

SAÉ 3.01 - DEVAPP

Dossier de conception PHP

Groupe 2B11



1 - Administration	2
1) GestionContact.php	2
2) consultMessage.php	4
3) AnswerTo.php	6
2 - Affichage des produits en fonction du prix	7
3 - Ajout d'un produit au panier	8
4 - Gestion de ses paniers	12
5 - Contrôle des saisies	17
6 - Insertion des données	18

1 - Administration

1) GestionContact.php

Dans cette page, nous faisons afficher un tableau avec les messages envoyés depuis la page contact du site et qui ont été enregistrés dans notre base de donnée :

```
<input type="submit" value="Retour"
onclick="document.location.href='../indexAdmin.php'">
```

```
<div class="recherche">
```

```
<form method='POST' action="./GestionContact.php">
```

```
<div class="icon"><i class="fa-sharp fa-solid
fa-magnifying-glass"></i></div>
```

```
</form>
```

```
</div>
```

```
<?php
```

```
require_once "../includeAdmin/connect.inc.php"; //connexion à la
base de donnée
```

```
$mssgQ = 'SELECT * FROM MESSAGERIE ORDER BY IDMESSAGE'; //selection
des messages ordonné selon leur identifiant
```

```
$messages = oci_parse($connect, $mssgQ); //liaison entre la base de
donnée et la requête
```

```

$result = oci_execute($messages); // exécution de la requête

if($result == false){ // si l'exécution ne renvoie rien alors on
affiche un petit message d'erreur

echo 'Aucun message';

}

?>

<div class="table container">

<div class="tableMess table">

<table class='table table-hover' border="2">

<!-- création d'un tableau qui va contenir toutes le nom de
l'expéditeur, l'objet du message, la date de l'envoi et le statut de
lecture du message -->

<thead>

<tr>

<th>De :</th>

<th>Objet :</th>

<th>Envoyé le:</th>

<th>Lu :</th>

</tr>

</thead>

<tbody>

<?php while ( ($mess = oci_fetch_assoc($messages)) !=false) { //on boucle
pour avoir tous les messages enregistrés dans la bae de donnée ?>

<tr>

<td><?=$mess['NOMMESSAGE'] ?></td> <!--Nom de l'expéditeur-->

```

```

<td><?='<a
href="./consultMessage.php?pidMess=' . $mess['IDMESSAGE'] . '">' . $mess
['OBJETMESSAGE'] . '</a>' ?></td><!--Objet du message sous forme de lien
qui redirige vers

la page de consultation du message en mettant en paramètre l'identifiant
du message-->

<td><?=$mess['DATEMESSAGE'] ?></td><!--date d'envoi-->

<td><?=$mess['MESSAGELU'] ?></td><!--statut de lecture du message-->

</tr>

<?php } //fin boucle while ?>

</tbody>

</table>

</div>

</div>

```

2) consultMessage.php

Cette page permet la consultation en détail d'un message sélectionné parmi ceux envoyés depuis la page de contact :

```

<?php

require_once "../includeAdmin/connect.inc.php"; //connexion à la
base de donnée

//Mise a jour du statu de lecture du message ouvert

$luQ = "UPDATE MESSAGERIE SET MESSAGELU='o' WHERE
IDMESSAGE='" . $_GET['pidMess'] . "'";

$lu = oci_parse($connect,$luQ);

$res = oci_execute($lu);

oci_commit($connect);

//Selection des dnnées associés au message ouvert

