam(':product_id', $product_id, PDO::PARAM_INT);
    $stmt->bindParam(':product_name', $product_name,
PDO::PARAM_STR);

    $stmt->execute();
    // トランザクション確定

    $db->commit();

} catch(PDOException $poe) {
    $db->rollBack();
    // TODO:debug 用メッセージ(本番ではセキュリティ上表示しないこと!!)

    echo "DB 接続エラー". $poe->getMessage();
}finally{
    $stmt = null;
    $db = null;
}

```

【注意事項】



例外処理 catch 文に記載している処理について

```
「echo "DB 接続エラー" . $poe->getMessage();」
```

開発中にエラー内容が分かり易いように、**デバッグ用として出力**しています。

(本授業では以降の課題もデバッグ用コードを使用しますが、)

PDO のエラーメッセージを出力してしまうと、エラーメッセージから読み取れる情報(テーブルやカラム名など)が多いため、実稼働においてはセキュリティ上よろしくありません。**本番環境では画面出力はせず、ユーザーがアクセスできないサーバーのログファイルに書き込むなどの対策をとみましょう。**