



UNIVERSITE SIDI MOHAMED BEN ABDELLAH
FACULTE DES SCIENCES ET TECHNIQUES



Licence Sciences et Techniques (LST)
CALCUL SCIENTIFIQUE ET APPLICATIONS

MEMOIRE DE FIN D'ETUDES
Pour l'obtention du Diplôme de Licence Sciences et Techniques

Gestion de la Bibliothèque de la FLSH FES-SAÏS

Présenté par :

◆ RHCHIM Abdellah

Encadré par :

◆ Pr. ETTAOUIL Mohamed (FST de Fès)

Soutenu Le 17 Juin 2011 devant le jury composé de:

- Pr. ETTAOUIL Mohamed (FST de Fès)
- Pr. ELMOUTAOUAKIL Karim (FST de Fès)
- Pr. GHANOU Youssef (EST de Meknès)
- Pr. LOQMAN Chakir (EST de Meknès)

Stage effectué à la Faculté des lettres et des Science humaine Saïs - Fès

Année Universitaire 2010 / 2011

FACULTE DES SCIENCES ET TECHNIQUES FES – SAÏS


☒ B.P. 2202 – Route d'Imouzzer – FES


☎ 212 (0)5 35 61 16 86 – Fax : 212 (0)5 35 60 82 14


Site web : <http://www.fst-usmba.ac.ma>


Dédicace

Je dédie ce travail

-  A l'hommage de mes chères parents et à toute ma famille le symbole de ma réussite et qui m'ont apporté de l'aide durant toute cette période de stage pour leurs affection et amour, leur patience et confiance à qui j'exprime mon respect et mon amour dévoué.

-  A monsieur ETTAOUIL Mohamed mon encadrant durant la période de stage qui m'a formé accompagné tout au long de cet expérience avec beaucoup de patience et pédagogie

-  A tous le personnel de service de la bibliothèque de la FLSH SAIS qui ont participé de près ou de loin pour la réussite de projet en particulier Mr. Mohamed Hajji

-  A Ms. LAZAAR Mohamed et tous les membres de laboratoire de Modélisation et Calcul scientifique pour leurs efforts et orientations

REMERCIEMENT

J'exprime toute ma gratitude à monsieur le professeur ETTAOUIL Mohamed, pour l'effort fourni, les conseils prodigués, sa patience et sa persévérance dans le suivi.

Je remercie très sincèrement, les membres de jury, Mr. GHANOU Youssef, Mr. LOQMAN Chakir et Mr. ELMOUTAOUKIL Karim d'avoir bien voulu accepter de faire partie de la commission d'examineur.

Je tiens à remercier aussi l'ensemble du personnel de service bibliothèque de la faculté des lettres et des sciences humaines SAIS-FES, en particulier le responsable Mr. Mohamed Hajji

J'adresse également mes remerciements, à tous mes enseignants, qui m'ont donnée les bases de la science.

Je remercie a l'occasion, toute personne qui a participé de près ou de loin pour l'accomplissement de modeste travail.

TABLES DE MATIÈRES :

Introduction	5
Chapitre 1 : Présentation de la FLSH Sais-Fès	6
I. Introduction	7
II. Formations	7
III. Bibliothèque	10
Chapitre 2 : Base de données	11
I. Concepts de base de données	12
II. Système de gestion de base de données	13
III. Architecture général d'une base de données	15
IV. Les différents modèles de base de données	17
V. Les langages des bases de données	19
Chapitre 3 : Langage PHP	28
I. Généralités sur les langages informatiques	29
II. Des langages particuliers : les langages du web	31
III. Les bases du langage PHP	35
Chapitre 3 : Application	43
I. Espace étudiant	44
II. Espace administrateur	44
III. Espace bibliothécaires	49
Conclusion	53
Références	54

Introduction

La faculté des lettres et des sciences humaines (FLSH) offre à ses étudiants les conditions nécessaires pour leurs études en citant ainsi l'espace étudiant « la bibliothèque » où ils peuvent améliorer leurs connaissances et préparer leurs examens en servant des ouvrages. Et vue le développement d'informatique reconnu de nos jours la FLSH cherche à informatiser le milieu bibliothèque afin de faciliter les tâches aux étudiants et aux bibliothécaires ce qui est l'intérêt de notre projet. Dans ce travail nous proposons une base de données qui gère la bibliothèque des étudiants et plus précisément nous proposons un système qui gère la recherche, l'ajout, la modification, l'emprunt et la restitution des bouquins.

Pour ce faire, ce mémoire se compose de quatre chapitres :

Dans le premier nous donnons une vision globale des formations offertes par la faculté ainsi que les diplômes délivrés.

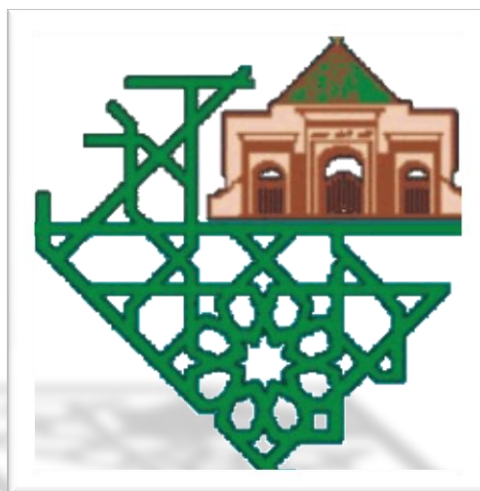
Le deuxième chapitre est consacré aux notions de base de données et son importance dans l'espace professionnel comme l'optimisation du temps.

Dans le troisième chapitre nous donnons une vision générale sur les langages de programmation et la gestion de base de données « PHP/MySQL »

Et finalement dans le quatrième chapitre nous expliquons le fonctionnement de l'application réalisé au cours de ce stage, et à la fin de ce mémoire nous donnons une conclusion.

Chapitre 1

Présentation de la FLSH



I. Introduction

La Faculté des Lettres et des Sciences Humaines Saïs Fès est l'un des établissements de l'Université Sidi Mohamed Ben Abdellah, créée en Octobre 1992, dans le cadre de la décentralisation de l'enseignement supérieur.

Elle draine une population étudiante importante qui totalise un chiffre de 4000 étudiants, issue de toutes les provinces de la Région. Elle accueille aussi des étudiants étrangers appartenant à des pays essentiellement africains.

L'administration de la FLSH Saïs Fès est constituée du professeur **Brahim AKDIM** doyen, de professeur **El houcine OUAZZANI IBRAHIMI** Vice-doyen Chargé de la Recherche Scientifique et de la Coopération, de professeur **Abdelilah SIYAR** Vice-doyen Chargé des affaires Académiques et Pédagogiques, et de professeur **Mohamed Redouane EL AZIFI** Secrétaire Générale ainsi que toute l'équipe administrative qui déploie un effort considérable pour rénover la FLSH soit sur le plan pédagogique ou sur le plan méthodologique.

II. Formation

Afin de réussir sa mission de formation, la faculté des lettres et des sciences humaines Sais-Fès offre à ses étudiants un enseignement semestriel et modulaire dans les nouvelles orientations fondamentales et professionnelles.

Elle procure à l'étudiant des chances d'élargir ses horizons en accédant au système pédagogique LMD (Licence Master Doctorat).

1. Système pédagogique LMD

La réforme LMD, est venue pour répondre à des objectifs que nous pouvons résumer dans les points suivants :

- Améliorer de la qualité de la formation universitaire ;
- Harmoniser notre système de Formation supérieur avec le reste du monde ;
- Proposer des parcours de formation diversifiés et adaptés ;
- Faciliter l'accueil et l'orientation des étudiants, par la mise en place de dispositifs d'accompagnement des étudiants ;
- Favoriser le travail personnel des étudiants ;
- Favoriser l'ouverture de l'Université sur le monde socio économique ;

Ces éléments représentent, la garantie pour réussir le défi d'un enseignement de qualité ouvert au plus grand nombre d'étudiants.

Comme indique son nom le système pédagogique LMD est basé sur trois niveaux essentiels à savoir Licence, Master, et doctorat.

a) Premier niveau : Licence

La filière licence comprenant six semestres d'enseignement sachant qu'une année universitaire est composée de deux semestres ce qui est équivalent à (Bac+3).

