

Développement Web

1. PHP

1.1 Ecriture de code PHP conceptuel

Donner le code PHP conceptuel qui permet de produire le code **XHTML** ci-dessous.

```
<ol>
  <li >
    <p class='texte' id='recoMed'>
      Recommandations médecins
    </p>
  </li>
  <li >
    <p class='texte' id='recoPharm'>
      Recommandations pharmaciens
    </p>
  </li>
</ol>
```

On utilisera les fonctions de **std.php** pour générer un fragment de code complet pour **XHTML 1.0** valide au sens de la grammaire *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.

1.2 Compréhension de code PHP conceptuel

Expliquer ce que produit le code ci-dessous écrit en PHP conceptuel :

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

$img = "aa.png" ;
p() ;
  nbsp(7) ;
  sdl() ;
  echo href($img,img($img,$img,600)) ;
fimp() ;

?>
```

2. JAVASCRIPT

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

```
<!doctype html><html lang="fr">
  <head><meta charset="UTF-8"><title>affichage masquage</title>
    <link rel="stylesheet" type="text/css" href="std.css" title="gh"/>
  </head>
  <body>

    <p id="clicProd" onclick="dscProd()" class="boutonProd">
      montrer la description
    </p>

    <div id="description" class="invisible">
      <blockquote>
        <h3>Description du produit</h3>
        <p class="descriptionProduit">
          Ce produit est...
        </p>
      </blockquote>
    </div>

  </body>
</html>
```

2.1 Javascript standard

Donner le code Javascript complet de la fonction citée dans ce code **HTML 5** qui réalise les actions suivantes :

- si le paragraphe dont l'*id* est clicProd contient exactement le texte `montrer la description`, alors le clic sur le paragraphe rend visible la division dont l'*id* est `description` et le contenu du paragraphe devient `masquer la description`;
- si le paragraphe dont l'*id* est clicProd contient exactement le texte `masquer la description`, alors le clic sur le paragraphe rend invisible la division dont l'*id* est `description` et le contenu du paragraphe devient `montrer la description`.

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

On pourra ignorer les espaces multiples et les retours à la ligne dans le paragraphe clicProd.

2.2 Javascript avec JQuery ou Prototype

Donner le code Javascript correspondant à la fonction demandée en utilisant cette fois-ci, au choix, la bibliothèque JQuery ou la bibliothèque Prototype ou les deux à fois.

3. XML/DTD

On dispose de la grammaire DTD suivante ;

```
<!ELEMENT a (b,c*,d?)+ >
<!ELEMENT b (e*,g?,f,g?) >
<!ELEMENT c EMPTY >
<!ELEMENT d EMPTY >
<!ELEMENT e EMPTY >
<!ELEMENT f EMPTY >
<!ELEMENT g EMPTY >
```

Pour les 4 documents XML listés ci-dessous, indiquer lesquels sont valides et lesquels ne sont pas valides pour cette grammaire DTD. Vous expliquerez succinctement à chaque fois pourquoi.

