Licence d'Informatique, 2ème année
Internet & Outils (IO2)

26 mai 2010

Cet énoncé porte le numéro 12 , **merci de le reporter sur votre copie**.
Les documents ne sont pas autorisés à l'exception d'une feuille A4 recto verso.
L'utilisation d'un téléphone pour quelque raison que ce soit se traduira par une exclusion de l'épreuve.

I Questions générales

1/ Comment est placé par défaut un élément de type bloc ?

2/ Que désigne l'unité de mesure `ex` en css ?

3/ Par quel mécanisme le protocole Internet permet-il la transmission simultanée, sur un même canal, de plusieurs flux de données ?

4/ Pourquoi le xhtml 2 ne verra-t-il probablement jamais le jour ?

5/ Décrire brièvement le principe de communication client/serveur

6/ Java et javascript sont tous les deux des langages orientés objet. Mais Java utilise des classes alors que javascript utilise des prototypes. Expliquer brièvement la différence.

II XHTML + CSS

Considérez les codes html et css fournis en annexe à la fin de l'énoncé.

1/ Le code html décrit un élément `body`. Ecrire un document xhtml 1.0 strict valide incluant cet élément body fourni (bien entendu, vous ne devez pas recopier tout le contenu de body dans votre copie). Ce document aura pour feuille de style le fichier `style.css` dont le contenu est le code css fourni et l'encodage de caractère utilisé sera utf8.

2/ Dessiner ce qui s'affichera sur un navigateur au chargement de la page (si nécessaire, vous préciserez les couleurs au moyen d'une flèche avec leur nom. Merci d'éviter l'utilisation d'un stylo rouge) et précisez ce qui peut changer si la page est redimensionnée de sorte que tout le contenu ne peut plus être affiché.

3/ Ajouter du code html produisant le résultat présenté dans la capture d'écran fournie en annexe. Précisez où vous insérerez le code dans le document original.

III PHP + formulaires XHTML

1/ Ecrire une instruction php qui affiche un paragraphe html avec la date (dans un format quelconque). Que faut-il faire pour que ce paragraphe apparaisse dans un document html complet ?

2/ Ecrire le code html du formulaire ci-dessous. Les données seront transmises au moyen de la methode GET.

paramètre ? OK

Que faut-il faire pour que le champ de saisie soit activé par un clic sur le mot *paramètre* ?

3/ Dans un fichier fonctions.php sont définies trois fonctions (qu'on ne vous demande pas d'écrire), `print_form()`, `display()` et `print_doctype_html_and_head()`.

La dernière produit tout le code html du début d'un document jusqu'à l'élément head compris.

Ecrire un programme truc.php qui sera chargé avec un paramètre `param` au moyen de la méthode GET. Votre programme effectuera les opérations suivantes :

- inclure le fichier fonctions.php,
- appeller la fonction `print_doctype_html_and_head()`,
- afficher la balise body
- tester si le paramètre `param` est fourni.
- si c'est le cas, appeller la fonction `display()` en lui passant la valeur du paramètre.
- sinon, appeller la fonction `print_form()`.
- pour finir, votre programme fermera toutes les balises nécessaires. (les fonctions `display()` et `print_form()` n'affichent que des éléments complets, donc elles n'ouvrent pas de balises sans les fermer)

IV MySQL + PHP

On souhaite réaliser un site web qui permette à ses utilisateurs d'organiser des événements auxquels les autres utilisateurs peuvent s'inscrire.

La page d'un utilisateur présentera les prochains événements auxquels il participera, suivis des derniers événements auxquels il a participé.

On utilisera une base de données avec trois tables dont les champs seront les suivants :

- Utilisateurs
  - id : Identifiant unique numérique, choisi par le système
  - username : Identifiant unique textuel, choisi par l'utilisateur
  - passwd : Mot de passe
  - prenom : Prénom
  - nom : Nom
  - email : Adresse e-mail
- Evénements
  - id : Identifiant unique numérique, choisi par le système
  - auteur : Identifiant numérique du créateur
  - date : Date à la minute près
  - texte : Descriptif (limité à 140 caractères)
- Participations
  - id : Identifiant unique numérique, choisi par le système
  - eve : Identifiant d'un événement
  - user : Identifiant numérique d'un utilisateur

Pour commencer

1/ Ecrire des requêtes MySQL pour chacune des opérations suivantes.