```

```

$mssgQ = "SELECT * FROM MESSAGERIE WHERE
IDMESSAGE='".$$_GET['pidMess']."'";

$messages = oci_parse($connect, $mssgQ);

$result = oci_execute($messages);

if($result == false){ // si l'exécution ne renvoie rien alors on
affiche un petit message d'erreur

echo 'Aucun message';

}

$infoMess = oci_fetch_assoc($messages);

$contenu=html_entity_decode($infoMess['CONTENUMESSAGE']);
//décodage du contenu du message enregistré dans la base de donnée

?>

<fieldset>

<form
action="/~SAESYS11/admin/pagesAdmin/AnswerTo.php?pidMess=$infoMess
['IDMESSAGE']" method="post"><!--Formulaire qui permet la lecture
du message et de ces infomations, redirige vers une page de
réponse-->

<label for="#">De :</label> <br>

<input type="text" name="nom" value="<?=$infoMess['NOMMESSAGE']."
".$infoMess['PRENOMMESSAGE'] //champ avec le nom et prénom de
l'expéditeur?" readonly> <br>

<label for="#">E-mail :</label> <br>

<input type="email" name="email"
value="<?=$infoMess['MAILMESSAGE'] //champ avec le mail de
l'expéditeur?" readonly> <br>

<label for="#">Objet :</label> <br>

```

```

<input type="text" name="objet"
value="<?=$infoMess['OBJETMESSAGE'] //champ avec l'objet du
message'?" readonly> <br>

<label for="#">Message :</label> <br>

<textarea name="contenu" cols="39" rows="4" maxlength="4000"
type="text" readonly><?=$contenu //champ avec le contenu du
message?></textarea> <br>

<input type="submit" name="answer" value="Répondre">

</form>

<input type="submit" value="Retour"
onclick="document.location.href='./GestionContact.php'"><!--bouton
de retour à la page de gestion des messages-->

</fieldset>

```

3) AnswerTo.php

L'objectif de cette page est de permettre la réponse des administrateurs aux clients. Cependant, elle n'est pas vraiment fonctionnelle puisqu'il faudrait permettre de gérer la connexion à un serveur mail externe :

```

<?php

//suite à la redirection depuis la page de lecture du message on
reprends les infos nécessaires enregistré dans le fomulaire
précédent

$person = $_POST['nom'];

$mail = $_POST['email'];

$obj = $_POST['objet'];

?>

<fieldset>

<form action="" method="post">

<label for="#">Répondre à :</label> <br>

```

```

<input type="text" name="nom" value="<?=$person //champ avec le
nom et prénom de l'expéditeur?" readonly> <br>

<label for="#">E-mail :</label> <br>

<input type="email" name="email" value="<?=$mail //champ avec le
mail de l'expéditeur?" readonly> <br>

<label for="#">Objet :</label> <br>

<input type="text" name="objet" value="<?="RE : ".$obj //champ
avec l'objet du message?" readonly> <br>

<label for="#">Message :</label> <br>

<textarea name="contenu" cols="39" rows="4" maxlength="4000"
type="text"></textarea> <br><!--champ où écrire le contenu de la
réponse-->

<input type="submit" name="answer" value="Répondre">

</form>

<input type="submit" value="Retour"
onclick="document.location.href='./GestionContact.php'"><!--bouton
de retour à la page de gestion des messages-->

</fieldset>

```

2 - Affichage des produits en fonction du prix

Tout ceci est une mise à jour du fonctionnement de la page produit.php.

Tout d'abord nous avons mis en place un champ caché dans le formulaire qui récupère le choix de l'utilisateur.

```

<body>
<?php include "../include/header.php"; ?>
<main class="produits">
  <div class="produit_header">
    <div class="title">Votre recherche : <?php echo $_GET['recherche'] ?></div>
    <div class="input_trie">
      <form>
        <select name="choix" id="optPrix" class="slct" value="Trier par" onchange="window.location.href='produit.php?recherche=<?php
        echo htmlspecialchars($_GET['recherche'], ENT_QUOTES, 'UTF-8'); ?>&tri='+this.value'">
          <option value="ASC" <?php if (!isset($_GET['tri']) || ($_GET['tri'] == 'ASC' && ctype_alnum($_GET['tri']))) {
            echo 'selected';
          } ?>>Prix croissant</option>
          <option value="DESC" <?php if (!isset($_GET['tri'])) {
            echo '';
          } else if ($_GET['tri'] == 'DESC' && ctype_alnum($_GET['tri'])) {
            echo 'selected';
          } ?>>Prix décroissant</option>
        </select>
      </form>
    </div>
  </div>
  <div class="product_container">

