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とautocommit

**■トランザクションとは？**

DBMSに対して複数のSQL文を送る場合に、1つ以上のSQL文をひとつの単位として扱うように指示することができる。このひとつの単位のことを「トランザクション」という。

複数のテーブルの内容を変更する処理（DML：insert、update、delete文）を一度に行う時、その途中で何らかのトラブルが発生する場合がある。そのようなトラブルを想定して、そのような複数の処理を行う場合は、すべての処理が成功した場合のみその変更を有効にするといった場合に、このトランザクションは必要となる。（ACID特性を調べてみてください）

**■MySQLにおけるautocommitモード**

MySQLでは、デフォルトの設定として、autocommit（自動更新モード）が「ON」になります。

したがって、DMLを実行した時点で、commitも実行されます。

ただし、データベース演習の環境では、autocommitモードが「OFF」に設定されているので、DMLを実行してもcommitを実行するまではその変更は確定されません。

どちらの設定をとるのかは、時と場合によりますが、プログラム内でトランザクション処理をする場合は、挙動が変わってきますので注意が必要です。

autocommitモードの設定の確認の仕方（MySQLプロンプトで）

select @@autocommit;

と入力し、０だったらOFF、１だったらONです。

**■MySQLのautocommitモードがOFFだった場合のトランザクション**

MySQLはデフォルトではautocommitはONなので、トランザクション処理をしようとする場合、「BEGIN」と「COMMIT（またはROLLBACK）」の間に必要な処理を記述します。

しかし、autocommitがOFFの場合は、BEGINでトランザクション処理をすることはもちろんできるが、その前にSQLを発行した場合（それが処理に関係のないSELECT文だったとしても）勝手にトランザクションが開始されるので、さらにBEGINを使ってしまうとエラーになるので注意する必要があります。

■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。ただし、今回はautocommitがOFFになっているので、使用しません。

　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。

|  |  |
| --- | --- |
| 警告 単色塗りつぶし | 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。 |

**■DMLのSQL実行結果について**

前回までの課題は、参照のSQLだったので、実行結果として、結果セットが返ってきました。

しかし、今回はINSERT文なので、結果セットはなく、実際に処理を行った件数が返却されます。

したがって、その返却された件数と、処理を行った件数が同じかどうかを確認して、OKであればCOMMIT、間違っていればROLLBACKを行う必要があります。

■DML実行の例（autocommit　OFFの前提なのでbeginTransactionは実行しない）

//プレースホルダーを使用したSQL文の作成

$sql = "INSERT INTO OLDPRODUCT VALUES(:product\_no, ＊＊＊＊以下略＊＊＊＊＊)";

//SQLの準備

$stmt = $db->prepare($sql);

//プレースホルダーのバインド

$stmt->bindParam('product\_no', $product\_no, PDO::PARAM\_INT);

＊＊＊＊以下略＊＊＊＊＊

//SQL実行

$stmt->execute();

//SQL実行結果件数の取得

$result["result"] = $stmt->rowCount();//直前のexecuteした件数

//挿入件数によってコミットかロールバックする

if ($result["result"] === 1) {

    // トランザクション確定

    $db->＊＊＊＊以下略＊＊＊＊＊

    $result["message"] = "データの登録に成功しました!\*\*\*\*\*\*。<br>";

}

//件数が違うのでロールバック

else {

    ＊＊＊＊以下略＊＊＊＊＊

    $result["message"] = "データの登録に失敗しました↓↓↓↓↓↓。<br>";;

}