

le **cnam** Développement Web coté le **cnam**  
serveur (NFA017)

première session  
Corrigé de l'examen de Juin 2017

---

**Durée :** 3h

**Modalités :** Tous documents papiers autorisés.

---

Vous voulez gérer vos photos. Vous disposez déjà de plein de photos dans un repertoire **images** et vous avez déjà exécuter le code suivant :

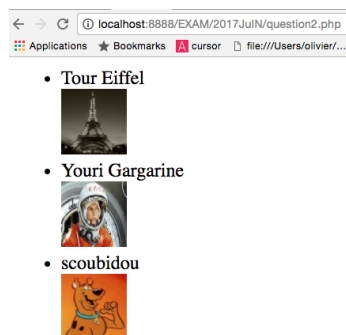
```
1 <?php
2 $con=mysqli_connect("localhost","root","root")
3     or die("Erreur".mysqli_connect_error());
4
5
6 mysqli_query($con,"CREATE DATABASE IF NOT EXISTS exam_2017")
7     or die("Erreur :".mysqli_error($con));
8
9 mysqli_select_db($con,"exam_2017");
10
11 $sql="CREATE TABLE IF NOT EXISTS Photo(
12     id int NOT NULL AUTO.INCREMENT,
13     titre varchar(20) NOT NULL,
14     url varchar(100) NOT NULL ,
15     PRIMARY KEY (id));";
16
17
18 mysqli_query($con,$sql)
19     or die("Erreur ".mysqli_error($con));
20
21 $sql="INSERT INTO Photo (id,titre,url)
22     VALUES(null,"Tour Eiffel", "images/eiffel.png");
23
24 mysqli_query($con,$sql)
25     or die("Erreur ".mysqli_error($con));
26
27 /// .... autre insert ...
28
29 $sql="INSERT INTO Photo (id,titre,url)
30     VALUES(null,"Youri_Gargarine", "images/gagarine.png");
31
32 mysqli_query($con,$sql)
33     or die("Erreur ".mysqli_error($con));
34
35 mysqli_close($con);
36 echo "tout est ok";
37 ?>
```

1. Expliquer en quelques lignes ce que fait ce code (2pt).

**corrigé :**

- lignes 2 et 3, on essaye de se connecter au serveur de base de données qui tourne sur la machine *localhost*, avec le login *root* et le mot de passe *root*<sup>1</sup>. Si la connexion est un succès, l'un objet renvoyé par `mysqli_connect`, qui représente la connexion au serveur MySQL, est stocké dans la variable `$con`. Si elle échoue le message "*Erreur raison de l'erreur*" est affiché et le programme termine.
- lignes 6 et 7, on essaye de créer une base *exam\_2017* sur le serveur. si cela échoue le message "*Erreur raison de l'erreur*" est affiché et le programme termine.
- ligne 9, on sélectionne la base *exam\_2015* comme base de données par défaut pour les requêtes
- lignes 11 à 16, on met dans la variable `$sql` le code *SQL* nécessaire à la création d'une base *Phoo* à 3 champs *id*, *titre*, *url*, *ine* ou *id* est la clef primaire.
- lignes 18 et 19, on essaie d'exécuter la requête précédente sur le serveur. Si cela échoue, le message "*Erreur raison de l'erreur*" est affiché et le programme termine.
- lignes 21 et 22 on met dans une variable `sql` le code nécessaire à l'insertion d'une photo de titre *Tour Eiffel*, et d'URL *images/eiffel.png*.
- lignes 24 et 25 on tente d'exécuter ce code.
- lignes 27 à 37 autres insertions.
- ligne 35, on ferme la connexion au serveur de base de données.
- ligne 36, on affiche le message `< tout est ok >`

2. Écrivez un programme PHP qui affiche la liste des photos de la base, sous forme d'une liste HTML comme ci-dessous (3pt) :



**corrigé :** Vous étiez libre d'utiliser *mysqli* ou *PDO*. Dans la suite nous utilisons le premier.

Comme on va devoir faire de fréquentes connexions à la base de données, on peut définir une fonction de connexion dans un fichier *utils.php* qui sera chargé dans les autres fichiers.

