

CT JS/PHP : Session 1 - 2h - Sur machine

Téléchargez et décompressez l'archive déposée sur Moodle pour cet examen. Le dossier résultant contient différents fichiers à réutiliser ou à compléter. A l'issue du temps imparti, archivez votre dossier et déposez l'archive sur Moodle. Pour visualiser ce qui est attendu, consultez ce [démonstrateur](#).
Le sujet comporte deux parties (PHP et JS). Les exercices sont indépendants les uns des autres.

Partie 1 : PHP

Le fichier `_courses.xml` décrit une maquette de formation. La formation se compose de cours (élément `course`), chaque cours comporte une ou plusieurs parties (élément `part`) et chaque partie est enseignée à un ou plusieurs groupes (élément `group`). Cours, parties et groupes sont nommés (attribut `name`) et cours et parties peuvent être étiquetés (attribut `label`). Chaque partie définit le nombre de séances dispensées pour chacun de ses groupes (attribut `nrSessions`). Chaque groupe a un seuil d'effectif maximum (attribut `maxHeadCount`) et certains groupes peuvent avoir un groupe parent (attribut `parent`).

Exercice 1. (XML) Compléter le fichier `xml.php` pour convertir le fichier `_courses.xml` sous forme de tableau PHP. Le tableau attendu est défini dans le fichier `_courses_parts_groups.php`. Ignorer les attributs `label` et `parent` lors de la conversion et prendre soin de convertir les valeurs d'attributs XML au bon type PHP.

Exercice 2. (Objets) Le fichier `_classes.php` définit 3 classes PHP modélisant cours, parties et groupes et leurs liens de composition. En s'appuyant sur ces classes, compléter le fichier `objects.php` pour convertir les cours, parties et groupes figurant dans le tableau défini dans le fichier `_courses_parts_groups.php` en 3 tableaux d'objets `$courses`, `$parts` et `$groups`. Veiller à composer les objets après construction (p. ex., chaque partie stocke ses groupes).

Exercice 3. (Filtres, tris, transformations) Le fichier `parts_groups.php` définit le tableau des parties et de leurs groupes. Modifier ce tableau en 3 étapes :

- Eliminer toute partie dont le nom (clé) contient la chaîne `Eval` ;
- Trier les parties en nombre de séances décroissant ou, en cas d'égalité, en nombre de groupes croissant ;
- Remplacer dans chaque partie le tableau de clé `groups` par le nombre de groupes `y` figurant.

Le tableau attendu est défini dans le fichier `_parts_groups.php`.

Exercice 4. (Formulaire et base de données) Importer sous phpMyAdmin le script SQL qui créera la base de données `l3info_ct_21_1_courses.sql`. La base comporte 2 tables (voir Figure 1).

La table `PARTIE` est dotée des colonnes suivantes :

- `id` : identifiant de la partie (clé primaire auto-incrémentée)
- `name` : nom de la partie (`varchar(50)`)
- `nrSessions` : nombre de séances par groupe de la partie (`int(2)`).

La table `GROUPE` est dotée des colonnes suivantes :

- `id` : identifiant du groupe (clé primaire auto-incrémentée)
- `name` : nom du groupe (`varchar(50)`)
- `maxHeadCount` : seuil d'effectif maximum du groupe (`int(2)`)
- `part` : identifiant de la partie du groupe (clé étrangère).

Le formulaire `form.php` doit permettre de sélectionner une partie et d'y ajouter un groupe. Compléter ce fichier comme indiqué ci-dessous en adaptant au préalable la fonction de connexion figurant dans le fichier `connexion.php` :

- Générer les options du menu déroulant permettant la sélection d'une partie : la valeur HTML de chaque option correspond au champ `id` de la partie, le texte affiché est le champ `name` dans lequel les tirets sont remplacés par des espaces ;

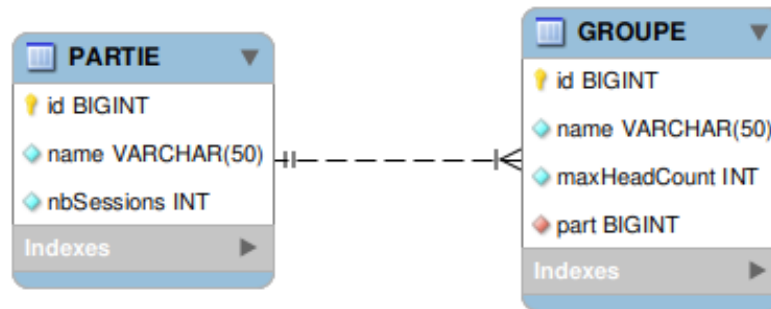


FIGURE 1 – Modèle de la base de données.

