

# Développement Web

## 1. PHP

### 1.1 Ecriture de code PHP conceptuel

Donner le code PHP conceptuel qui permet de produire le fragment de code **XHTML** ci-dessous. On utilisera les fonctions de **std.php** pour générer ce fragment de code valide au sens de la grammaire **XHTML 1.0 stricte**. On essaiera de respecter le formatage des lignes, mais ce n'est pas grave s'il manque des espaces ou s'il y en a en trop, de même pour les sauts de ligne.

```
<blockquote>
<div class='connexion'>

  <div class='form-group' id='group1'>
    <label for='usernameSI' id='label01'>Utilisateur</label>
    <input type='text' id='usernameSI' name='usernameSI' value='' size='10' />
  </div>

  <div class='form-group' id='group2'>
    <label for='passwordSI' id='label02'>Mot de passe</label>
    <input type='password' id='passwordSI' name='passwordSI' value='' size='30' />
  </div>

  <div id='group3'>
    <input type='submit' value='submitSI' name='envoi' />
    <input type='reset' value='Reset' name='vidage' />
  </div>

</div>
</blockquote>
```

## 1.2 Compréhension de code PHP conceptuel

Expliquer ce que réalise le code ci-dessous écrit en PHP conceptuel dont les lignes ont été numérotées :

```
0001 <?php
0002 include("std.php") ;
0003
0004 h1("Ouverture des magasins") ;
0005
0006 ul() ;
0007
0008     debutli() ;
0009         p("texte") ;
0010             echo b("Angers&nbsp;:")." 9 h - 18 h " ;
0011             echo ghRouge("sauf le dimanche")."." ;
0012         finp() ;
0013     finli() ;
0014
0015     debutli() ;
0016         p("texte") ;
0017             echo b("Nantes&nbsp;:")." 9 h - 18 h (y compris le dimanche)." ;
0018         finp() ;
0019     finli() ;
0020
0021 finul() ;
0022 ?>
```

## 2. JAVASCRIPT

On dispose du fragment de code **HTML 5** suivant :

```
<h1>Suivi de livraisons</h1>

<form action='suivi.php' method='get' onsubmit='afficheSuivi() ; return false'>

    <p>
        Nombre de colis reçus&nbsp;:
    <br />
        <input type='text' id='ncr' name='ncr' value=' ' size='10' />
    </p>
```

```
<p>
  Nombre de colis commandés&nbsp;:
<br />
  <input type='text' id='ncc' name='ncc' value=' ' size='10' />
</p>

<p>
  <input type='submit' value='envoi' />
</p>

</form>

<div class='invisible cadre' id='resultats'></div>
```

Le rendu de la page ressemble fortement à l'image ci-dessous :



**Suivi de livraisons**

Nombre de colis reçus :

Nombre de colis commandés :

Il s'agit donc d'un formulaire avec deux champs d'entrée texte censés contenir des nombres entiers strictement positifs et un bouton pour exécuter le formulaire.

## 2.1 Javascript standard

Donner le code Javascript complet de la fonction citée dans le code **HTML** du formulaire qui réalise les actions suivantes :

- l'élément division qui correspond aux résultats est rendu visible via les styles visible et cadre;
- si une entrée est incorrecte (vide ou non entière ou non entière strictement positive) ou si le nombre de colis reçus est supérieur au nombre de colis commandés, on met un texte d'erreur associé dans la division et on s'arrête;
- s'il n'y a pas d'erreur de saisie, on affiche le taux arrondi de colis reçus en **respectant scrupuleusement** le texte indiqué dans l'image ci-dessous :

### Suivi de livraisons

Nombre de colis reçus :

Nombre de colis commandés :

Avec 3 colis reçus pour 7 colis commandés, le taux de colis reçu est 43 %.

On utilisera du Javascript standard et classique, sans faire appel à aucune librairie.

Dans la mesure où il s'agit d'un examen, vous n'êtes pas obligé de factoriser les tests sur les deux entrées avec une fonction commune de vérification. En d'autres termes, vous pouvez tout écrire dans une seule fonction.

## 2.2 Javascript avec JQuery

Indiquer ce qui change dans votre code Javascript pour cette fonction si on utilise la bibliothèque JQuery.