Listing 1 – *utils.php*

```
1 <?php
2 function connection(){
3     $con=mysqli_connect("localhost","root","root") or
4         die("connexion localhost impossible");
5     mysqli_select_db($con,"exam_2017");
6     return $con;
7 }
8 ?>
```

---

1. d'une manière générale il faut mieux créer un autre utilisateur que *root*, ayant des droits restreints et utiliser un mot de passe conséquent

---

Listing 2 – question2.php

---

```
1 <?php
2 require "utils.php";
3
4 //rechercher
5 $con=connection();
6 $sql="SELECT * from Photo";
7 $res= mysqli_query($con,$sql) or die("Erreur :".mysqli_error($con));
8
9 echo "<ul>";
10 while($ligne=mysqli_fetch_assoc($res)){
11     echo "<li>";
12     echo $ligne["titre"]."<br/>";
13     echo "<img width='55em' height='55em' src='". $ligne["url"]."' />";
14     echo "</li>";
15 }
16 echo "</ul>";
17
18
19 mysqli_close($con);
20 ?>
```

---

3. Pour organiser vos photos vous voulez pouvoir marquer chaque photo avec des labels. Un label est une chaîne d'au plus 20 caractères. Une photo peut avoir plusieurs labels et un label peut être associé à plusieurs photos. Il peut y avoir des photos sans label et des labels non associés à des photos.

Écrivez le code PHP permettant de créer les 2 tables nécessaires. *Label* (avec au moins les champs id, et label) et *Label\_Photo* (avec au moins les champs idPhoto et idLabel) (2pt).

**corrigé :**

---

Listing 3 – question3.php

---

```
1 <?php
2 require "utils.php";
3 $con=connection();
4
5 $sql="CREATE TABLE IF NOT EXISTS Label(
6     id int NOT NULL AUTO.INCREMENT,
7     label varchar(20) NOT NULL ,
8     PRIMARY KEY (id)
9     )";
10
11 mysqli_query($con,$sql) or
12     die("Erreur". $con->error);
13
14 echo "Label done <br>";
15
16 $sql="CREATE TABLE IF NOT EXISTS Label_Photo(
17     idPhoto int ,
18     idLabel int
19     )";
20
21 mysqli_query($con,$sql) or
```

```

22         die("Erreur".$con->error);
23
24     echo "Label_Photo done <br>";
25     mysqli_close($con);
26
27 ?>

```

---

4. Écrivez une fonction PHP `insert_label_type` qui recherche un label dans la table `Label` et l'insert s'il n'est pas présent (vous pouvez faire plusieurs requêtes).

La fonction renverra dans tous les cas l'**id** associé au label (2pt).

Pour obtenir l'id du dernier element inséré vous pouvez utiliser :

— avec *mysqli*

`mysqli_insert_id ($connexion)` ou `$connexion` est le lien de connexion renvoyé par `mysqli_connect`

— avec *PDO*

`$connexion->lastInsertId()` ou `$connexion` est le résultat de `new PDO(...)`

**corrigé :**

Listing 4 – question4.php

---

```

1  <?php
2
3  require_once "utils.php";
4  function insert_label($lab){
5      $id;
6      $con=connection();
7      //rechercher le label
8      $sql="SELECT * from Label where label=\"\$lab\"";
9      $res=mysqli_query($con,$sql);
10
11     if($line=mysqli_fetch_assoc($res)){
12         $id=$line["id"];
13         echo "$lab déjà présent. id : $id";
14     }
15     else
16     {
17         echo "insertion du label :$lab<br/>";
18         $sql="INSERT INTO Label (id,label) VALUES (null, \"\$lab\")";
19
20         mysqli_query($con,$sql) or die("erreur".$con->error);
21         $id=mysqli_insert_id($con);
22         echo "fait , $lab id: $id <br/>";
23     }
24     mysqli_close($con);
25     return $id;
26 }
27
28 //pour tester decommenter
29
30 //insert_label("chien");insert_label("cosmonaute");insert_label("chien");
31
32
33 ?>

```

---

5. L'ami photographe, qui vous a fourni les premières photos (présentes dans le repertoire images et référencées dans la table *Photo*, vous envoie un marquage de ces photos sous la forme d'un fichier texte nommé `descriptif.txt`. Les lignes de ce fichier ont le format : `nomDuFichierImage#label#label2#label3...` c'est-à-dire qu'elle sont constituées du nom d'un fichier image (attention pour en déduire l'url il faudra rajouter le préfixe « images »), suivi d'au moins un # et un label ; d'autres labels peuvent suivre séparés par des #.

Remplir les tables *Label* et *Photo\_Label* avec les données de ce fichier (3pt).

Par exemple **SI** le fichier contenait :

```
gagarine.png#cosmonaute#hero#russe
scoubidou.png#chien
eiffel.png#tour#monument
```

alors, vous devriez :

- (a) insérer (si ils n'y sont pas déjà) les labels `cosmonaute`, `hero`, `russe`, `chien`, `tour` et `monument` dans la table *Label* , faire le lien dans la table *Label\_Photo* entre les identifiants de chacun de ses labels avec les identifiants correspondant aux photo dont le nom est passé. par exemple lié l'id associe a `cosmonaute` à l'id de la photo dont l'url est `images/gagarine.png`.