```

Ensuite nous mettons ce choix dans une variable \$tri en php pour pouvoir l'utiliser dans les différentes requêtes d'affichage des produits afin d'afficher les produits en fonction du prix, soit croissant soit décroissant.

```

$allowed_values = array('ASC', 'DESC');
$tri = isset($_GET['tri']) && in_array(strtoupper(trim($_GET['tri'])), $allowed_values) ? strtoupper(trim($_GET['tri'])) : 'ASC';

$recherche = filter_var($_GET['recherche'], FILTER_SANITIZE_STRING);

switch ($recherche) {
  // CATÉGORIES MÈRES
  case 'Produit':
    $query = "
      select *
      from produit
      order by prixProduit $tri
    ";
    break;
  case 'Aliments':
    $query = "
      select *
      from produit
      where idcategorie in ('1', '01', '02', '03', '04', '05', '06', '001', '002', '003', '004', '005', '006')
      order by prixProduit $tri
    ";
    break;
  case 'Hygiène':
    $query = "
      select *
      from produit
      where idcategorie in ('2', '07', '08', '09', '007', '008', '009')
      order by prixProduit $tri
    ";
    break;
  case 'Entretien':
    $query = "
      select *
      from produit

```

```

      case 'Menage':
        $query = "
          select *
          from produit
          where idcategorie in ('12', '010', '13', '14', '15', '16', '17', '18', '19')
          order by prixProduit $tri
        ";
        break;
      case 'Décoration':
        echo "développement en cours";
        break;

      // SOUS CATÉGORIES FILLES

      // COMPORTEMENT PAR DÉFAUT :
      default:
        $query = "
          select *
          from produit
          where lower(nomproduit) LIKE lower('%$recherche%')
          order by prixProduit $tri
        ";
    }

```


3 - Ajout d'un produit au panier

Si le client est connecté alors avec le code suivant on récupère son id:

```
if (!isset($_SESSION['access']) && $_SESSION['access'] == 'oui') {
    $nc = filter_var($_SESSION['nomClient'], FILTER_SANITIZE_STRING);
    $pc = filter_var($_SESSION['prenomClient'], FILTER_SANITIZE_STRING);

    $query = "SELECT * FROM CLIENTS WHERE NOMCLIENT = :nc AND PRENOMCLIENT = :pc";

    $lesclients = oci_parse($connect, $query);
    oci_bind_by_name($lesclients, ':nc', $nc);
    oci_bind_by_name($lesclients, ':pc', $pc);
    $result = oci_execute($lesclients);

    if ($result === false) {
        echo 'Aucun client';
    }

    $infoClients = oci_fetch_assoc($lesclients);
    if ($infoClients) {
        $idClient = $infoClients['IDCLIENT'];
    } else {
        $idClient = 0;
    }
} else {
    $idClient = 0;
}
```

Ensuite, en plus de l'affichage des produits, nous mettons dans des champs cachés l'id du produit, l'id du client, l'url et le nom de la fonction que l'on veut utiliser, ici ajoutAuPanier.

```
echo '
<div class="product_content">
    <div class="product_card">
        <div class="product_img">
            
        </div>
        <div class="product_title">' . $leProduit['NOMPRODUIT'] . '</div>
        <div class="product_data">
            <div class="product_description">
                ' . $leProduit['DESCRIPTIONPRODUIT'] . '
            </div>
            <div class="product_prix">
                prix au kg : <br />
                <span class="prix">' . $leProduit['PRIXPRODUIT'] . ' €</span>
            </div>
        </div>
        <div class="product_btn">
            <div class="btn_add btn">
                <form action="/~SAESYS11/include/AjoutAuPanier.php" method="post">
                    <input type="hidden" name="idProduit" value="' . $leProduit['IDPRODUIT'] . '">
                    <input type="hidden" name="idClient" value="' . $idClient . '">
                    <input type="hidden" name="fonction" value="ajoutAuPanier">
                    <input type="hidden" name="url" value="?recherche=' . $_GET['recherche'] . '&tri=' . $_GET['tri'] . '">
                    <button type="submit">Ajouter au panier</button>
                </form>
            </div>
            <div class="btn_info btn">
                <button class="modal-open" data-modal="modal" . $i . '">En savoir plus</button>
            </div>
        </div>
    </div>
</div>
```