Les spécialités disponibles dans la faculté des lettres et des sciences humaines Sais-Fès sont :

- Études Islamiques
- Études françaises
- Géographie
- Histoire et civilisation
- Études Anglaises
- Études Arabes
- Sciences de l'Information et de la Communication
- Sociologie et Anthropologie
- Assistance Social
- Études Amazigh

b) Deuxième niveau : Master

La filière Master comprenant quatre semestres ce qui est équivalent à (Bac+5)

Le master est ouvert aux titulaires d'une licence de formation ou bien d'un diplôme équivalent et d'ailleurs l'admission est proposée par l'équipe pédagogique, spécifiée dans le descriptif de la filière.

c) Troisième niveau : Doctorat

Le niveau Doctorat est équivalent à (Bac+8 au minimum). Il est ouvert aux étudiants titulaires d'un master ou un diplôme équivalent.

Le système pédagogique (LMD) peut représenter de la manière suivante :

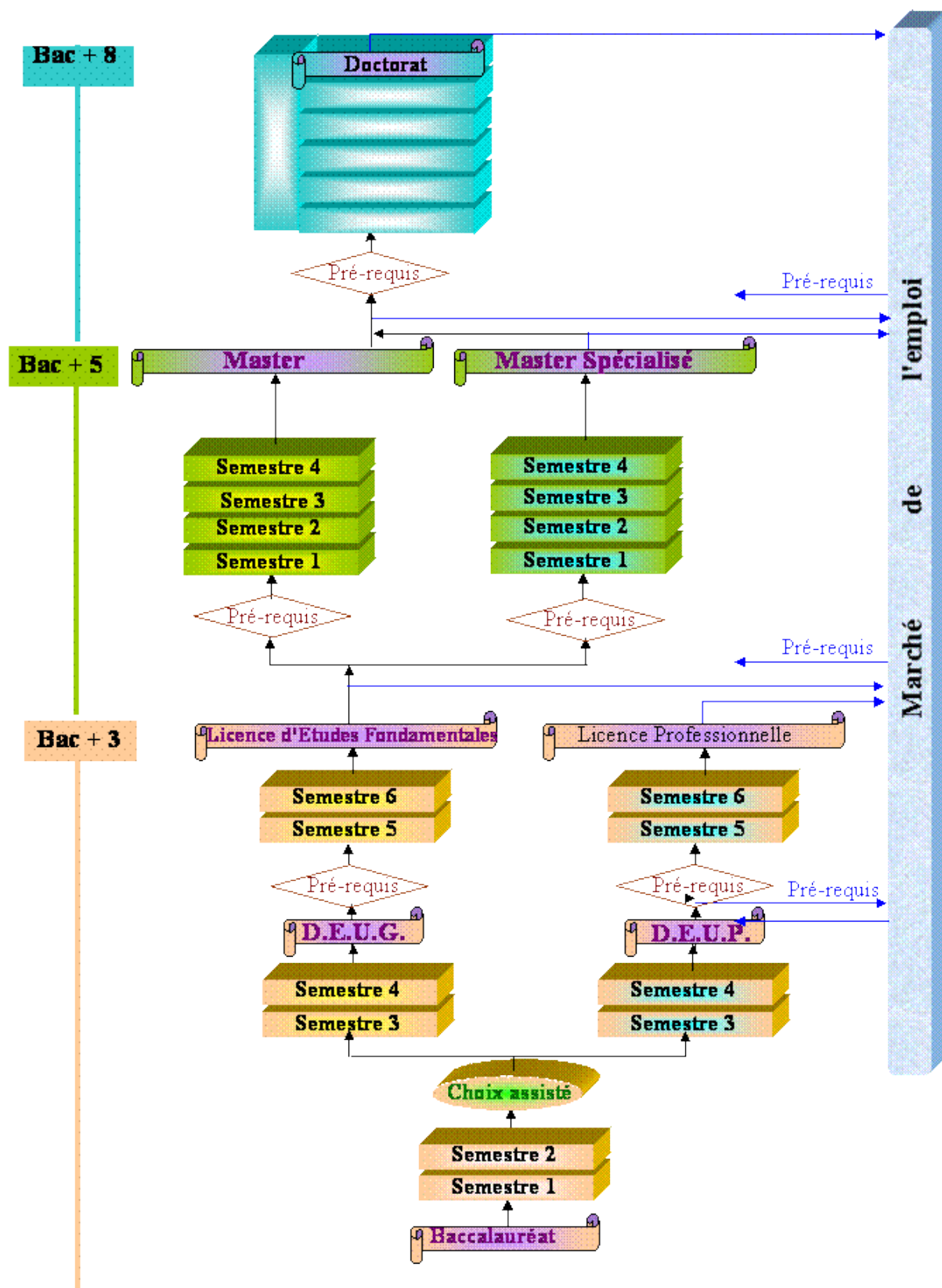


Figure 1 : Représentation du système pédagogique LMD

III. Bibliothèque

Quant à la bibliothèque présente deux services intellectuels un destiné aux étudiants. Cette dernière présente l'espace étudiants où ils passent leur deuxième temps d'étude afin d'améliorer leurs connaissances, leur savoir et de se préparer pour les contrôles continus et les examens en offrant ainsi des moyens convenables à savoir sept postes d'ordinateurs disponibles aux étudiants, ainsi qu'un nombre très important d'ouvrages qui valent 40.000 dans la bibliothèque d'étudiants en couvrant bien sur plusieurs domaines tels que les études islamiques, les études françaises, géographie, histoire et civilisation ...etc.

Afin de réussir leur tâche la bibliothèque s'est dirigée par une forte équipe à leur tête le responsable Mr. Mohamed Hajji et ses collaborateurs (cinq bibliothécaires).

Chapitre 2

Base de Données



I. Concepts de base de données

1. Définition

Une base de données (BDD) est un ensemble structuré de données enregistrées sur des supports informatisés, pouvant satisfaire simultanément plusieurs utilisateurs de façon sélective, en un délai raisonnable et peut-être définie aussi comme une collection structurée et non redondante de données. Elle enregistre les faits ou les événements qui surviennent dans la vie d'un organisme pour les restituer à la demande. On parle généralement de système d'information pour désigner toute la structure regroupant les moyens mis en place pour pouvoir partager des données.

Le schéma suivant peut donner une vision très claire sur la notion de base de données :

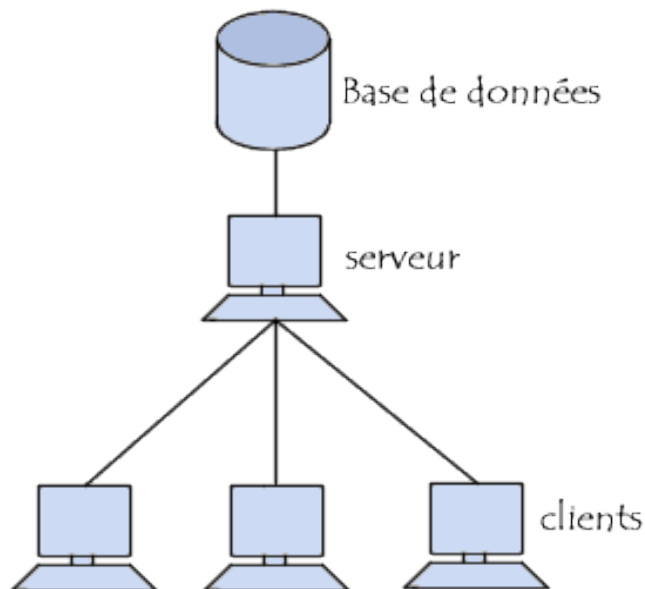


Figure 2 : Représentation d'une base de données

2. L'utilité d'une base de données

Une base de données permet de mettre des informations à disposition d'utilisateurs pour qu'ils puissent les consulter, les modifier et les mettre à jour. Dans une sorte d'exemple une base de données pourrait nous permettre de trouver rapidement les informations : entrons le nom, nous obtenons toutes les informations concernant, générant ainsi une optimisation de temps pour une telle association.