### 3. MySQL

On veut automatiser un système de suivi de livraisons de colis pour des commandes effectuées par des clients. On suppose qu'on dispose d'une base de données nommée SUIVI qui contient principalement les 3 tables suivantes : clients, commandes, livraisons.

Ce qui suit est une version simplifiée des tables qui composent la base.

La table clients contient, entre autres, les champs :

- idClient : l'identifiant du client, entier auto-incrémenté, clé primaire,
- nomClient : le nom du client, chaîne de caractères non vide,
- adresseClient : une chaîne de caractères non vide, déjà bien formatée, pour l'envoi du ou des colis.

La table commandes contient, entre autres, les champs :

- idCommande : l'identifiant de la commande, entier auto-incrémenté, clé primaire,
- dateCommande : la date à laquelle la commande a été passée, chaîne de caractères non vide au format YYYY-MM-JJ,
- refClient : entier, clé étrangère liée à idClient.

La table livraisons contient, entre autres, les champs :

- idLivraison : l'identifiant de la livraison, entier auto-incrémenté, clé primaire,
- statut : une chaîne de caractères qui indique si la livraison a été effectuée (contenu "oui") ou si elle ne l'est pas (contenu "non" – par défaut),
- dateLivraison : la date de livraison prévue, chaîne de caractères non vide au format YYYY-MM-JJ,
- refCommande : entier, clé étrangère liée à idCommande.
- nbColis : entier strictement positif, nombre de colis à livrer.

Donner les instructions SQL au sens de MySQL qui permettent de répondre aux questions suivantes :

- combien y a-t-il de commandes en tout dans la table des commandes ?
- combien y a-t-il de livraisons qui n'ont pas encore été effectuées ?
- à titre de vérification, y a-t-il des livraisons dont la date de livraison est antérieure à la date de la commande associée, ce qui est bien sûr impossible en théorie si le logiciel de saisie des commandes et de suivi des livraisons est bien programmé ?
- quel est le plus grand nombre de colis à livrer ou livrés en une seule livraison, que la livraison ait déjà eu lieu ou pas ?
- quel est le nom du ou des clients pour lequel ou lesquels une ou plusieurs livraisons correspondent à ce plus grand nombre de colis à livrer ? On triera l'affichage par ordre alphabétique de nom de client. Si plusieurs telles livraisons correspondent à un même client, on n'affichera ce client qu'une seule fois.

Exemple d'affichage :

Le plus grand nombre de colis à livrer en une seule fois est 5.  
Voici les clients associés à de telles livraisons :

```
+-----+-----+
| NomClient | nbColis |
+-----+-----+
| DUPONT   |      5 |
| DURAND   |      5 |
+-----+-----+
```

## 4. Développement d'une application

On voudrait mettre en place une page Web qui permet, entre autres, de choisir entre afficher les livraisons du jour (effectuées ou non, avec l'indication du statut) et afficher les livraisons non encore effectuées (triées par date croissante), sachant que la page affiche le nombre de livraisons impliquées dans chacun des cas.

## Question 4.1

Expliquer comment vous construiriez cette page en PHP conceptuel avec des appels à MySQL. Vous détaillerez au passage l'interface HTML qui vous paraît adaptée (tableau, formulaire, liens, boutons avec ou sans code Javascript...). On ne demande ici aucun code PHP ni aucun code Javascript.

Voici un exemple du rendu d'une telle page, sans présumer que les zones encadrées sont des boutons, des zones avec des *onclick* ou des champs de type *submit* :

**Choisissez l'action à exécuter :**

afficher les **23** livraisons du jour

afficher les **6** livraisons non encore effectuées

Vous devez au passage justifier votre choix de l'utilisation ou non d'AJAX pour que l'affichage demandé se fasse dans la même page ou dans une autre page.

## Question 4.2

Quelles fonctions PHP pourriez-vous être amené à développer pour produire cette page et avec quels paramètres ? Là encore, on ne demande ici aucun code PHP. Si vous devez appeler ce ou ces fonctions PHP via du Javascript, vous l'indiquerez, sans écrire le code Javascript associé.