Lorsque l'on clique sur le bouton "Ajouter au panier" cela envoie vers la page contenant la fonction d'ajout au panier. Cette page fonctionne de la manière suivante:

Si la variable "fonction" est définie dans la demande POST, le script utilise une instruction switch pour déterminer quelle fonction doit être appelée. Dans ce cas, lorsque la variable "fonction" est définie sur "ajoutAuPanier", la fonction ajoutAuPanier() est appelée. La fonction ajoutAuPanier() prend en argument deux arguments: l'ID du produit et l'ID du client. Il inclut d'abord le fichier connect.inc.php, qui établit une connexion à la base de données. Si l'utilisateur n'est pas connecté, la fonction le redirige vers la page de connexion.

Si l'utilisateur est connecté, elle effectue une validation et une sanitization des données d'entrée (ID produit et ID client) passées en argument. Si l'entrée est invalide, elle appelle la fonction showAlert(), qui affiche un message d'erreur et redirige l'utilisateur vers la page des produits. Si l'entrée est valide, la fonction construit alors une requête et utilise les fonctions oci_* pour exécuter une procédure stockée dans la base de données Oracle qui ajoute le produit au panier avec l'ID du produit et l'ID du client fournis. Ensuite, elle appelle la fonction showAlert() pour afficher un message de succès et rediriger l'utilisateur vers la page des produits. En résumé, ce script permet l'ajout d'un produit au panier pour un utilisateur connecté en utilisant une procédure stockée dans une base de données Oracle.

Contrôle des champs

```

<?php
session_start();

if(isset($_POST['fonction'])){
    switch ($_POST['fonction']) {
        case 'ajoutAuPanier':
            ajoutAuPanier($_POST['idProduit'],$_POST['idClient']);
            break;
    }
}

function showAlert($msg) {
    echo "<script>alert('$msg');</script>";
    header("Refresh: 0; URL=~SAESYS11/pages/produit.php".$_POST['url']);
    exit();
}

function ajoutAuPanier($idProduit, $idClient) {
    require_once("connect.inc.php");
    if (!isset($_SESSION['access']) || $_SESSION['access'] != 'oui') {
        header('Location: ../../SAESYS11/pages/connexion.php');
        exit();
    } else {
        // Validate and sanitize user input
        $idProduit = filter_var($idProduit, FILTER_VALIDATE_INT);
        $idClient = filter_var($idClient, FILTER_VALIDATE_INT);

        // Check if input is valid
        if ($idProduit === false || $idClient === false) {
            showAlert("Erreur : entrée non valide");
        } else {
            try {
                $query = "begin
                    ADD_TO_CART(
                        P_ID_PANIER => :p_id_panier,
                        P_ID_CLIENT => :p_id_client,
                        P_ID_PRODUIT => :p_id_produit
                    );
                end;";

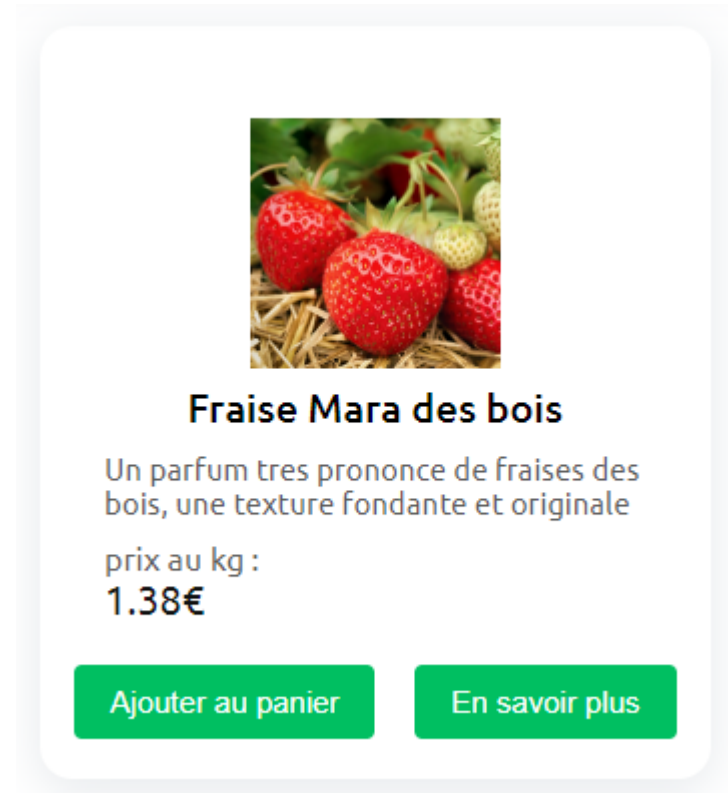
                $stid = oci_parse($connect, $query);
                oci_bind_by_name($stid, ':p_id_panier', $idClient);
                oci_bind_by_name($stid, ':p_id_client', $idClient);
                oci_bind_by_name($stid, ':p_id_produit', $idProduit);
                oci_execute($stid);
                oci_free_statement($stid);
                oci_close($connect);

                showAlert("Produit ajouté au panier !");
            } catch (PDOException $e) {
                echo $e->getMessage();
            }
        }
    }
}
}