Nous pouvons aussi citer quelque utilité classique d'une base de données à savoir :

a) Partage des données

Une base de données permet de faire partager une même quantité d'information par un ensemble d'utilisateurs. Elle permet à plusieurs utilisateurs par exemple de modifier la même donnée « en même temps ». Ou bien assurer un résultat d'interrogation cohérent pour un utilisateur consultant une table pendant qu'un autre la modifie.

b) Diminution des redondances

La diminution des redondances apparaît au niveau des mises à jour afin d'éviter les incohérences entre les données. Donc chaque donnée ne doit être présentée qu'une et une seule fois dans la base.

c) Structuration

La structuration concerne le choix du modèle de représentation des données et des connaissances. L'objectif de cette modélisation est d'englober les différents modèles spécifiques d'une application ou d'un utilisateur pour obtenir un modèle plus complet de représentation de la réalité.




d) Stock de gros volumes d'informations

Une base de données permet de stocker une grande quantité de données au même temps.




II. Système de Gestion de Base de Données (SGBD) :

1. Définition :

Un système de gestion d'une base de données (SGBD), est l'ensemble des logiciels permettant à l'utilisateur d'interagir avec une base de données ainsi de gérer cette dernière c'est-à-dire :

-  Permettre l'accès aux données de façon simple.
-  Autoriser un accès aux informations à de multiples utilisateurs.
-  Manipuler les données présentes dans la base de données (insertion, suppression, modification ...)

Généralement un système de gestion de base de données (SGBD) se décompose en trois sous systèmes différents à savoir :

-  **Le système de gestion de fichier** : il permet le stockage des informations sur un support physique.
-  **Le système de base de données interne** : il gère l'ordonnancement des informations.
-  **Le système de base de données externe** : il représente l'interface avec l'utilisateur.

Un système de gestion de base de données (SGBD) fonctionne de la manière suivante :

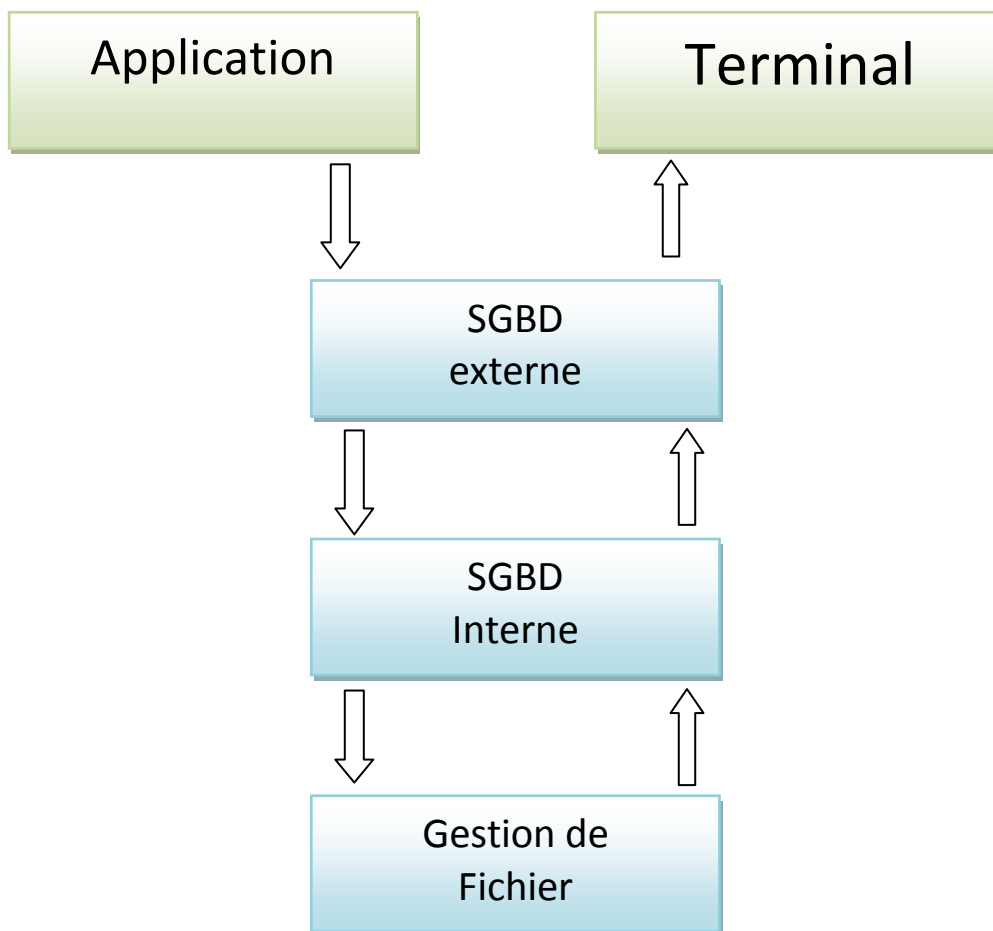


Figure 3 : Fonctionnement d'une base de données

III. Architecture générale d'une base de données

1. Les niveaux correspondent à une base de données

La base de données peut décrire en trois niveaux principaux (ANSI/SPARC) en 1975 à savoir :

- Le niveau conceptuel
- Le niveau interne
- Le niveau externe

a) Le niveau conceptuel

Permet de décrire les différentes entités prise en compte par le modèle (par exemple les ouvrages, les étudiants, la bibliothèque) et les liaisons ou associations entre ces entités. La description du modèle conceptuel comprend la description des structures autorisées (objet et relation entre les objets), ainsi que la description des contraintes imposées sur cette structure. En générale on peut dire que le niveau conceptuel définit l'arrangement des informations au sien de la base de données. Et il est appelé aussi MCD (Modèle Conceptuel des Données) ou MLD (Modèle logique des données)

b) Le niveau interne (ou physique)

Correspond aux structures de stockage des informations, c'est-à-dire la façon selon laquelle sont stockées les données et les méthodes pour y accéder.

c) Le niveau externe

Le niveau externe correspond aux sous-ensembles du modèle général, propres à chaque utilisateur ou groupe d'utilisateurs, appelés vues. Il existe une infinité potentielle de vues pour un modèle général donné.

Les trois niveaux d'une base de données peuvent se présenter alors dans le schéma suivant :