```
document 1 : <a><b><f /></b></a>
document 2 : <a><c><d /></c></a>
document 3 : <a><b><f /><g /><g /><g /></b></a>
document 4 : <a><b><f /></b><b><f /></b></a>
```

4. SQL

On veut mettre en place un système de conseil de produits, disons des médicaments, à l'aide de recommandations exprimées par des personnes dites expertes, disons ici des médecins et des pharmaciens. On suppose qu'on dispose d'une base de données nommée **RECOPR** qui contient principalement les 4 tables suivantes **produits**, **medecins**, **pharmaciens**, **recommandations**.

La table **produits** contient, entre autres, les champs

- **idProduit** : l'identifiant du produit, entier auto-incrémenté,
- **nomProduit** : le nom du produit, chaîne de caractères,
- **refProduit** : la référence du produit, chaîne de caractères,
- **dscProduit** : la description du produit, chaîne de caractères.

La table **medecins** contient, entre autres, les champs

- **idMedecin** : l'identifiant du médecin, chaîne de caractères, clé primaire,
- **nomMedecin** : le nom du médecin, chaîne de caractères.

La table **pharmaciens** contient, entre autres, les champs

- **idPharmacien** : l'identifiant du pharmacien, chaîne de caractères, clé primaire,
- **nomPharmacien** : le nom du pharmacien, chaîne de caractères.

La table **recommandations** contient, entre autres, les champs

- **idRecommandation** : l'identifiant de la recommandation, entier auto-incrémenté,
- **texteRecommandation** : le texte de la recommandation, chaîne de caractères,
- **refProduit** : entier censé être associé au champ **idProduit** (clé étrangère),
- **refExpert** : chaîne de caractères, censée être associée soit à un champ **idMedecin**, soit à un champ **idPharmacien**, (clé étrangère).

Donner les instructions SQL au sens de MySQL ou SQLite3 qui permettent de répondre aux questions suivantes :

- combien y a-t-il en tout de produits ?
- combien y a-t-il en tout de produits distincts recommandés ?
- quel est le nom du produit le plus recommandé par les médecins seulement ?
- quel est le nom du produit le plus recommandé en même temps par les médecins et par les pharmaciens ?

5. PHP

On voudrait mettre en place une page Web qui affiche le *top 5* des produits, c'est-à-dire la liste des 5 meilleurs produits, selon le critère de choix lié au plus grand nombre de recommandations :

- choix 1** : de médecins seulement ;
- choix 2** : de pharmaciens seulement ;
- choix 3** : de médecins ou de pharmaciens ;
- choix 4** : de médecins et pharmaciens pour le même produit.

Donner le code PHP conceptuel d'une fonction nommée `topCinq()` qui admet deux paramètres dans cet ordre `$choix` et `$desc`.

Le paramètre `$choix` doit être un entier entre 1 et 4 et correspond au critère de choix. Pour ne pas pénaliser ceux qui n'arriveraient pas à écrire le code SQL correspondant, on pourra admettre qu'il existe un tableau `$cmdSQL` tel que pour le choix `$i` on doit utiliser le code SQL mis dans `$cmdSQL[$i]`.

Le paramètre `$desc` est un paramètre facultatif qui permet de savoir si on doit mettre trois ou quatre colonnes dans le tableau résultat : si `$desc` vaut 0, pas de description, on affiche seulement les numéros 1, 2, 3, 4, 5 puis le nom du produit puis le nombre de recommandations retournées par le code SQL. Par contre si `$desc` vaut 1 on affiche en plus une colonne "description du produit". Par défaut, ce paramètre vaut 0.

Voici un exemple fictif d'affichage à reproduire en PHP conceptuel suite à l'appel de `topCinq(3,1)`.

```
<table border="1" cellpadding="15" class='collapse sortable' summary="choix 3" >
  <tr><th align='center'>Numéro</th>
    <th align='center'>Nom du produit</th>
    <th align='center'>Recommandations</th>
    <th align='center'>Description</th>
  </tr>
  <tr><td align='right'>1</td>
    <td align='left'>Diloprane</td>
    <td align='right'>19&nbsp;321</td>
    <td align='left'>Antalgique non opiacé</td>
  </tr>
  <tr><td align='right'>2</td>
    <td align='left'>Omidium</td>
    <td align='right'>3&nbsp;850</td>
    <td align='left'>Traitement symptomatique des diarrhées</td>
  </tr>
  [...]
</table>
```

On utilisera obligatoirement la fonction `entetesTableau()` et on respectera les cadrages et formatages du tableau (comme 19 321 au lieu de 19321).

Si vous n'avez pas le temps de tout écrire, expliquez vos choix et inventez des fonctions...

6. Discussion

Essayez de répondre à la question suivante :

Va-t-on voir apparaître dans les prochaines années beaucoup de nouvelles fonctionnalités HTML et Javascript ?

Votre réponse devra mettre en évidence votre culture naissante ainsi que votre recul et votre esprit de synthèse en matière de développement Web. 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 ».

Il y a un élément paragraphe produit, correctement ouvert et fermé. Il contient 7 espaces insécables, puis un lien qui est donc décalé qui contient une image. Si on clique sur le lien, l'image est affichée en vraie grandeur alors que dans le lien, elle est affichée avec une largeur de 600 pixels seulement. Le texte alternatif dans l'image contient le nom du fichier source de l'image.

2. JAVASCRIPT

2.1 Javascript standard

Pour tester si paragraphe contient un certain texte, ce qui est sans doute suffisant ici, on peut utiliser la méthode `includes` de la classe `String`, soit la solution :