```

4 - Gestion de ses paniers

Les utilisateurs s'il sont connectés ont la possibilité d'ajouter des produits dans leur panier. Pour ce faire, il doivent cliquer sur le bouton "Ajouter au panier" accessible sur les différentes cartes de produit sur la page "produit.php" :



Pour accéder à son panier; l'utilisateur doit cliquer sur l'onglet de panier disponible sur l'entête du site :



Si celui-ci possède plusieurs paniers, cette page affiche diffèrent liens cliquable redirigeant vers le panier choisi (le style de cette page est en cours de développement) :

Vos paniers :

[Panier n°75](#)[Panier n°88](#)

Détail de votre panier



Article : Huile demaquillante
toute douce lait d'amande
100ml

Prix : 8.8 €

[Supprimer](#)

Récapitulatif

4 article(s)	19.82 €
Frais de livraison	Gratuit
<hr/>	
Total panier	19.82 €

[Passer la commande](#)

Pour cela, nous faisons appel à la base de données dans un premier temps pour récupérer le nombre total de panier que possède le client et nous stockons le résultat dans une variable sous forme de integer :

```

$idClient = $_SESSION['idClient'];

// On recherche le nombre de panier du client
$query = "SELECT COUNT(IDPANIER) AS nbe_panier FROM PANIER WHERE IDCLIENT = '$idClient'";
$nbePanier = oci_parse($connect, $query);
$result = oci_execute($nbePanier);

if (!$result) {
    $e = oci_error($nbePanier); // on récupère l'exception liée au pb d'exécution de la requête
    print htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);

    // Redirection avec message d'erreur
    // header('location: /~SAESYS11/pages/connexion.php?error=Erreur connexion ... Réessayer');
    exit();
}

$nbePanier = oci_fetch_assoc($nbePanier);

$nbePanier = intval($nbePanier['NBE_PANIER']);

```

Ensuite; dans trois conditions nous vérifions si le nombre de paniers du client est soit < 1, soit égale à 1 soit supérieur à 1 pour faire des affichages différents :

```

if ($nbePanier < 1) {
    echo "<h6>Vous n'avez aucun panier</h6>";
}