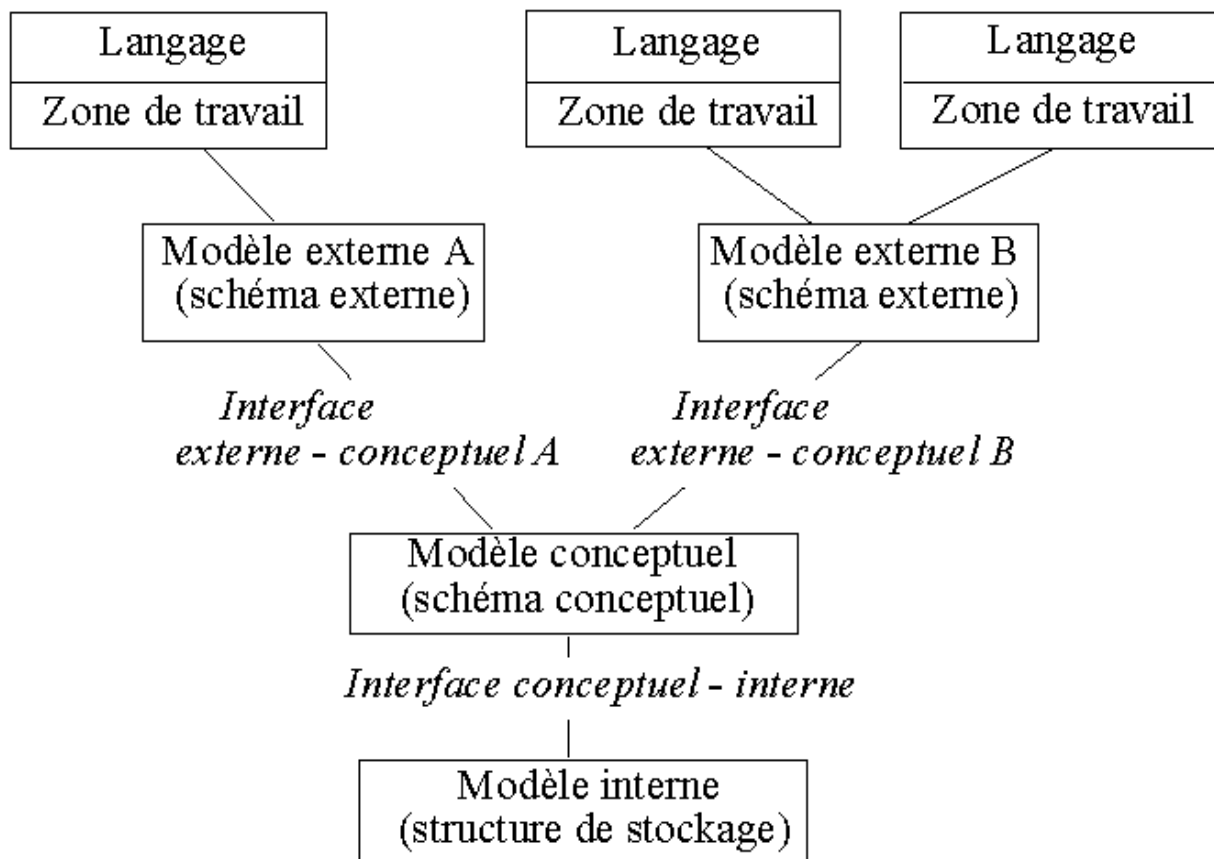


Figure 4 : les 3 niveaux d'une base de données

2. Correspondance entre les niveaux de description des données

L'architecture à trois niveaux définie par le standard ANSI/SPARC permet d'avoir une indépendance entre les données et les traitements. D'une manière générale un SGBD doit avoir les caractéristiques suivantes :

- **Indépendance physique** : le niveau physique peut être modifié indépendamment du niveau conceptuel. Cela signifie que tous les aspects matériels de la base de données n'apparaissent pas pour l'utilisateur, il s'agit simplement d'une structure transparente de représentation des informations
- **Indépendance logique** : le niveau conceptuel doit pouvoir être modifié sans remettre en cause le niveau physique, c'est-à-dire que l'administrateur de la base doit pouvoir la faire évoluer sans que cela gêne les utilisateurs
- **Manipulabilité** : des personnes ne connaissant pas la base de données doivent être capables de décrire leur requête sans faire référence à des éléments techniques de la base de données

- **Rapidité des accès** : le système doit pouvoir fournir les réponses aux requêtes le plus rapidement possibles, cela implique des algorithmes de recherche rapides
- **Administration centralisée** : le SGBD doit permettre à l'administrateur de pouvoir manipuler les données, insérer des éléments, vérifier son intégrité de façon centralisée
- **Limitation de la redondance** : le SGBD doit pouvoir éviter dans la mesure du possible des informations redondantes, afin d'éviter d'une part un gaspillage d'espace mémoire mais aussi des erreurs
- **Vérification de l'intégrité** : les données doivent être cohérentes entre elles, de plus lorsque des éléments font référence à d'autres, ces derniers doivent être présents
- **Partageabilité des données** : le SGBD doit permettre l'accès simultané à la base de données par plusieurs utilisateurs
- **Sécurité des données** : le SGBD doit présenter des mécanismes permettant de gérer les droits d'accès aux données selon les utilisateurs

IV. Les différents modèles de base de données

Les bases de données sont apparues à la fin des années 60, à une époque où la nécessité d'un système de gestion de l'information souple se faisait ressentir. Il existe quatre modèles de SGBD, différenciés selon la représentation des données qu'elle contient :

1. le modèle hiérarchique

Les données sont classées hiérarchiquement, selon une arborescence descendante. Ce modèle utilise des pointeurs entre les différents enregistrements. Il s'agit du premier modèle de SGBD

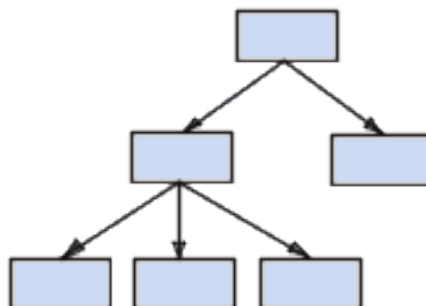


Figure 5 : Le modèle hiérarchique

2. le modèle réseau

Comme le modèle hiérarchique ce modèle utilise des pointeurs vers des enregistrements. Toutefois la structure n'est plus forcément arborescente dans le sens descendant

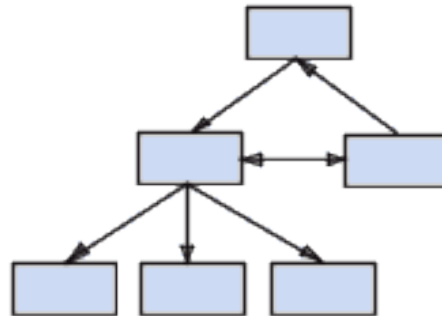


Figure 6 : Le modèle réseau

3. le modèle objet

Le modèle objet (SGBDO, Système de gestion de bases de données objet) : les données sont stockées sous forme d'objets, c'est-à-dire de structures appelées classes présentant des données membres. Les champs sont des instances de ces classes

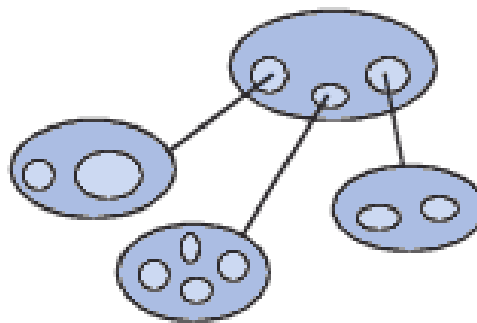


Figure 7 : Le modèle objet

4. le modèle relationnel

Les données sont enregistrées dans des tableaux à deux dimensions (lignes et colonnes). La manipulation de ces données se fait selon la théorie mathématique des relations.

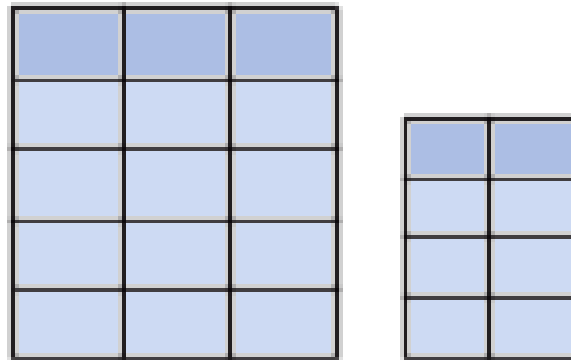


Figure 8 : Le modèle relationnel

V. Les langages des bases de données

Il y a plusieurs systèmes de gestion de base de données dont on cite les plus classiques a savoir :

Quelques SGBD				
Nom SGBD	Année	Editeur	Utilisation type	Licence
<u>Oracle Database</u>	1979	<u>Oracle Corporation</u>	<u>serveur</u>	<u>Licence propriétaire</u>
<u>DB2</u>	1983	<u>IBM</u>	<u>serveur</u>	<u>Licence propriétaire</u>
<u>PostgreSQL</u>	1985	<i>Michael Stonebraker</i>	<u>serveur</u>	<u>Licence BSD</u>
<u>FileMaker Pro</u>	1985	<i>FileMaker</i>	<u>logiciel applicatif</u>	<u>Licence propriétaire</u>




<u>SQL Server</u>	1989	<u>Microsoft</u> ⁶	<u>serveur</u>	<u>Licence propriétaire</u>
<u>Access</u>	1992	<u>Microsoft</u>	<u>L4G</u>	<u>Licence propriétaire</u>
<u>MySQL</u>	1995	<u>Oracle Corporation</u> ⁸	<u>serveur</u>	<u>GPL</u>
<u>SQLite</u>	2000	<i>D. Richard Hipp</i>	<u>composant logiciel</u>	<u>Domaine public</u>
<u>OpenOffice.org Base</u>	2002	<u>Oracle Corporation</u> ⁹	<u>Logiciel applicatif</u>	<u>LGPL</u>

Figure 9 : Quelques systèmes de gestion de base de données

Dans notre stage on va travailler avec MySQL

1. Vocabulaire

L'utilisation d'une base de données ainsi que le système de gestion de base de données apport toujours de nouveau terme on cite alors quelque un qui sera fréquemment utilisée en MySQL :

-  **Champ** : c'est une information nécessaire sur personne, ou une chose, ou un événement par exemple nom, prénom, CNE.
-  **Enregistrement** : un regroupement des champs qui décrivent une personne, ou une chose par exemple les informations d'un étudiant (nom, prénom, CNI, CNE ...)
-  **Table** : c'est un regroupement des enregistrements par exemple étudiants, ouvrages, restitution

2. Langage SQL

SQL (sigle de Structured Query Language) est un langage informatique normalisé qui sert à effectuer des opérations sur des bases de données. La partie langage de manipulation de données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données.