```
function dscProd() {  
  
    parClic = window.document.getElementById("clicProd")  
    divDsc = window.document.getElementById("description")  
  
    if (parClic.innerHTML.includes("montrer la description")) {  
        divDsc.setAttribute("class","visible")  
        parClic.innerHTML="masquer la description"  
    } else {  
        if (parClic.innerHTML.includes("masquer la description")) {  
            divDsc.setAttribute("class","invisible")  
            parClic.innerHTML="montrer la description"  
        } // finsi  
    } // finsi  
  
} // fin de fonction dscProd
```

Si on veut enlever les espaces multiples de début et tester exactement le texte du paragraphe, il faut passer par la méthode `trim` de la classe `String` et écrire `if (parClic.innerHTML.trim()=="montrer la description")` au lieu de `if (parClic.innerHTML.includes("montrer la description"))`.

2.1 Javascript avec JQuery ou Prototype

Solution JQuery

```
function dscProd() {

    parClic = $("#clicProd")
    divDsc = $("#description")
    if (parClic.html().trim()=="montrer la description") {
        divDsc.attr("class","visible")
        parClic.html("masquer la description")
    } else {
        if (parClic.html().trim()=="masquer la description") {
            divDsc.attr("class","invisible")
            parClic.html("montrer la description")
        } // finsi
    } // finsi

} // fin de fonction dscProd
```

Solution Prototype

```
function dscProd() {

    parClic = $("clicProd")
    divDsc = $("description")
    if (parClic.innerHTML.trim()=="montrer la description") {
        divDsc.setAttribute("class","visible")
        parClic.update("masquer la description")
    } else {
        if (parClic.innerHTML.trim()=="masquer la description") {
            divDsc.setAttribute("class","invisible")
            parClic.update("montrer la description")
        } // finsi
    } // finsi

} // fin de fonction dscProd
```

3. XML/DTD

Voici la réponse de xmstarlet :

```
$gh> xmlstarlet val -e -d licpro2019-03.dtd licpro2019-04-*.xml

licpro2019-04-1.xml - valid

licpro2019-04-2.xml:1.0: Element a content does not follow the DTD,
                        expecting (b , c* , d?)+, got (c)
licpro2019-04-2.xml:1.0: Element c was declared EMPTY this one has content
licpro2019-04-2.xml - invalid

licpro2019-04-3.xml:1.0: Element b content does not follow the DTD,
                        expecting (e* , g? , f , g?), got (f g g g)
licpro2019-04-3.xml - invalid

licpro2019-04-4.xml - valid
```

4. SQL

```
SELECT COUNT(idProduit) FROM produits ;
```

```
SELECT COUNT(DISTINCT(refProduit)) FROM recommandations ;
```

```
SELECT a.nomProduit, b.refProduit, COUNT(b.refProduit) AS cnt
FROM produits AS a,recommandations AS b , medecins
WHERE b.refExpert=idMedecin
      AND a.idProduit=b.refProduit
GROUP BY b.refProduit 0
ORDER BY cnt DESC
LIMIT 1;
```

```
SELECT a.nomProduit, b.refProduit, COUNT(b.refProduit) AS cnt
FROM produits AS a,recommandations AS b ,
      medecins AS c, pharmaciens as d
WHERE (b.refExpert=c.idMedecin OR b.refExpert=d.idPharmacien)
      AND a.idProduit=b.refProduit
GROUP BY b.refProduit
ORDER BY cnt DESC
LIMIT 1 ;
```

5. PHP

```
<?php

function topCinq($choix,$desc=0) {

# remarques : - le tableau cmdSQL est déjà rempli
#             - la fonction executeGHsql renvoie un tableau

$codeSql = $cmdSQL[ $choix ] ;
$tabRes  = executeGHsql( $codeSql ) ;

table(1,10,"collapse") ;

    $entetes = "Numéro Nom__du__produit Recommandation" ;
    if ($desc==1) {
        $entetes .= " Description" ;
    } # fin si
    entetesTableau( $entetes ) ;
    $numero = 0 ;
    foreach $tabRes as $resultat {
        $numero++;
        tr() ;
            td("R") ; echo $numero ; fintd() ;
            td("L") ; echo $resultat["produit"] ; fintd() ;
            td("R") ; echo fmtNombre($resultat["nbReco"]) ; fintd() ;
            if ($desc==1) {
                td() ; echo $resultat["description"] ; fintd() ;
            } # fin si
        fintr() ;
    } ; # fin pour chaque
fintable() ;

} # fin de fonction topCinq

function fmtNombre($n) { # produit 19 321 au lieu de 19321
    ...
} # fin de fonction fmtNombre

?>
```