- A la soumission du formulaire, insérer dans la base le groupe demandé pour la partie sélectionnée.
- Réafficher ensuite le formulaire suivi d'un tableau à 1 en-tête et 2 colonnes : l'en-tête contient une info-bulle contenant le nom de la partie et son nombre de séance (`nrSessions`), chaque ligne contient le nom du groupe en première colonne (`name`) et son seuil d'effectif en seconde colonne (`maxHeadCount`).

La figure 2 illustre le formulaire au chargement et après soumission.

La figure 2 présente deux versions du formulaire "Ajout de groupe".

À gauche (chargement) : Le formulaire contient un menu déroulant "Partie de cours :" avec une liste de parties. "Databases Lab" est sélectionné. Les autres champs sont vides.

À droite (après soumission) : Le menu "Partie de cours :" est maintenant "English Les". Le champ "Nom de groupe :" est vide. Le champ "Effectif maximum :" est réglé sur 10. Un bouton "Ajouter" est visible.

En dessous du formulaire, un tableau intitulé "Groupes et effectifs" affiche les données :

Groupes et effectifs	
Datab	Databases-Lab - 7 séances
Databases-Lab-2	20
Databases-Lab-3	20
Databases-Lab-New	12

FIGURE 2 – Ajout du groupe Databases-Lab-New de taille 12 à la partie Databases-Lab comportant 7 séances par groupe

Partie 2 : Javascript

Le fichier `js.html` présente un formulaire suivi d'un tableau à 3 colonnes comportant cours, parties de cours et groupes de parties. Les cellules du tableau sont classées en conséquence dans le code HTML. Le fichier HTML importe le fichier `js.js`. Compléter ce dernier sans modifier le premier pour répondre aux questions.

Exercice 5. (Suppression de colonne) Au chargement de la page, supprimer la première colonne du tableau (colonne des cours).

Exercice 6. (Renommage de groupe) Lorsque le visiteur clique sur une cellule de groupe, proposer une invite (`prompt`) lui permettant de renommer le groupe tel qu'illustré en Figure 3.

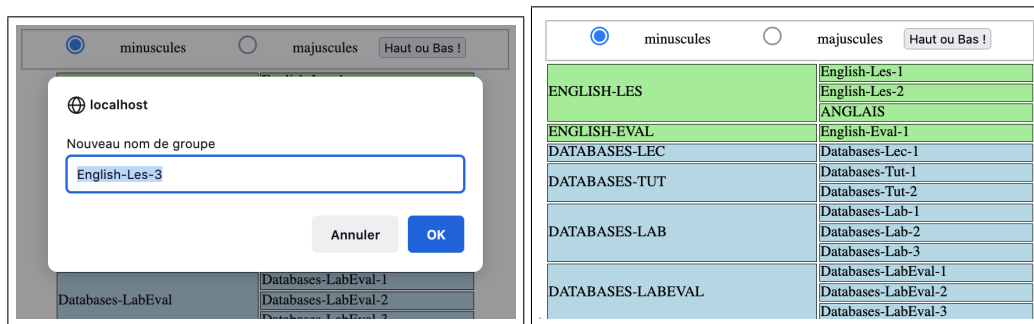


FIGURE 3 – Renommage du groupe English-Les-3 en ANGLAIS

Exercice 7. (Casse) Lorsque le visiteur clique sur l'un des 2 boutons radio, mettre le texte des cellules de parties en minuscules ou majuscules tel qu'illustré en Figure 4.