En plus du langage de manipulation de données, la partie langage de définition de données permet de créer, et de modifier l'organisation des données dans la base de données, la partie langage de contrôle de transaction permet de commencer et de terminer des transactions, et la partie langage de contrôle de données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes.

Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des systèmes de gestion de bases de données (SGBD) du marché.

Voici des exemples :

○ Modification d'un enregistrement

Code : SQL

1	UPDATE etudiants SET nom='sami',prenom='abdellah' WHERE cne=2847546547
---	---

○ Recherche dans une table

Code : SQL

1	SELECT nom,prenom,CNE FROM etudiants ORDER BY CNE WHERE nom='ahmed'
---	---

○ Suppression d'un enregistrement

Code : SQL

1	DELETE FROM etudiants WHERE cne=295245784
---	---

3. SGBD MySQL

MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server.

MySQL consiste en un ensemble de programmes chargés de gérer une ou plusieurs bases de données, et qui fonctionnent selon une architecture client/serveur

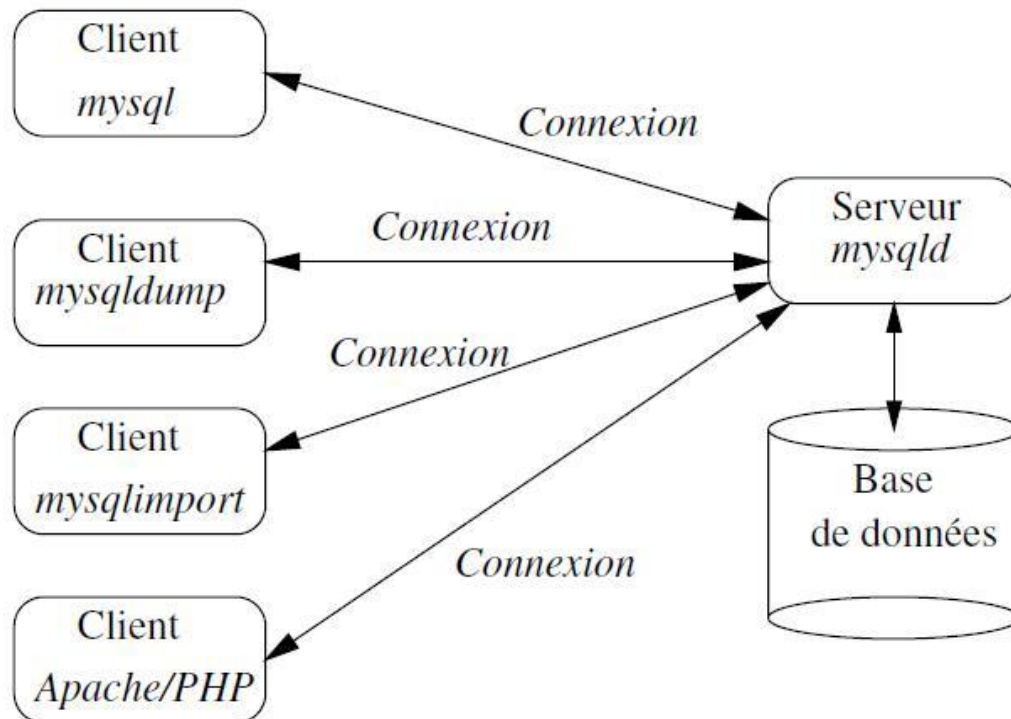


Figure 10 : Architecture client/serveur de MySQL

4. phpMyAdmin

PhpMyAdmin est un outil entièrement écrit en PHP qui fournit un interface simple et très complet pour administrer une base MySQL. La plupart des commandes de l'utilitaire mysql peuvent s'effectuer par l'intermédiaire de phpMyAdmin, les opérations possibles dépendant bien sûr des droits de l'utilisateur qui se connecte à la base.

Voici une liste des principales possibilités :

1. Créer et détruire des bases de données (sous le compte root de MySQL).
2. Créer, détruire, modifier la description des tables.
3. Consulter le contenu des tables, modifier certaines lignes ou les détruire, etc.
4. Exécuter des requêtes SQL interactivement.
5. Charger des fichiers dans des tables et, réciproquement, récupérer le contenu de tables dans des fichiers ASCII.
6. Administrer MySQL.

a) création d'une table

L'accueil de phpMyAdmin ressemble à ceci :

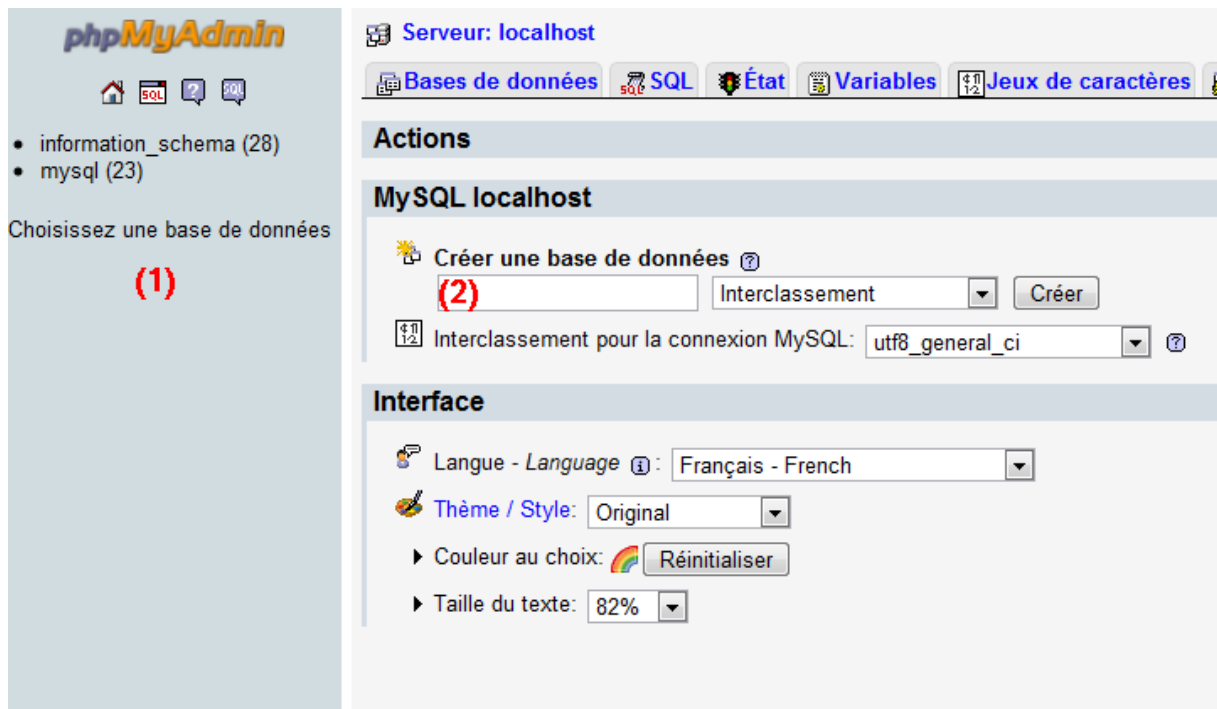


Figure 11 : Page d'accueil de phpMyAdmin

Nous avons 2 endroits importants, signalés par des numéros sur la capture d'écran :

1. Liste des bases : c'est la liste de nos bases de données. Le nombre entre parenthèses, c'est le nombre de tables qu'il y a dans la base. Sur ma capture d'écran, on a donc 2 bases : `information_schema`, qui contient 28 tables, et `mysql`, qui contient 23 tables.
2. Créer une base : pour créer une nouvelle base de données, tapons un nom dans le champ de formulaire à droite, et cliquons sur "Créer".