```

```

    echo "<h6>Vous n'avez aucun panier</h6>";
} else if ($nbePanier == 1) {
?>
    <div class="container">
        <form action="#" method="post">
            <div class="left">
                <div class="title">Détail de votre panier</div>
                <div class="produits">
                    <div class="produit">
                        <div class="produit-img">
                            
                        </div>
                        <div class="produit-infos">
                            <div class="produit-title">Article : Aile de poulet</div>
                            <div class="produit-prix">Prix : 13.99 €</div>
                        </div>
                        <div class="produit-btn">
                            <input type="number" value="1" min="1">
                            <input type="button" class="btn" value="Supprimer">
                            <!-- <button class=" btn" onclick="console.log('Click');">Qu
                            <button class="btn" onclick="document.location.href='/~SAESY
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        <div class="right">
            <div class="title">Récapitulatif</div>
            <div class="content">
                <div class="recap">
                    <div class="recap-content">
                        <div class="recap-li">
                            <div class="recap-text">4 articles</div>
                            <div class="recap-data">1533.72 €</div>
                        </div>
                        <div class="recap-li">
                            <div class="recap-text">Frais de livraison</div>
                            <div class="recap-data">Gratuit</div>
                        </div>
                    </div>
                </div>
                <div class="commander">
                    <div class="commander-content">
                        <div class="commander-text">Total panier</div>
                        <div class="commander-data">1539.72 €</div>
                    </div>
                    <input type="submit" value="Passer la commande" class="commander-btn">
                    <!-- <a href="#" class="commander-btn">Passer la commande</a -->
                </div>
            </div>
        </form>
    </div>
<?php

```

Pour la vérifications du nombre de paniers du client > 1, nous faisons appel à la base de données pour récupérer les informations des paniers du client pour stocker les IdPanier dans un tableau ce qui nous permettra d'afficher et de redirigez l'utilisateur vers le bon panier notamment en utilisant la variable \$_GET dans laquelle nous ajoutons à l'url une clé panier qui prend comme valeur l'Id du panier choisi :

```

// les paniers du client
$query = "SELECT * FROM PANIER WHERE IDCLIENT = '$idClient'";
$lespaniers = oci_parse($connect, $query);
$result = oci_execute($lespaniers);

if (!$result) {
    $e = oci_error($lespaniers); // on récupère l'exception liée au pb d'exécution de la requête (violation F
    print htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);

    // Redirection avec message d'erreur
    // header('location: /~SAESYS11/pages/connexion.php?error=Erreur connexion ... Réessayer');
    exit();
}

$idpaniers = [];

while (($lepanier = oci_fetch_assoc($lespaniers)) != false) {
    // echo '<pre>' . print_r($lepanier, true) . '</pre>';
    $idpaniers[] = $lepanier['IDPANIER'];
}

for ($i = 0; $i < count($idpaniers); $i++) {
    echo "<a href='/~SAESYS11/pages/affichagePanier.php?panier=$idpaniers[$i]>Panier n°$idpaniers[$i]</a>";
}

// echo '<pre>' . print_r($idpaniers, true) . '</pre>';
}