1. Créer ces tables.
2. Ajouter un utilisateur de nom Jacques Martin, d'identifiant jmartin et d'adresse martin@coldmail.fr
3. Obtenir la liste des tables de la base de données courante.
4. Obtenir la liste des utilisateurs avec leurs deux identifiants, leur mot de passe et leur mail.
5. Obtenir le descriptif de tous les événements créés par l'utilisateur ayant l'identifiant (id) 12.
6. Même chose en demandant le descriptif *et* la date.
7. Obtenir la liste de tous les participants à l'événement ayant l'identifiant 23.

2/ Ecrire un fichier initbase.php qui définit quatre constantes, SERVEUR, BASE, USER et PASSWD, de valeurs respectives localhost, mybase, toto et MdP+0+0, et qui établit la connexion avec le serveur web et charge la base de données.

3/ Ecrire le code html d'un formulaire permettant à un utilisateur de créer son compte.

4/ Ecrire une fonction php `ajoute_utilisateur()` qui exécute la requête permettant d'ajouter un utilisateur dans la table correspondante.

Les pages principales

5/ Ecrire un fichier event.php qu'on chargera (au moyen de la méthode GET) avec comme paramètre l'identifiant (id) d'un événement, et qui affichera une page html contenant le nom de son créateur suivi de la liste des participants.

6/ Que se passera-t-il si la liste est très longue ? Proposer une solution.

7/ Ecrire une fonction php `msg2html(date, texte)` qui prend en paramètre la date et le descriptif d'un événement et renvoie dans une chaîne de caractères du code html destiné à afficher ces deux informations (la fonction renverra le code html mais *sans l'afficher*).

8/ Décrire brièvement sans l'écrire ce que pourrait faire le fichier user.php selon le même principe que event.php (préciser les requêtes MySQL nécessaire, en particulier on aimerait que pour chaque événement affiché, le nombre de participant soit indiqué).

Pour aller plus loin

9/ Permettre à un utilisateur de saisir du texte, comme le descriptif d'un événement, qui sera ensuite affiché tel quel par un script php pose un problème de sécurité. Lequel ?

10/ La fonction php `htmlspecialchars()` prend une chaîne de caractères et convertit en entités html tous les caractères qu'elle contient et qui ont un sens spécial pour html. Donner deux utilités possibles à cette fonction.

11/ Expliquer pourquoi la fonction `htmlspecialchars()` doit être utilisée non seulement pour le descriptif des événements mais aussi pour les informations fournies lors de la création du compte utilisateur (nom, prénom...)

12/ Proposer (en une dizaine de lignes) une méthode pour éviter à un utilisateur de fournir son identifiant et son mot de passe à chaque fois qu'il accède à une page du site.

13/ Dans user.php, on aimerait que pour chaque événement, cliquer sur le nombre de participants en fasse apparaître la liste (éventuellement tronquée si elle est longue) sans changer de page. Quel avantage présente javascript sur php pour réaliser cette opération ? Décrire brièvement comment vous feriez.

Annexe pour la partie II

html

```
<body>
  <p>
    Texte texte Texte texte Texte texte
        texte Texte texte Texte texte texte
  </p>

  <h1>Titre</h1>
  <p>
    Toute l'astuce consiste a voir comment
        se positionne le menu par
    rapport au reste du body, en
        particulier les h1.
  </p>

  <h1>Titre</h1>
  <p>
    Texte texte Texte texte Texte texte
        texte Texte texte Texte texte texte
  </p>

  <div id="menu">
    <h1>Menu</h1>
    <ul>
      <li>A</li>
      <li>B</li>
      <li>C</li>
    </ul>
  </div>
</body>
```

capture d'écran

Les éléments à implémenter comprennent le titre (**photo**, en h1), la photo et le texte en dessous.

Photo

cette partie se situe
dans le bloc de menu
et la photo se trouve
dans le fichier
photo.jpg

CSS

```
html, body {
        margin: 0;
        padding: 0;
        font-size: 100%;
}

h1 {
    background-color: yellow;
    text-align: center;
}

p {
    margin: 1ex 9em 1ex 1em;
    padding: 0.5ex 40px 0.5ex 35px;
}

#menu {
    position: fixed;
    right: 15px;
    width: 9em;
    top: 0px;
    height: 100%;
    padding: 0.5em;
    border-left: solid medium black;
    border-right: solid medium black;
    background-color: white;
}

#menu h1 {
    text-align: left;
    font-size: 1em;
    background-color: white;
}

#menu ul {
    list-style-type: none;
    margin: 0.5em;
    padding: 0;
}
```