Nous allons créer une nouvelle base "bibliotheque" dans laquelle nous travaillerons dans notre stage. On utilise le formulaire à droite pour créer cette base.

L'écran suivant devrait alors s'afficher si la base a bien été créée :

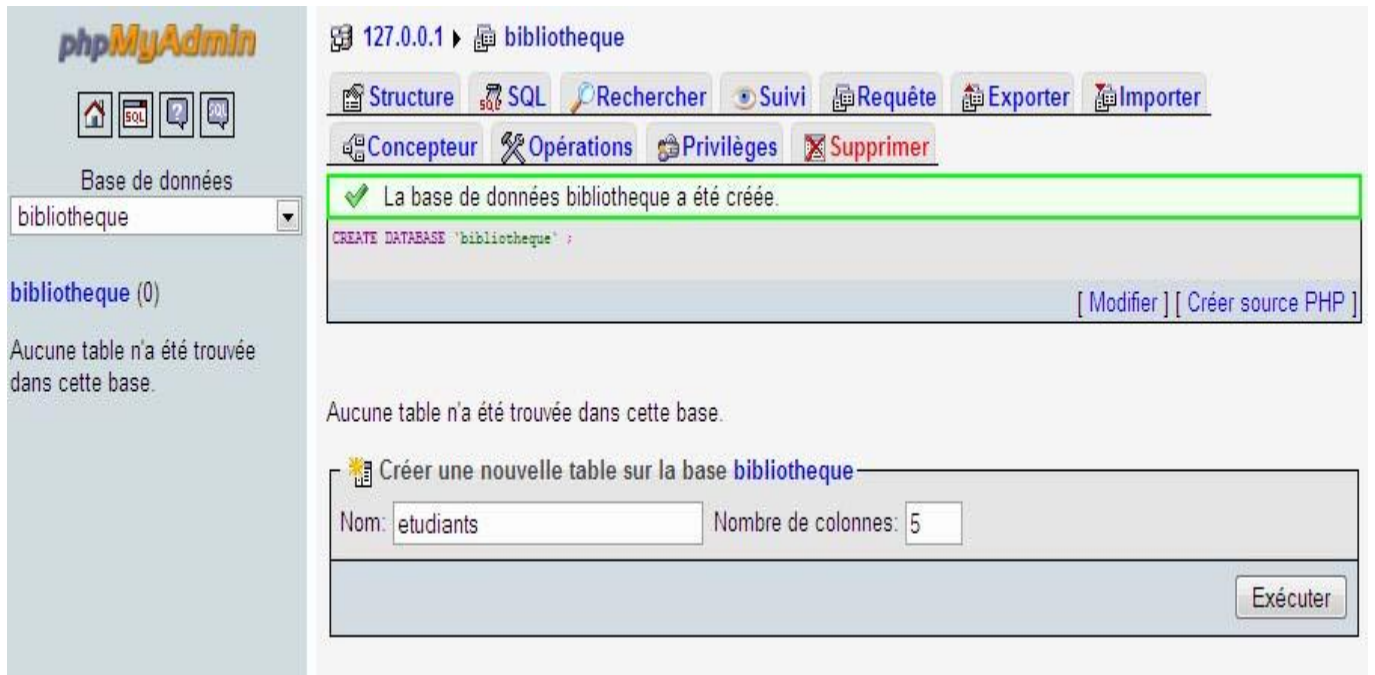


Figure 12 : Création d'une table

Dans le champ "Créer une nouvelle table sur la base bibliotheque", indiquons le nom etudiant et le nombre de champs 3 par exemple et Cliquons sur "Exécuter".

La table n'est pas immédiatement créée, il faut maintenant indiquer le nom des champs et les données qu'ils peuvent contenir.

Donc on va créer les 3 champs suivants pour cette table :

- CNE : c'est le code national d'étudiant
- nom : ce champ contiendra le nom d'étudiant.
- prenom : enfin, ce champ contiendra le prénom d'étudiant.

127.0.0.1 ▶ bibliotheque ▶ etudiants

Colonne	CNE	nom	prenom
Type	INT	VARCHAR	VARCHAR
Taille/Valeurs ¹	12	100	100
Défaut ²	Aucun	Aucun	Aucun
Interclassement			
Attributs			
Null	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index	PRIMARY	---	---
AUTO INCREMENT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Commentaires			
Type MIME			
Transformation			
Options de transformation ³			

Commentaires sur la table:

Moteur de stockage: MyISAM

Interclassement:

Définition de PARTITION:

Sauvegarder Ou Ajouter 1 colonne(s) Exécuter

Figure 13 : création des champs

Chaque colonne représente un champ. Nous avons demandé 3 champs, il y a donc 3 colonnes.

phpMyAdmin nous demande beaucoup d'informations, il n'est pas nécessaire de tout remplir. La plupart du temps, les sections les plus intéressantes seront :

- Champ : permet de définir le nom du champ (important)
- Type : le type de données que va stocker le champ (nombre entier, texte, date...)

- Taille/Valeurs : permet d'indiquer la taille maximale du champ, utile pour le type VARCHAR notamment afin de limiter le nombre de caractères autorisés.
- Index : active l'indexation du champ. Ce mot signifie notre champ sera adapté aux recherches. Le plus souvent, on utilise l'index PRIMARY sur le champ CNE.
- AUTO_INCREMENT : permet au champ de s'incrémenter tout seul à chaque nouvelle entrée.

b) Les types de champs MySQL

Si on déroule la liste des types que nous propose MySQL, nous devrions tomber à la renverse :

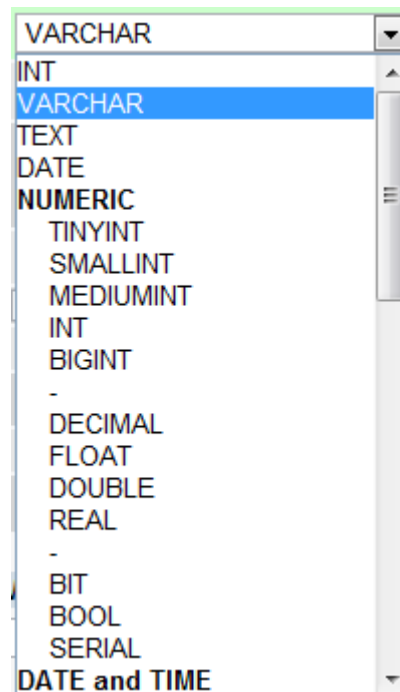


Figure 14 : Les types de données en MySQL

MySQL propose une quantité très importante de types de données.

En fait, ceux-ci sont classés par catégories :

- NUMERIC : ce sont les nombres. On y trouve des types dédiés aux petits nombres entiers (TINYINT), aux gros nombres entiers (BIGINT), aux nombres décimaux, etc.
- DATE and TIME : ce sont les dates et les heures. De nombreux types différents permettent de stocker une date, une heure, ou les deux à la fois.

- **STRING** : ce sont les chaînes de caractères. Là encore, il y a des types adaptés à toutes les tailles.

En fait, phpMyAdmin a eu la bonne idée de proposer au tout début de cette liste les 4 types de données les plus courants :

- **INT** : nombre entier.
- **VARCHAR** : court texte (entre 1 et 255 caractères).
- **TEXT** : long texte (on peut y stocker un roman sans problème).
- **DATE** : date (jour, mois, année).