## Question 4.3

Donner le code en PHP conceptuel pour afficher cette page. On supposera que le code pour le début de la page a déjà été exécuté. SI vous utilisez des fonctions ou des sous-programmes, **vous ne donnerez pas** leur code, vous vous contenterez d'expliquer ces fonctions et leur entête ou d'indiquer quelle fonction vue en cours vous utilisez.

## Question 4.4

On s'intéresse maintenant à la page qui afficherait dans un tableau les livraisons du jour non encore effectuées triées par ordre décroissant de nombre de colis à livrer. Voici un mini-exemple avec trois livraisons pour la date du 05/02/2021 :

<b>Suivi des 3 livraisons restantes en date du 05/02/2021</b>		
<b>Id+Lien</b>	<b>NomClient</b>	<b>NbColis</b>
<a href="#">21502</a>	DURANT	3
<a href="#">443217</a>	DUPOND	1
<a href="#">12856</a>	DUVALE	1

Expliquer quelle méthode vous utiliseriez pour construire cette page. Si cette page devrait être générée via une fonction PHP, quels tests seriez-vous amenés à effectuer pour garantir la validité et la robustesse de cette fonction ?

## Question 4.5

Donner le code en PHP conceptuel de cette fonction.

## 5. Discussion

Essayez de répondre à la question suivante :

*Est-ce que les technologies du développement Web vont beaucoup changer dans les cinq prochaines années ?*

Votre réponse devra mettre en évidence votre culture naissante ainsi que votre recul et votre esprit de synthèse en matière de développement Web **à l'aide d'exemples concrets**. Vous n'êtes pas obligé de vous limiter aux logiciels libres.

Votre réponse devra faire 10 lignes au minimum, sans limite de maximum. On utilisera au moins 3 mots de 4 syllabes ou plus pour transmettre « un contenu rédactionnel fort ».



# ESQUISSE DE SOLUTION

## 1. PHP

### 1.1 Ecriture de code PHP conceptuel

```
<?php
include("std.php") ;

blockquote() ;
div("connexion") ;
sdl() ;

div("form-group","group1") ;
    label("Utilisateur","usernameSI","label01") ;
    input_text("usernameSI","", "10") ;
findiv() ; sdl() ;

div("form-group","group2") ;
    label("Mot de passe","passwordSI","label02") ;
    input_password("passwordSI","", "30") ;
findiv() ; sdl() ;

div("", "group3") ;
    input_submit("submitSI","envoi") ;
    input_reset("Reset","vidage") ;
findiv() ; sdl() ;

findiv() ;
finblockquote() ;

?>
```

## 1.2 Compréhension de code PHP conceptuel

Le code fourni commence par charger à la ligne 2 les fonctions du fichier `std.php`. A la ligne 4 on produit un élément `<h1>` avec le texte *Ouverture des magasins*.

Il y a ensuite création d'une liste non ordonnée, élément `<ul>` ouverte à la ligne 6 et fermée à la ligne 21. Cette liste contient deux items de liste, éléments `<li>` qui contiennent chacun un paragraphe, élément `<p>`. Le premier `<li>` affiche *Angers* en gras avec l'heure d'ouverture en texte normal (non gras) puis l'indication "*sauf le dimanche*" écrite en rouge et gras. Le second `<li>` affiche *Nantes* en gras avec l'heure d'ouverture en texte normal (non gras).

Voici le code produit :

```
<h1>Ouverture des magasins</h1>

<ul>
<li >
<p class='texte'>
<strong>Angers&nbsp;:</strong> 9 h - 18 h <span class="grouge">sauf le dimanche</sp
</p>

</li>
<li >
<p class='texte'>
<strong>Nantes&nbsp;:</strong> 9 h - 18 h (y compris le dimanche).
</p>

</li>
</ul>
```

Le rendu en est :

## Ouverture des magasins

- **Angers** : 9 h - 18 h **sauf le dimanche**.
- **Nantes** : 9 h - 18 h (y compris le dimanche).

