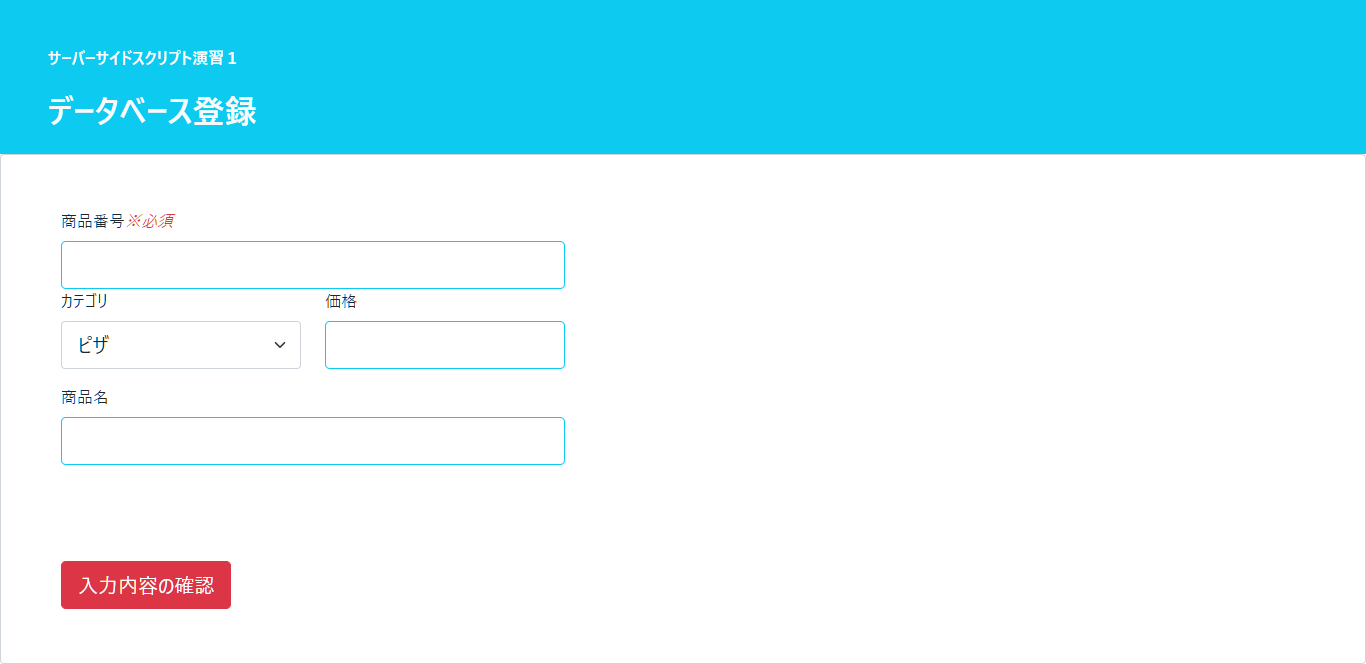
第１０回　データベース接続（INSERT）

★課題10－１　（ファイル名:kadai10\_1.php,kadai08\_1.php）

①　課題8\_1の新規登録ボタンの遷移先をkadai10\_1.phpに指定しましょう。

②　入力画面から、「入力内容の確認」ボタン押下時に、次のページに遷移するように設定しましょう。



■データ形式：POST

■送信先：kadai10\_2.php

★課題１０－２　第1段階（ファイル名：kadai10\_2.php）

DB登録処理を行いましょう。

1. 入力画面より送られたデータ（4項目）を変数に格納する。

filter\_input関数を使用する。カテゴリと値段は数値フィルタを設定する。

1. DB接続処理
2. SQL文の準備と実行　←ここが前回とは最も異なる部分です。（講義資料参照）
3. DB切断処理
4. 結果を画面に表示しましょう。

$result["message"]の内容をHTML内に表示する。

★課題１０－２　第２段階（ファイル名：kadai10\_2.php）

DB接続をして登録処理をする前に、入力値のチェックを行いましょう。

各入力値にチェックを追加してください。

エラーとなる場合は、$result["status"]にfalse、$result["message"]にエラーメッセージを設定しましょう。

■商品番号

　4桁数値であるかのチェック

　preg\_match("/^([0-9]{4})$/",$変数名)

　preg\_match関数で数値かどうかのチェックを行えます。

**※preg\_match関数および正規表現についての説明は別紙参照。**

　第1引数が正規表現です。数値かつ4桁を表しています。

　第2引数が調べたい文字列です。

　戻り値：マッチしていればtrue、マッチしていなければfalse

■商品名（空白チェック、商品名にスペースのみが入力された場合は空白とする）

　全角もしくは半角の空白があった場合、空にしましょう。

　str\_replace（置き換え対象となる文字、置き換え文字、処理をしたい文字列）

　変換後、空であれば、エラー扱いとしてください。

■価格

　数値である（filter\_input関数で取得する際に、型でフィルターを掛ける）かつ、空でない、の条件を満たさない場合は、エラー扱いとする

■カテゴリ

　数値を文字列に変換する。入力された値が１だった場合は「ピザ」、２だった場合は「ドリンク」とする。

３．データ挿入は、入力値にエラーがなかった場合のみ、行うようにしましょう。

★課題１０－２　第３段階（ファイル名：kadai10\_2.php）

DB登録の前に、既に同じ商品番号が登録されていないかを調べましょう。

OLDPRODUCTテーブルのPRODUCT\_NOは主キーに設定されているので、存在するPRODUCT＿NOで挿入しようとするとSQL実行時にエラーとなり、例外発生する。

例外処理として行うことも可能であるが、挿入処理前に存在チェックして、存在したらINSERTを行わないというロジックを行ってもよい（今回はこちら）。

＜＜実行例＞＞

//入力された商品番号のレコードがあるかチェックするSQL(LIMIT句はパフォーマンスのため今回はなくてもOK)

    $sqlSelect = "SELECT count(PRODUCT\_NO) FROM OLDPRODUCT WHERE PRODUCT\_NO = :product\_no LIMIT 1";

//SQLの準備とバインド

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

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

//SQLの実行

$stmt->execute();

//SQLの実行結果$countの中に上記SQLに実行結果がセットされているので、その内容でINSERTするかどうか

//処理を振り分ける

$count = $stmt->fetchColumn();