

1 Bases de données, scripts, formulaires et Google (8 points)

On considère un serveur web et un gestionnaire de bases de données MySQL similaires à ceux présentés lors des travaux pratiques. MySQL gère une base de données politique. Sur ce serveur, un utilisateur nommé utilisateur peut se connecter à cette base avec le mot de passe toto154. Soit le fichier script.php suivant, stocké sur le serveur :

```
<?php
echo "<html>";
echo "<head><title>Contenu de la base</title>";
echo "</head>";
echo "<body>";
echo "<h1>Contenu de la base</h1>";
echo "<table>";
$connexion=mysql_connect("localhost","utilisateur","toto154");
mysql_select_db("politique",$connexion);
$res=mysql_query("select Candidat.nom,Candidat.prenom,Parti.nomParti
                 from Candidat,Parti
                 where Candidat.age<30
                 and Candidat.numeroParti=Parti.numeroParti",$connexion);
echo "<table>";

echo "<tr>";
while($tuplet=mysql_fetch_object($res)){
    echo "<td>$tuplet->nom</td><td>$tuplet->prenom</td>";
}
echo "</tr>";

echo "</table>";
echo "</body>";
echo "</html>";
?>
```

- (1 point) Donnez un schéma de base de données pour politique compatible avec l'exécution sur le serveur de ce script Php (nom de tables, nom des attributs correspondants.)
- (1 point) On suppose qu'il y a n tuples (ou lignes) dans le résultat de cette requête SQL. En fonction de n , combien de colonnes comporte alors le tableau fabriqué par ce script? Combien de lignes?
- (1 point) Un utilisateur *extérieur* se connecte avec un navigateur via HTTP à l'URL `http://cortes.cnam.fr:8080/script.php`, correspondant au script précédent sur le serveur. Il obtient une page avec un tableau non vide. Il enregistre la page obtenue sur son disque, dans un fichier `truc.php`. Le mot de passe `toto154` est-il présent dans `truc.php`? Motivez votre réponse.
- (1 point) On suppose que le serveur web et la base de données sont hors service. L'utilisateur ouvre son fichier `truc.php` dans son navigateur. Que voit-il (motivez votre réponse)?
 - Uniquement les mots "Contenu de la base" et un tableau vide?
 - Un message d'erreur indiquant que le serveur est hors service?
 - Le même contenu que précédemment?
 - Le code source du script Php sans le mot de passe `toto154`?
- (2 points) Proposez un document HTML et un script Php permettant la saisie d'un âge, puis l'affichage dans un tableau du **nom et prénom** de tous les candidats ayant **moins de cet âge**.
- (2 points) L'engin de recherche Google permet des interrogations en mode GET. Par exemple, pour se connecter directement à la meilleure page web correspondant à la recherche "Yves Pergoulaut", il suffit de se connecter à la page `http://www.google.com/search?q=Yves+Pergoulaut&btn=direct`. Modifier votre script pour associer à chaque candidat un lien HTML vers la requête Google le concernant. On dispose d'une fonction Php `Replace(s)` qui renvoie une copie de la chaîne `s` où les espaces ont été remplacés par le symbole `+` (Les candidats n'ont pas le symbole `+` dans leur nom et prénom.) On rappelle que la concaténation de chaînes s'obtient en Php avec l'opérateur `."`.

2 XML (7 points)

Voici un document XML qui contient des informations sur des cours en informatique :

```
<?xml version='1.0' encoding='iso-8859-1' ?>
<ENSEIGNEMENT>
  <COURS INTITULE='Commerce Electronique'>
    <SALLE>Amphi A</SALLE>
    <MODULE INTITULE='PHP et MySQL'>
      <ENSEIGNANT>Gross-Amblard</ENSEIGNANT>
    </MODULE>
    <MODULE INTITULE='XML'>
      <ENSEIGNANT>Amann</ENSEIGNANT>
      <ENSEIGNANT>Villemin</ENSEIGNANT>
    </MODULE>
    <ETUDIANT NUMERO=' I235678' >
      <NOTE>16</NOTE>
    </ETUDIANT>
    <ETUDIANT NUMERO=' I784521' />
  </COURS>
  <COURS INTITULE='Bases de données'>
    <SALLE>Painlevé</SALLE>
    <MODULE INTITULE='SQL'>
      <ENSEIGNANT>Amann</ENSEIGNANT>
    </MODULE>
    <MODULE INTITULE='Optimisation'>
      <ENSEIGNANT>Scholl</ENSEIGNANT>
    </MODULE>
    <ETUDIANT NUMERO=' A675432' >
      <NOTE>14</NOTE>
    </ETUDIANT>
    <ETUDIANT NUMERO=' I235678' />
    <ETUDIANT NUMERO=' B568709' >
      <NOTE>12</NOTE>
    </ETUDIANT>
  </COURS>
</ENSEIGNEMENT>
```