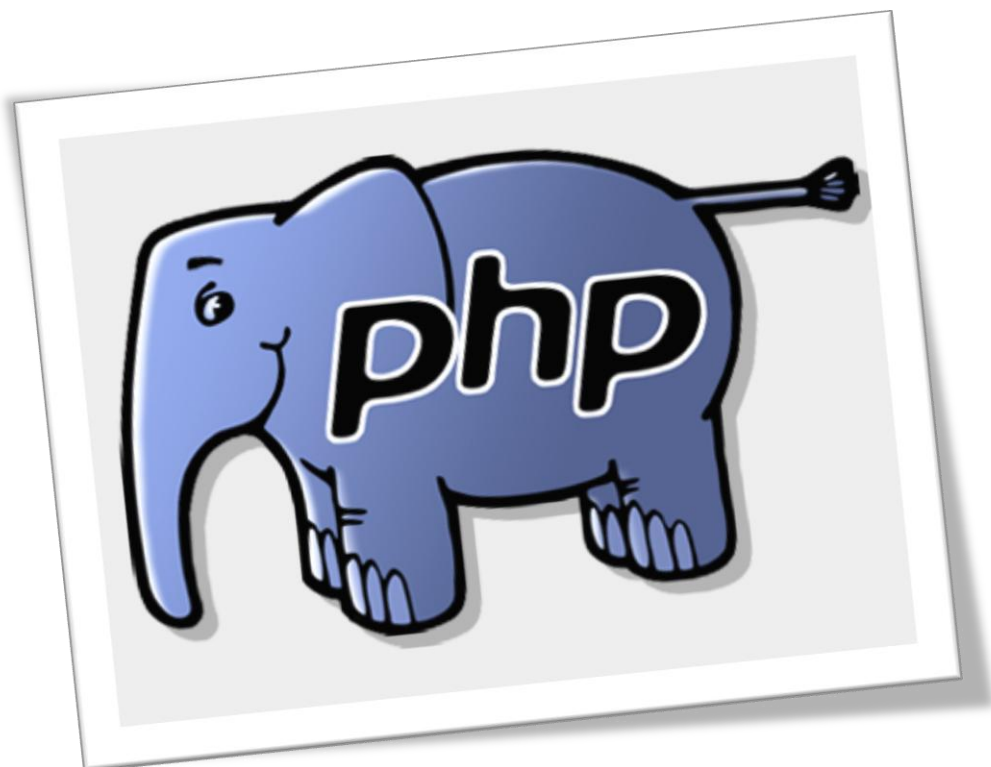
c) Les clés primaires

Toute table doit posséder un champ qui joue le rôle de clé primaire. La clé primaire permet d'identifier de manière unique une entrée dans la table.

Par exemple chaque étudiant de notre application doit pouvoir être identifiée de manière unique. Le moyen le plus simple pour cela est de lui donner un numéro unique, dans un champ nommé "CNE". Il ne peut pas y avoir 2 étudiants avec le même CNE ! Il en irait de même pour les autres tables de notre application : par exemple, chaque ouvrage doit se voir attribuer un numéro unique (code d'ouvrage par exemple).

Chapitre 3

Langage PHP



I. Généralités sur les langages informatiques

1. les langages informatiques et le système d'information

Les langages informatiques interviennent à différents moments dans le cycle de vie d'un système d'information (SI).

Certains sont utilisés lors des phases de spécification. UML et les formalismes graphiques définis dans Merise font partie de ces langages. Ils permettent de définir des modèles de données et de traitements. Même si ces langages sont principalement destinés aux humains et non aux machines, on les classe parmi les langages informatiques car ils font partie du processus de développement des systèmes. De plus, certains logiciels permettent de produire du code source, exprimé dans un langage de programmation, à partir de langages de conception. Des méthodes formelles (langages mathématiques) et des langages algorithmiques sont également utilisés pour concevoir des traitements.

La phase de mise en œuvre suit la phase de conception. Durant cette étape, sont principalement utilisés :

- des langages de programmation qui implantent physiquement les modèles de données et de traitements, les méthodes formelles, et algorithmes. Ils sont appelés langages de programmation généralistes ;
- des langages de définition de données qui rendent concrets les modèles conceptuels de données ;
- des langages de requête, comme SQL, qui permettent de réaliser les traitements sur les données définies.

2. Langage interprété

Un langage informatique est par définition différent du langage machine. Il faut donc le traduire pour le rendre intelligible du point de vue du processeur. Un programme écrit dans un langage interprété a besoin d'un programme auxiliaire (l'interpréteur) pour traduire au fur et à mesure les instructions du programme.

3. Langage compilé

Un programme écrit dans un langage dit « compilé » va être traduit une fois pour toutes par un programme annexe (le compilateur) afin de générer un nouveau fichier qui sera autonome, c'est-à-dire qui

n'aura plus besoin d'un programme autre que lui pour s'exécuter (on dit d'ailleurs que ce fichier est exécutable).

Un programme écrit dans un langage compilé a comme avantage de ne plus avoir besoin, une fois compilé, de programme annexe pour s'exécuter. De plus, la traduction étant faite une fois pour toute, il est plus rapide à l'exécution. Toutefois il est moins souple que programme écrit avec un langage interprété car à chaque modification du fichier source (fichier intelligible par l'homme: celui qui va être compilé) il faudra recompiler le programme pour que les modifications prennent effet.

D'autre part, un programme compilé a pour avantage de garantir la sécurité du code source. En effet, un langage interprété, étant directement intelligible (lisible), permet à n'importe qui de connaître les secrets de fabrication d'un programme et donc de copier le code voire de le modifier. Il y a donc risque de non-respect des droits d'auteur. D'autre part, certaines applications sécurisées nécessitent la confidentialité du code pour éviter le piratage (transaction bancaire, paiement en ligne, communications sécurisées, ...).

4. Langages intermédiaires

Certains langages appartiennent en quelque sorte aux deux catégories (LISP, Java, Python, ..) car le programme écrit avec ces langages peut dans certaines conditions subir une phase de compilation intermédiaire vers un fichier écrit dans un langage qui n'est pas intelligible (donc différent du fichier source) et non exécutable (nécessité d'un interpréteur). Les applets Java, petits programmes insérés parfois dans les pages Web, sont des fichiers qui sont compilés mais que l'on ne peut exécuter qu'à partir d'un navigateur internet (ce sont des fichiers dont l'extension est « .class »).

II. Des langages particulier : les langages du web

1. Les sites statiques et dynamiques

On considère qu'il existe 2 types de sites web : les sites **statiques** et les sites **dynamiques**.

- **Les sites statiques** : ce sont des sites réalisés uniquement à l'aide des langages (X)HTML et CSS. Ils fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement : il faut que le propriétaire du site (le webmaster) modifie le code source pour y ajouter des nouveautés. Ce n'est pas très pratique quand on doit mettre à jour un site plusieurs fois dans la même journée ! Les sites statiques sont donc bien adaptés pour réaliser des sites "vitrine", pour présenter par exemple une entreprise, mais sans aller plus loin.
- **Les sites dynamiques** : plus complexes, ils utilisent d'autres langages en plus de (X)HTML et CSS, tels que PHP et MySQL. Le contenu de ces sites web est dit "dynamique" parce qu'il peut changer sans l'intervention du webmaster !

Leur utilité se fait sentir quand on doit écrire des pages qui doivent s'adapter (besoin de l'heure, accès à des bases de données, réponse personnalisée, traitement de formulaire, etc.), mais aussi pour faciliter la gestion d'un site (de grande taille par exemple), séparation du contenu (la structure) et du traitement (la présentation).

2. Comment fonctionne un site web ?

L'internet est un réseau composé d'ordinateurs. Ceux-ci peuvent être classés en deux catégories :

- Les clients : ce sont les ordinateurs des internautes.



Figure 15 : symbole d'un client

- Les serveurs : ce sont des ordinateurs puissants qui stockent et délivrent des sites web aux internautes.

La plupart du temps, le serveur est dépourvu d'écran : il reste allumé et travaille tout seul sans intervention humaine, 24h/24, 7j/7.



