

データベース連携

▼①接続テスト

sample11/connect フォルダを作成し、フォルダ内に下記ファイルを作成しましょう。

PDO_testdb.php

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8">
<title>PDOでデータベースに接続する</title>
<link href="../../css/style.css" rel="stylesheet">
</head>
<body>
<div>
  <?php
    // データベースユーザ
    $user = 'testuser';
    $password = 'pw4testuser';
    // 利用するデータベース
    $dbName = 'testdb';
    // MySQLサーバ
    $host = 'localhost:8889';
    // MySQLのDSN文字列
    $dsn = "mysql:host={$host};dbname={$dbName};charset=utf8";

    //MySQLデータベースに接続する
    try {
      $pdo = new PDO($dsn, $user, $password);
      // プリペアドステートメントのエミュレーションを無効にする
      $pdo->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
      // 例外がスローされる設定にする
      $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
      echo "データベース{$dbName}に接続しました。";
      // 接続を解除する
      $pdo = NULL;
    } catch (Exception $e) {
      echo '<span class="error">エラーがありました。</span><br>';
    }
  </?php>
</div>
</body>
</html>
```

```
        echo $e->getMessage();  
        exit();  
    }  
    ?>  
</div>  
</body>  
</html>
```

▼②フォーム入力データによりデータベースを検索

sample11 フォルダの下に form フォルダを作成し、下記ファイルを作成しましょう。

searchform.html

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8">
<title>名前検索</title>
<link href="../../css/style.css" rel="stylesheet">
</head>
<body>
<div>
  <!-- 入力フォームを作る -->
  <form method="POST" action="search.php">
    <ul>
      <li>
        <label>名前を検索します（部分一致） : <br>
        <input type="text" name="name" placeholder="名前を入れてください。">
        </label>
      </li>
      <li><input type="submit" value="検索する"></li>
    </ul>
  </form>
</div>
</body>
</html>
```

search.php

```
<?php
require_once("../../lib/util.php");
$gobackURL = "searchform.html";

// 文字エンコードの検証
if (!cken($_POST)){
    header("Location:{$gobackURL}");
    exit();
}

// nameが未設定、空のときはエラー
if (empty($_POST)){
    header("Location:searchform.html");
    exit();
} else if(!isset($_POST["name"])||($_POST["name"]=="")){
    header("Location:{$gobackURL}");
    exit();
}

// データベースユーザ
$user = 'testuser';
$password = 'pw4testuser';
// 利用するデータベース
$dbName = 'testdb';
// MySQLサーバ
$host = 'localhost:3306';
// MySQLのDSN文字列
$dsn = "mysql:host={$host};dbname={$dbName};charset=utf8";
?>

<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8">
<title>名前検索</title>
<link href="../../css/style.css" rel="stylesheet">
<!-- テーブル用のスタイルシート -->
```

```
<link href="../../../css/tablestyle.css" rel="stylesheet">
</head>
<body>
<div>
    <?php
    $name = $_POST["name"];
    //MySQLデータベースに接続する
    try {
        $pdo = new PDO($dsn, $user, $password);
        // プリペアドステートメントのエミュレーションを無効にする
        $pdo->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
        // 例外がスローされる設定にする
        $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        // SQL文を作る
        $sql = "SELECT * FROM member WHERE name LIKE(:name)";
        // プリペアドステートメントを作る
        $stm = $pdo->prepare($sql);
        // プレースホルダに値をバインドする
        $stm->bindValue(':name', "%{$name}%", PDO::PARAM_STR);
        // SQL文を実行する
        $stm->execute();
        // 結果の取得（連想配列で受け取る）
        $result = $stm->fetchAll(PDO::FETCH_ASSOC);
        if(count($result)>0){
            echo "名前に「{$name}」が含まれているレコード";
            // テーブルのタイトル行
            echo "<table>";
            echo "<thead><tr>";
            echo "<th>", "ID", "</th>";
            echo "<th>", "名前", "</th>";
            echo "<th>", "年齢", "</th>";
            echo "<th>", "性別", "</th>";
            echo "</tr></thead>";
            // 値を取り出して行に表示する
            echo "<tbody>";
            foreach ($result as $row){
                // 1行ずつテーブルに入れる
                echo "<tr>";
                echo "<td>", es($row['id']), "</td>";
                echo "<td>", es($row['name']), "</td>";
```

```
        echo "<td>", es($row['age']), "</td>";
        echo "<td>", es($row['sex']), "</td>";
        echo "</tr>";
    }
    echo "</tbody>";
    echo "</table>";
} else {
    echo "名前に「{$name}」は見つかりませんでした。";
}
} catch (Exception $e) {
    echo '<span class="error">エラーがありました。</span><br>';
    echo $e->getMessage();
}
?>
<hr>
<p><a href="<?php echo $gobackURL ?>">戻る</a></p>
</div>
</body>
</html>
```