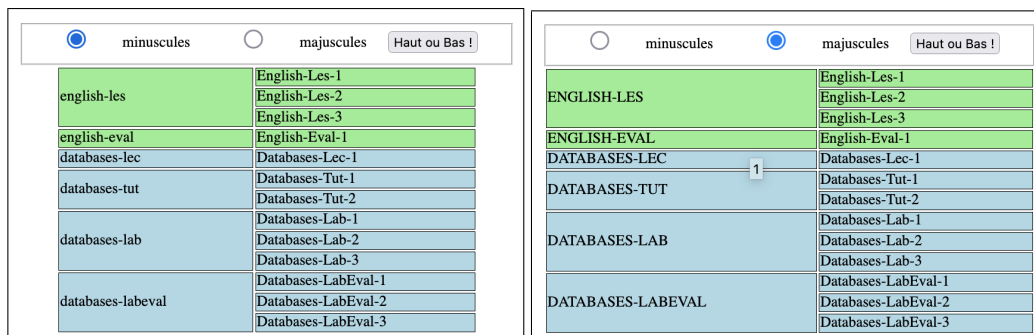


FIGURE 4 – Mise en minuscules (figure de gauche) ou majuscules (figure de droite) des parties de cours

Exercice 8. (Déplacement) Au chargement de la page, cloner le formulaire et placer la copie en bas de page tout en la rendant invisible (propriété CSS `visibility`). Lorsque le visiteur clique sur le bouton `Haut ou Bas !`, masquer le menu du haut (resp., du bas) s'il est affiché et faire apparaître celui du bas (resp., du haut) tel qu'illustré en Figure 5.

ENGLISH-LES	English-Les-1
	English-Les-2
	English-Les-3
ENGLISH-EVAL	English-Eval-1
DATABASES-LEC	Databases-Lec-1
DATABASES-TUT	Databases-Tut-1
	Databases-Tut-2
DATABASES-LAB	Databases-Lab-1
	Databases-Lab-2
	Databases-Lab-3
DATABASES-LABEVAL	Databases-LabEval-1
	Databases-LabEval-2
	Databases-LabEval-3
WEB-DEVELOPMENT-LEC	Web-Development-Lec-1
WEB-DEVELOPMENT-LAB	Web-Development-Lab-1
	Web-Development-Lab-2
	Web-Development-Lab-3
WEB-DEVELOPMENT-LABEVAL	Web-Development-LabEval-1
	Web-Development-LabEval-2
	Web-Development-LabEval-3
COMPUTER-GENERATED-IMAGES-LEC	Computer-Generated-Images-Lec-1
COMPUTER-GENERATED-IMAGES-LAB	Computer-Generated-Images-Lab-1
PYTHON-LES	Python-Les-1
PYTHON-LAB	Python-Lab-1
	Python-Lab-2
INTELLIGENT-ALGORITHMS-LEC	Intelligent-Algorithms-Lec-1
INTELLIGENT-ALGORITHMS-TUT	Intelligent-Algorithms-Tut-1
INTELLIGENT-ALGORITHMS-LAB	Intelligent-Algorithms-Lab-1
	Intelligent-Algorithms-Lab-2
	Intelligent-Algorithms-Lab-3
QT-LEC	Qt-Lec-1
QT-LAB	Qt-Lab-1
	Qt-Lab-2
FUNCTIONAL-PROGRAMMING-LEC	Functional-programming-Lec-1
FUNCTIONAL-PROGRAMMING-TUT	Functional-programming-Tut-1
FUNCTIONAL-PROGRAMMING-LAB	Functional-programming-Tut-2
	Functional-programming-Lab-1
	Functional-programming-Lab-2
LOGIC-PROGRAMMING-LEC	Functional-programming-Lab-3
	Logic-programming-Lec-1
	Logic-programming-Tut-1
LOGIC-PROGRAMMING-TUT	Logic-programming-Tut-2
LOGIC-PROGRAMMING-LAB	Logic-programming-Lab-1
	Logic-programming-Lab-2
	Logic-programming-Lab-3
<div><div><input checked="" type="radio"/> minuscules</div><div><input type="radio"/> majuscules</div><div>Haut ou Bas !</div></div>	

FIGURE 5 – Bascule du formulaire de haut en bas (ou inversement) au clic sur Haut ou Bas !