```


5 - Contrôle des saisies

```
// Déclaration des expressions régulières pour la validation des champs
$nomCarteRegex = '/^[a-zA-Z]+$/' ;
$prenomCarteRegex = '/^[a-zA-Z]+$/' ;
$numCarteRegex = '/^[0-9]{16}$/' ;
$cryptogrammeRegex = '/^[0-9]{3}$/' ;
$nomClientRegex = '/^[a-zA-Z]+$/' ;
$villeRegex = '/^[a-zA-Z]+$/' ;
$codePostalRegex = '/^[0-9]{5}$/' ;
$adresseRegex = '/^[a-zA-Z0-9\s,#+-]*$/' ;
$telephoneRegex = '/^[0-9]{10}$/' ;

// Initialisation d'un tableau d'erreurs vide
$errors = array();

// Récupération des données du formulaire
$nomCarte = htmlspecialchars($_POST['nomCarte'], ENT_QUOTES, 'UTF-8');
$prenomCarte = htmlspecialchars($_POST['prenomCarte'], ENT_QUOTES, 'UTF-8');
$numCarte = htmlspecialchars($_POST['numCarte'], ENT_QUOTES, 'UTF-8');
$dateExpiration = htmlspecialchars($_POST['dateExpiration'], ENT_QUOTES, 'UTF-8');
$cryptogramme = htmlspecialchars($_POST['cryptogramme'], ENT_QUOTES, 'UTF-8');
$nomClient = htmlspecialchars($_POST['nomClient'], ENT_QUOTES, 'UTF-8');
$ville = htmlspecialchars($_POST['ville'], ENT_QUOTES, 'UTF-8');
$codePostal = htmlspecialchars($_POST['codePostal'], ENT_QUOTES, 'UTF-8');
$adresse = htmlspecialchars($_POST['adresse'], ENT_QUOTES, 'UTF-8');
$telephone = htmlspecialchars($_POST['telephone'], ENT_QUOTES, 'UTF-8');
```

```
// Validation des champs
if(!preg_match($nomCarteRegex, $nomCarte)) {
    $errors[] = "Le nom du titulaire de la carte n'est pas valide";
}
if(!preg_match($prenomCarteRegex, $prenomCarte)) {
    $errors[] = "Le prénom du titulaire de la carte n'est pas valide";
}
if(!preg_match($numCarteRegex, $numCarte)) {
    $errors[] = "Le numéro de la carte n'est pas valide";
}
if(!preg_match($cryptogrammeRegex, $cryptogramme)) {
    $errors[] = "Le cryptogramme n'est pas valide";
}
if(!preg_match($nomClientRegex, $nomClient)) {
    $errors[] = "Le nom du client n'est pas valide";
}
if(!preg_match($villeRegex, $ville)) {
    $errors[] = "Le nom de la ville n'est pas valide";
}
if(!preg_match($codePostalRegex, $codePostal)) {
    $errors[] = "Le code postal n'est pas valide";
}
if(!preg_match($adresseRegex, $adresse)) {
    $errors[] = "L'adresse n'est pas valide";
}
if(!preg_match($telephoneRegex, $telephone)) {
    $errors[] = "Le numéro de téléphone n'est pas valide";
}
if(strtotime($dateExpiration) < time()){
    $errors[] = "Date d'expiration non valide";
}
// Si il y a des erreurs, affichage du message d'erreur
if(!empty($errors)) {
    echo "Erreurs de validation : <br>";
    foreach($errors as $error) {
        echo $error . "<br>";
    }
}
```

Ce code est destiné à valider les informations saisies dans un formulaire de paiement. Il définit des expressions régulières pour chaque champ du formulaire (nom et prénom du titulaire de la carte, numéro de la carte, date d'expiration, cryptogramme, nom du client, ville, code postal, adresse, numéro de téléphone) pour vérifier si les données entrées respectent un format spécifique.

Il récupère ensuite les données saisies dans le formulaire en utilisant `$_POST['nomDuChamp']` et les stock dans des variables. Il vérifie ensuite si les données entrées correspondent aux expressions régulières définies. Si une donnée ne respecte pas l'expression régulière, un message d'erreur est ajouté dans un tableau d'erreurs. En fin, il vérifie si la date d'expiration est supérieur à la date actuelle.

Si le tableau d'erreurs est vide, cela signifie que toutes les données sont valides et on peut continuer le traitement, sinon, on peut afficher les erreurs pour que l'utilisateur corrige les informations incorrectes.