## 2. JAVASCRIPT

### 2.1 Javascript standard

```
function afficheSuivi() {

    // on rend visible la zone des résultats

    window.document.getElementById("resultats").setAttribute("class","visible cadre")

    // récupération des entrées

    nbCr    = window.document.getElementById("ncr").value
    nbCc    = window.document.getElementById("ncc").value

    // gestion des entrées : si entrées incorrectes, on indique l'erreur
    // et on quitte la fonction

    if (nbCr=="") {
        texte = " nombre de colis reçus non renseigné "
        window.document.getElementById("resultats").innerHTML = texte
        return
    } // fin si

    if (nbCc=="") {
        texte = " nombre de colis commandés non renseigné "
        window.document.getElementById("resultats").innerHTML = texte
        return
    } // fin si

    nbCr = parseInt(nbCr)
    if (isNaN(nbCr)) {
        texte = " nombre de colis reçus non entier "
        window.document.getElementById("resultats").innerHTML = texte
        return
    } // fin si

    if (nbCr<0) {
        texte = " nombre de colis reçus non entier strictement positif"
        window.document.getElementById("resultats").innerHTML = texte
        return
    }
}
```

```

} // fin si

if (nbCc=="") {
    texte = " nombre de colis commandés non renseigné "
    window.document.getElementById("resultats").innerHTML = texte
    return
} // fin si

nbCc = parseInt(nbCc)
if (isNaN(nbCc)) {
    texte = " nombre de colis commandés non entier "
    window.document.getElementById("resultats").innerHTML = texte
    return
} // fin si

if (nbCc<0) {
    texte = " nombre de colis commandés non entier strictement positif"
    window.document.getElementById("resultats").innerHTML = texte
    return
} // fin si

// si aucune erreur, on calcule le taux de colis reçus et on affiche
// la phrase demandée

texte = ""
txDr = 100 * parseInt(nbCr)/parseInt(nbCc)
txDrInt = Math.round(txDr)
texte = " Avec " + nbCr + " colis reçus pour " + nbCc + " colis commandés, "
texte += " le taux de colis reçu est " + txDrInt + " %."
window.document.getElementById("resultats").innerHTML = texte
return

} // fin de fonction afficheSuivi

```

## 2.1 Javascript avec JQuery

Les seuls changements à effectuer consistent à remplacer les appels classiques de Javascript à `window.document.getElementById(id)` par des appels à `$('#'+id)` avec les fonctions correspondantes, c'est-à-dire `html()` pour `.innerHTML`, `val()` pour `.value`, `attribute()` pour `.setAttribute()`.

### 3. SQL

```
# nombre de commandes en tout

SELECT COUNT(*) FROM commandes ;

# nombre de livraisons non encore effectuées

SELECT COUNT(*) FROM livraisons WHERE statut="non" ;

# vérifications : livraisons avant commande

SELECT idLivraison, dateLivraison, idCommande, dateCommande
      FROM livraisons, commandes
      WHERE refCommande=idCommande AND dateLivraison < dateCommande ;

# plus grand nombre de colis en une seule livraison

SELECT @maxNbColis := MAX(nbColis) FROM livraisons ;

# nom des clients pour ce nombre maximum de colis

SELECT DISTINCT(nomClient) AS NomClient, nbColis
      FROM clients, livraisons, commandes
      WHERE idClient=refClient
            AND refCommande=idCommande
            AND nbColis= @maxNbColis
      ORDER BY NomClient ;
```

### 4. PHP

#### Question 4.1

Il est sans doute raisonnable de produire une page XHTML ou HTML 5 avec avec deux zones cliquables, que ce soit des boutons ou des éléments `<span>` avec des attributs `onclick` pour que Javascript réalise via AJAX les affichages dans la page car ce sera le plus pratique pour la visualisation.

Il faut certainement utiliser une fonction PHP nommée `combien()` comme vue en cours pour déterminer le nombre de livraisons du jour à effectuer et le nombre de livraisons non effectuées.

Il n'y a donc pas besoin de formulaire ni de tableau.

## Question 4.2

A part la fonction `combien(base,table,condition)` il n'y pas besoin d'autres fonctions PHP autres que celles de `std.php` pour produire la page.

Le code Javascript fera sans doute appel à d'autres pages Web écrites en PHP pour afficher les livraisons demandées.

On peut imaginer que les fonctions Javascript associées aux actions se nomment `livraisonsDuJour()` et `livraisonsRestantes()`.