▼③フォーム入力データをデータベースに登録

sample11/form フォルダの下に、下記ファイルを作成しましょう。

insertform.html

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8">
<title>名前検索</title>
<link href="../../css/style.css" rel="stylesheet">
</head>
<body>
<div>
  <!-- 入力フォームを作る -->
  <form method="POST" action="search.php">
    <ul>
      <li>
        <label>名前を検索します（部分一致）：<br>
        <input type="text" name="name" placeholder="名前を入れてください。">
        </label>
      </li>
      <li><input type="submit" value="検索する"></li>
    </ul><!DOCTYPE html>
</div>
<div>
  <!-- 入力フォームを作る -->
  <form method="POST" action="insert_member.php">
    <ul>
      <li>
        <label>名前：
        <input type="text" name="name" placeholder="名前">
        </label>
```

```
</li>
<li>
  <label>年齢 :
  <input type="number" name="age" placeholder="半角数字">
</label>
</li>
<li>性別 :
  <label><input type="radio" name="sex" value="男" checked>男性</label>
  <label><input type="radio" name="sex" value="女">女性</label>
</li>
<li><input type="submit" value="追加する"></li>
</ul>
</form>
</div>
</body>
</html>
</form>
</div>
</body>
</html>
```


insert_member.php

```
<?php
require_once("../../lib/util.php");
$gobackURL = "insertform.html";

// 文字エンコードの検証
if (!cken($_POST)){
    header("Location:{$gobackURL}");
    exit();
}

// 簡単なエラー処理
$errors = [];
if (!isset($_POST["name"])|!( $_POST["name"]=="")){
    $errors[] = "名前が空です。";
}
if (!isset($_POST["age"])|!( ctype_digit($_POST["age"]))) {
    $errors[] = "年齢には数値を入れてください。";
}
if (!isset($_POST["sex"])|!(in_array($_POST["sex"], ["男","女"]))) {
    $errors[] = "性別が男または女ではありません。";
}

//エラーがあったとき
if (count($errors)>0){
    echo '<ol class="error">';
    foreach ($errors as $value) {
        echo "<li>", $value , "</li>";
    }
    echo "</ol>";
    echo "<hr>";
    echo "<a href=", $gobackURL, ">戻る</a>";
    exit();
}

// データベースユーザ
$user = 'testuser';
```

```
$password = 'pw4testuser';
// 利用するデータベース
$dbName = 'testdb';
// MySQLサーバ
$host = 'localhost:3306';
// MySQLのDSN文字列
$dsn = "mysql:host={$host};dbname={$dbName};charset=utf8";
?>

<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8">
<title>レコード追加</title>
<link href="../../../css/style.css" rel="stylesheet">
<!-- テーブル用のスタイルシート -->
<link href="../../../css/tablestyle.css" rel="stylesheet">
</head>
<body>
<div>
  <?php
    $name = $_POST["name"];
    $age = $_POST["age"];
    $sex = $_POST["sex"];
    //MySQLデータベースに接続する
    try {
      $pdo = new PDO($dsn, $user, $password);
      // プリペアドステートメントのエミュレーションを無効にする
      $pdo->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
      // 例外がスローされる設定にする
      $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

      // SQL文を作る
      $sql = "INSERT INTO member (name, age, sex) VALUES (:name, :age, :sex)";
      // プリペアドステートメントを作る
      $stm = $pdo->prepare($sql);
      // プレースホルダに値をバインドする
      $stm->bindValue(':name', $name, PDO::PARAM_STR);
      $stm->bindValue(':age', $age, PDO::PARAM_INT);
      $stm->bindValue(':sex', $sex, PDO::PARAM_STR);
```

```

// SQL文を実行する
if ($stm->execute()){
    // レコード追加後のレコードリストを取得する
    $sql = "SELECT * FROM member";
    // プリペアドステートメントを作る
    $stm = $pdo->prepare($sql);
    // SQL文を実行する
    $stm->execute();
    // 結果の取得（連想配列で受け取る）
    $result = $stm->fetchAll(PDO::FETCH_ASSOC);
    // テーブルのタイトル行
    echo "<table>";
    echo "<thead><tr>";
    echo "<th>", "ID", "</th>";
    echo "<th>", "名前", "</th>";
    echo "<th>", "年齢", "</th>";
    echo "<th>", "性別", "</th>";
    echo "</tr></thead>";
    // 値を取り出して行に表示する
    echo "<tbody>";
    foreach ($result as $row) {
        // 1行ずつテーブルに入れる
        echo "<tr>";
        echo "<td>", es($row['id']), "</td>";
        echo "<td>", es($row['name']), "</td>";
        echo "<td>", es($row['age']), "</td>";
        echo "<td>", es($row['sex']), "</td>";
        echo "</tr>";
    }
    echo "</tbody>";
    echo "</table>";
} else {
    echo '<span class="error">追加エラーがありました。</span><br>';
};
} catch (Exception $e) {
    echo '<span class="error">エラーがありました。</span><br>';
    echo $e->getMessage();
}
?>
<hr>

```

```
<p><a href="<?php echo $gobackURL ?>">戻る</a></p>
</div>
</body>
</html>
```