6 - Insertion des données

```
$query = "SELECT MAX(IDPAIEMENT) as MAX_ID FROM PAIEMENT";
$stmt = oci_parse($connect, $query);
oci_execute($stmt);
$row = oci_fetch_array($stmt, OCI_ASSOC);
$max_id = $row['MAX_ID'];
oci_free_statement($stmt);

$query = "INSERT INTO PAIEMENT (IDPAIEMENT, TYPEPAIEMENT) VALUES (:max_id+1, 'Carte Bleu')";
$stmt = oci_parse($connect, $query);
oci_bind_by_name($stmt, ':max_id', $max_id);
$r = oci_execute($stmt);
oci_free_statement($stmt);

$dateinvalidite = date('Y-m-d', strtotime($dateExpiration));
$query = "INSERT INTO DETAILPAIEMENT (IDCLIENT, IDPAIEMENT, NUMEROCARTE, DATEINVALIDITE, CRYPTOGRAMMECARTE)
VALUES (:idclient, :idpaiement+1, :numerocarte, TO_DATE(:dateinvalidite, 'yyyy-mm-dd'), :cryptogrammecarte)";
$stmt = oci_parse($connect, $query);
oci_bind_by_name($stmt, ':idclient', $idcli);
oci_bind_by_name($stmt, ':idpaiement', $max_id);
oci_bind_by_name($stmt, ':numerocarte', $numCarte);
oci_bind_by_name($stmt, ':dateinvalidite', $dateExpiration);
oci_bind_by_name($stmt, ':cryptogrammecarte', $cryptogramme);
$r = oci_execute($stmt);

$query = "SELECT MAX(TO_NUMBER(IDCOMMANDE)) as max_number FROM COMMANDE";
$stmt = oci_parse($connect, $query);
oci_execute($stmt);
$row = oci_fetch_array($stmt, OCI_ASSOC);
oci_free_statement($stmt);
```

```
$max_idcom = $row['MAX_NUMBER'];

$datecommande = date('Y-m-d', strtotime($datecommande));

$datecommande = date('Y-m-d');
$idcommande = $max_idcom + 1;
$idpaiement = $max_id+1;
$fraislivraison = number_format($fraislivraison, 2, '.', '');
$query = "INSERT INTO COMMANDE (IDCOMMANDE, IDPAIEMENT, IDCLIENT, DATECOMMANDE, FRAISLIVRAISON, ADRESSELIVRAISON, NOMLIVRAISON, CODEPOSTALLIVRAISON, VILLELIVRAISON, TELEPHONELIVRAISON)
VALUES (:idcommande, :idpaiement, :idclient, TO_DATE(:datecommande, 'yyyy-mm-dd'), :fraislivraison, :adressedlivraison, :nomlivraison, :codepostallivraison, :villelivraison, :telephonelivraison)";
$stmt = oci_parse($connect, $query);
oci_bind_by_name($stmt, ':idcommande', $idcommande);
oci_bind_by_name($stmt, ':idpaiement', $idpaiement);
oci_bind_by_name($stmt, ':idclient', $idcli);
oci_bind_by_name($stmt, ':datecommande', $datecommande);
oci_bind_by_name($stmt, ':fraislivraison', $fraislivraison);
oci_bind_by_name($stmt, ':adressedlivraison', $adresse);
oci_bind_by_name($stmt, ':nomlivraison', $nomClient);
oci_bind_by_name($stmt, ':codepostallivraison', $codePostal);
oci_bind_by_name($stmt, ':villelivraison', $ville);
oci_bind_by_name($stmt, ':telephonelivraison', $telephone);
$r = oci_execute($stmt);
oci_free_statement($stmt);
oci_close($connect);
```

La première requête récupère l'ID du dernier paiement inséré en utilisant une séquence générée par la base de données.

La deuxième requête prépare une insertion de données dans la table "PAIEMENT" avec les valeurs des variables \$nomCarte, \$prenomCarte, \$numCarte, \$dateExpiration, \$cryptogramme.

La troisième requête prépare une insertion de données dans la table "DETAILPAIEMENT" avec les valeurs des variables, \$idcli (IDCLIENT), \$max_id, \$numCarte, \$dateExpiration, \$cryptogramme.

La quatrième requête prépare une insertion de données dans la table "COMMANDE" avec les valeurs des variables \$max_id (IDPAIEMENT), \$idcli (IDCLIENT), \$datecommande, \$fraislivraison, \$adresse, \$nomClient, \$codePostal, \$ville, \$telephone.