## Question 4.3

Voici le code en PHP conceptuel, non factorisé :

```
<?php

h1("Choisissez l'action à réaliser ");

blockquote() ;

    $today = date("d/m/Y") ;
    $nbLdj = combien("SUIVI","livraisons","dateLivraison='$today'") ;
    div("cadre") ;
        $txt1 = "afficher les ".b("$nbLdj")." livraisons du jour" ;
        $act1 = "onclick='livraisonsDuJour()'" ;
        echo s_span($txt1,"","ldj",$act1) ;
    findiv() ;

    $nbLne = combien("SUIVI","livraisons","statut='non'") ;
    div("cadre") ;
        $txt2 = "afficher les ".b("$nbLdj")." livraisons non encore efectuées" ;
        $act2 = "onclick='livraisonsRestantes()'" ;
        echo s_span($txt2,"","ldj",$act2) ;
    findiv() ;

finblockquote() ;
?>
```

## Question 4.4

On pourrait certainement penser qu'il faudrait utiliser la fonction `combien()` pour déterminer le nombre de livraisons restantes à effectuer pour pouvoir l'afficher dans le titre en élément `<h1>`.

Ensuite, les fonctions `table()`, `entetesTableau()`, `tr`, `td` et `href` de `std.php` sont suffisantes pour réaliser l'affichage demandé.

Pour trier le tableau, on peut utiliser la classe `sorttable` vue lors du cours de Javascript.

La fonction PHP pour générer le tableau ne devrait certainement ne devrait comporter qu'un seul paramètre, la date, ce qui permet de voir les livraisons futures à effectuer. Toutefois, dans la mesure où PHP dispose de la fonction `checkdate(month,day,year)` on utilisera ces trois paramètres dans l'ordre indiqué.

Un test pour vérifier que le paramètre est une date valide est le minimum à prévoir.

Pour l'affichage, deux cas sont à envisager. S'il n'y a aucune livraison restante, pas besoin de tableau. Sinon, il faut effectivement prévoir l'affichage en tableau.

Au lieu de passer par la fonction `combien`, on peut raisonnablement supposer qu'une fonction `infosLivraisonsRestantes()` interroge la base de données et renvoie un tableau avec les informations `id`, `nom`, `nbcolis` auquel cas le nombre de livraisons restantes est la taille du tableau.

Une fonction `lienClient(idc)` doit alors pouvoir produire le lien demandé.

Voici le tableau correspondant à l'exemple :

```
[
  [21502,"DURANT",3] ,
  [44327,"DUPOND",1] ,
  [12856,"DUVALE",1]
]
```

## Question 4.5

```
<?php
```

```
#####
```

```
function livraisonsRestantes($mois,$jour,$an) {
```

```
#####
```

```

$laDate = $jour."/".$mois."/".$an ;

# vérification de la date
# exemple 1 : livraisonsRestantes(2,05,2021) OK ;
# exemple 2 : livraisonsRestantes(2,35,2021) pas OK ;

if (!checkdate($mois,$jour,$an)) {
    h2("date $laDate invalide") ;
    return ;
} ; # fin si

# récupération des informations

$tabLr = infosLivraisonsRestantes($mois,$jour,$an) ;
$nbLne = count($tabLr) ;
# sans doute équivalent à
# $nbLne = combien("SUIVI","livraisons","statut='non' AND dateLivraison='$laDate'") ;

# y a-t-il quelque chose à afficher ?

if ($nbLne==0) {
    h2("aucune livraison restante en date du $laDate") ;
    return ;
} ; # fin si

# si oui, affichage en tableau

h2("Suivi des $nbLne livraisons restantes") ;
h2(s_ncmp(8)."en date du $laDate") ;
pvide() ;

blockquote() ;

table(1,10,"collapse sorttable") ;

    entetesTableau("Id+Lien NomClient NbColis") ;

    for ($idl=0;$idl<$nbLne;$idl++) {

        $idClient = $tabLr[$idl][0] ;
        $nomClient = $tabLr[$idl][1] ;
        $nbColis = $tabLr[$idl][2] ;
    }

```



```
        tr() ;
            td() ; echo lienClient($idClient) ; fintd() ;
            td() ; echo $nomClient ; fintd() ;
            td("R") ; echo $nbColis ; fintd() ;
        fintr() ;

    } ; # fin pour idl

fintable() ;

finblockquote() ;

} ; # fin de fonction livraisonsRestantes
```