Questions XPath : Traduisez les questions suivantes en expressions XPath :

1. Les noms des enseignants ?

Solution :

```
//ENSEIGNANT
```

2. Les noms des enseignants du module XML ?

Solution :

```
//MODULE[@INTITULE='XML']/ENSEIGNANT
```

3. Les intitulés de cours avec plus que 2 inscrits ?

Solution :

```
//COURS[count(ETUDIANT) > 2]/@INTITULE
```

4. Les numéros d'étudiant qui n'ont pas de note pour le cours 'Commerce électronique' ?

Solution :

```
//COURS[@INTITULE='Commerce Electronique']/ETUDIANT[not(NOTE)]/@NUMERO
```

Voici un programme XSLT avec trois règles de transformation :

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="text" />

  <xsl:template match="/">
    <xsl:apply-templates select="ENSEIGNEMENT/COURS" />
  </xsl:template>

  <xsl:template match="COURS">
    -----
    <xsl:value-of select="@INTITULE" />
    Salle   : <xsl:value-of select="SALLE" />
    Modules : <xsl:apply-templates select="MODULE" />
    Inscrits (note):
      <xsl:for-each select="ETUDIANT">
        - <xsl:value-of select="@NUMERO" />
          <xsl:if test="NOTE"> (<xsl:value-of select="NOTE" />)</xsl:if>
      </xsl:for-each>
  </xsl:template>

  <xsl:template match="MODULE">
    <xsl:value-of select="@INTITULE" />
    <xsl:if test="position() &lt; last()">, </xsl:if>
  </xsl:template>

</xsl:stylesheet>

```

Questions XSLT :

1. Donnez le résultat de la transformation du document précédent.

Solution :

```

-----
  Commerce Electronique
  Salle   : Amphi A
  Modules : PHP et MySQL, XML
  Inscrits (note):

      - I235678 (16)
      - I784521
-----

  Bases de données
  Salle   : Painlevé
  Modules : SQL, Optimisation
  Inscrits (note):

      - A675432 (14)
      - I235678
      - B568709 (12)

```

2. Modifiez le programme afin qu'il affiche :

- (a) que les numéros d'étudiants avec une note ;
- (b) les enseignants pour chaque module (entre parenthèses après le nom du module) ;

Solution :

```

<?xml version="1.0" encoding="iso-8859-1"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="text" />

```

```

<xsl:template match="/">
  <xsl:apply-templates select="ENSEIGNEMENT/COURS"/>
</xsl:template>

<xsl:template match="COURS">
  -----
  <xsl:value-of select="@INTITULE"/>
  Salle   : <xsl:value-of select="SALLE"/>
  Modules : <xsl:apply-templates select="MODULE"/>
  Inscrits (note):
    <xsl:for-each select="ETUDIANT[NOTE]">
      - <xsl:value-of select="@NUMERO"/>
      <xsl:if test="NOTE"> (<xsl:value-of select="NOTE"/>)</xsl:if>
    </xsl:for-each>
</xsl:template>

<xsl:template match="MODULE">
  <xsl:value-of select="@INTITULE"/>
  <xsl:text> (</xsl:text>
  <xsl:for-each select="ENSEIGNANT[position() &lt; last()]">
    <xsl:value-of select="."/>, </xsl:for-each>
  <xsl:value-of select="ENSEIGNANT[last()]">
  <xsl:text>)</xsl:text>
  <xsl:if test="position() &lt; last()">, </xsl:if>
</xsl:template>

</xsl:stylesheet>

```