Vous pouvez utiliser la fonction de la question précédente même si vous n'avez pas su l'écrire.

**corrigé :**

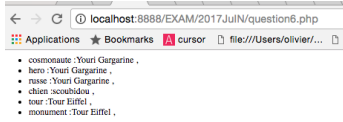
Listing 5 – question5.php

---

```
1 <?php
2 require_once "utils.php";
3 require_once "question4.php";
4
5 $con=connection();
6
7 $d=fopen("descriptif.txt","r");
8
9 while ($ligne=fgets($d)){
10     $tag=explode("#",$ligne);
11     var_dump($tag);
12     //trouver l'idPhoto
13     $sql="SELECT id FROM Photo where url='images/' . $tag[0].' '";
14     echo "SQL $sql <br/>";
15     $res=mysqli_query($con,$sql) or die("Erreur : ".mysqli_error($con));
16     $idPhoto=mysqli_fetch_assoc($res)["id"];
17     echo "trouvé $idPhoto ";
18     for($i=1;$i<count($tag);$i++){
19         echo "insertion ". $tag[$i]. "<br/>";
20         $idLabel=insert_label($tag[$i]);
21         echo "idLabel $idLabel <br/>";
22         //insert idLabel idPhoto
23         $sql="INSERT INTO Label_Photo (idPhoto,idLabel) VALUES($idPhoto,$idLabel)";
24         mysqli_query($con,$sql) or die("Erreur : ".mysqli_error($con).
25         $sql");
26     }
27 }
28 fclose($d);
29 ?>
```

---

6. Écrivez le code PHP qui affiche, la liste des labels chacun étant suivi des titres des photos associées. Comme dans l'exemple ci-dessous : (3pt).



corrige :

Listing 6 – question6.php

```
1 <?php
2
3     require_once "utils.php";
4
5     $con=connection ();
6     //rechercher le label
7     $sql="SELECT label , titre from Label ,Label_photo ,Photo  where Label.id=IdLabel
8     $res=mysqli_query ($con , $sql );
9     $VUE=array ();
10    while ($line=mysqli_fetch_assoc ($res)){
11        $VUE[ $line [ "label" ] ][] = $line [ "titre" ];
12    }
13    echo "<ul>";
14    foreach ($VUE as $label=>$titres){
15        echo "<li> $label :";
16        foreach ($titres as $titre){
17            echo "$titre ,";
18        }
19    }
20    echo "</ul>";
21
22    mysqli_close ($con);
23
24 ?>
```

7. On veut pouvoir ajouter des photos par un formulaire. Il faudra uploader une photo, insérer un titre et l'url d'upload dans la table *Photo*, ajouter un label et son type dans la table *Label* et mettre à jour *Photo\_Label* en conséquence.

Écrivez le code nécessaire à cela (vous pouvez faire un ou deux fichiers (formulaire plus traitement). (4pt)

corrige :

8. On dispose d'un fichier PHP *Publication.php* contenant le code PHP suivant :

```
1 <?php
2 class Publication {
3     public $url;
4     public $texte;
5
6
```

```

7     function __construct($url,$text){
8         $this->url=$url;
9         $this->texte;
10    }
11
12    public function publish($cible){
13
14        switch ($cible){
15            case "facebook": echo "publication sur Facebook";break ;
16            case "twitter": echo "publication sur Twitter";break;
17            case "instagram": echo "publication sur Instagram";break;
18            default:echo "$cible pas encore gérée";
19
20        }
21    }
22    ?>

```

En utilisant cette classe, écrivez le code PHP nécessaire pour publier sur Twitter une photo dont on donnera le titre, accompagné d'un texte. Le titre et le texte sont supposés passés par GET et on ne demande pas le formulaire.(2pt)

**corrigé :**

Listing 7 – question8.php

---

```

1  <?php
2
3  require_once "utils.php";
4  require_once "Publication.php";
5
6  if (empty($_GET["titre"]) || empty($_GET["texte"])){
7      die("donnez le titre et le texte associé");
8  }
9      $con=connection();
10 $titre=mysqli_real_escape_string($con, $_GET['titre']);
11 //rechercher le texte et l'url
12 $sql="SELECT url FROM Photo where titre='".$titre'";
13
14 $res=mysqli_query($con,$sql);
15
16 if($line=mysqli_fetch_assoc($res)){
17     $url=htmlspecialchars($line["url"]);
18     $texte=htmlspecialchars($_GET["texte"]);
19     $pub=new Publication($url,$texte);
20     $pub->publish("twitter");
21
22 }else{die("titre non trouvé");}
23
24 mysqli_close($con);
25
26
27
28 ?>

```

---