Figure 16 : symbole d'un serveur

a) Cas d'un site statique

Lorsque le site est statique, le schéma est très simple. Cela se passe en deux temps :

1. Le client demande au serveur à voir une page web.
2. Le serveur lui répond en lui envoyant la page réclamée.

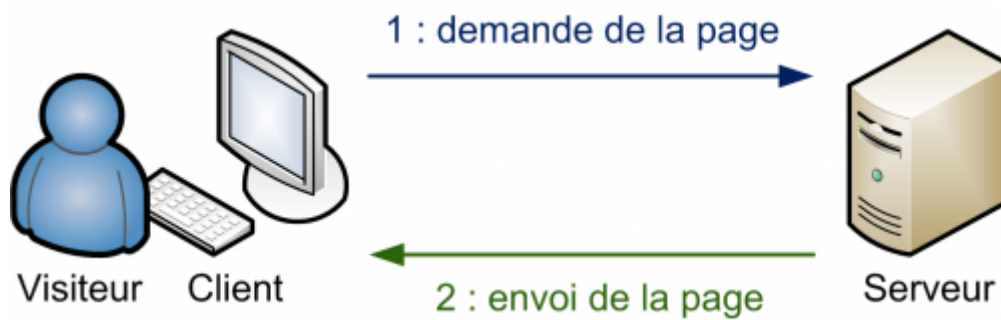


Figure 17 : Fonctionnement d'un site web statique

b) Cas d'un site dynamique

Lorsque le site est dynamique, il y a une étape intermédiaire : la page est générée.

- Le client demande au serveur à voir une page web.
- Le serveur prépare la page spécialement pour le client.
- Le serveur lui envoie la page qu'il vient de générer.



Figure 18 : Fonctionnement d'un site web dynamique

3. Les langages du web

a) Pour un site statique :

- De nombreux langages ont été créés pour produire des sites web. Deux d'entre eux constituent une base incontournable pour tous les webmasters :
- **XHTML** : c'est le langage à la base des sites web. Il ressemble beaucoup au HTML mais impose quelques règles un peu plus strictes. XHTML est un langage simple à apprendre qui fonctionne à partir de balises. Voici un exemple de code XHTML :

Code : XHTML

```
1 <p>Bonjour, je suis un <em>paragraphe</em> de texte !</p>
```

- **CSS** : c'est le langage de mise en forme des sites web. Tandis que le XHTML permet d'écrire le contenu de nos pages web et de les structurer, le langage CSS s'occupe de la mise en forme et de la mise en page. C'est en CSS que l'on choisit notamment la couleur, la taille des menus et bien d'autres choses encore. Voici un code CSS :

Code : CSS

```
1 div.banner {
2   text-align: center;
3   font-weight: bold;
4   font-size: 120%;
5 }
```

b) Pour un site dynamique :

Quel que soit le site web que l'on souhaite créer, XHTML et CSS sont donc indispensables. Cependant, ils ne suffisent pas pour réaliser des sites dynamiques. Il faut les compléter avec d'autres langages.

- PHP : c'est un langage que seuls les serveurs comprennent et qui permet de rendre notre site dynamique. C'est PHP qui "génère" la page web comme on l'a vu sur un des schémas précédents. Il peut fonctionner seul, mais il ne prend vraiment de l'intérêt que s'il est combiné à un outil tel que MySQL. Voici un code PHP

Code : PHP

```
1 <?php
2 $nbre_visiteurs=20 ;
3 echo "Vous êtes le visiteur n°" . $nbre_visiteurs;
4 ?>
```

- SQL : c'est un langage qui permet de communiquer avec la base de données MySQL. Voici un code en langage SQL :

Code : SQL

```
1 SELECT nom,prenom,CNE FROM etudiants ORDER BY CNE DESC LIMIT 0,10
```

4. PHP génère du XHTML

Les clients sont incapables de comprendre le code PHP, ils ne connaissent que le XHTML et le CSS. Seul le serveur est capable de lire du PHP.

Le rôle de PHP est justement de générer du code XHTML (on peut aussi générer du CSS, mais c'est plus rare), code qui est ensuite envoyé au client de la même manière qu'un site statique :

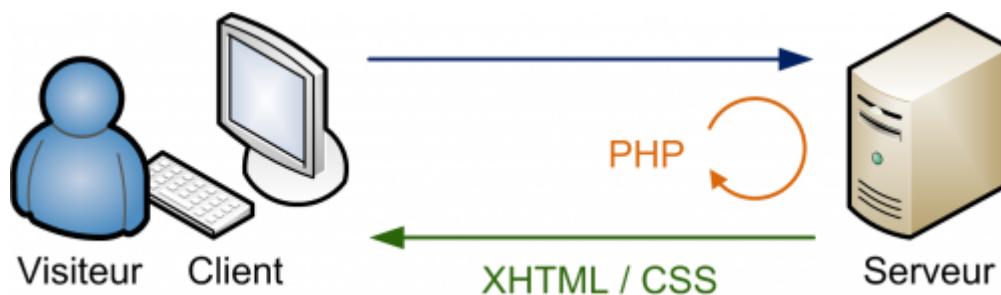






Figure 19 : PHP décide ce qui va être affiché sur la page web envoyée au visiteur

PHP est un langage de programmation utilisé sur de nombreux serveurs pour prendre des décisions. C'est PHP qui décide du code XHTML qui sera généré et envoyé au client à chaque fois.

III. Les bases du langage PHP

Ces notes suppose une connaissance de base de la syntaxe du langage C.

1. Quelques particularités syntaxiques

-  Diverses notations pour commentaires : //, /* ...*/, #
-  PHP distingue la casse
-  Les variables simples ont un identificateur commençant par \$
-  Les variables ne font pas l'objet d'une déclaration préalable de type. Cette apparente facilité est une source commune d'erreurs syntaxiques (l'interpréteur ne remarquera pas une petite faute !)

2. Structures de données

a) Types

- On distingue en PHP les types scalaires entier (sur 4 octets, par exemple 2), double (2.0) et chaîne ("2") et les types structurés tableau (liste ou associatif) et classe
- Pas de type booléen
Comme en C une valeur numérique zéro 0 ou 0.0 signifie FAUX, de même que "0" et la chaîne vide "".
Toute autre valeur est considérée comme vraie
Malgré l'absence du type, les constantes usuelles TRUE et FALSE existent.

Code : PHP

```
1 $trouve=0 /* trouve est FALSE */
2 while (!$trouve) {
3     /* au début !$trouve = 1 = TRUE */
4     if (...)
5     trouve = 1 ;
6     /* trouve devient vraie */
7 }
```

b) Constantes et variable

On peut définir des constantes, de valeur invariable, par la fonction *define*. Par exemple :

Code : PHP

```
1 define(PI, 3.14159);
2 if (defined("PI")) echo "La valeur approchée de PI est ", PI
```

Les variables ont des noms précédés du symbole distinctif \$ Elles ne nécessitent pas de déclaration avant usage. Leur type est fixé dès leur première affectation. Par exemple :

Code : PHP

```
1 $a = 3 ; $b=7;
2 $c= $a + $b ;
3 echo " \$a + \$b = $a + $b <BR>" ;
4 echo " \$a + \$b = $c" ;
```

Exécution

```
$a + $b = 10
$a + $b = 10
```

c) Tableaux

- Les tableaux sont de taille variable, et les éléments peuvent être de type différents

Les valeurs sont des ensembles ordonnés et indicés , ou bien associées à une clé (tableau associatif)

L'initialisation est permise avec le constructeur array, ou en donnant la suite des éléments

- Nombre d'éléments

La fonction *count*(\$tab) donne le nombre d'éléments actuels du tableau \$tab

Exemple :

Code : PHP

```
1 <?php
2 $tab = array (10, 15, "trois");
3 echo "\$tab[1] vaut $tab[1] <BR>";
4 $villes[0] = "Fès";
  $villes[]="Rabat";
  $villes[] = "casa";
  echo "Je me déplace de $villes[0] à $villes[2] <BR>" ;
  ?>
```

Exécution

```
$table[1] vaut 15
```

```
Je me déplace de Fès à casa
```

- Parcours d'un tableau indicé

La construction suivante est standard dans le cas usuel d'un tableau dont le premier élément porte l'indice 0

Code : PHP

1	\$nb= count(\$tab);
2	for (\$i=0; \$i < \$nb ; \$i++)
3	echo \$tab[\$i] . " " ;

3. Expressions et opérateurs

- Une expression est une construction du langage utilisant des variables, des littéraux et des opérateurs et qui est évaluable.
Toute expression peut être interprétée, suivant le contexte, comme une expression booléenne.
- Les opérateurs arithmétiques, d'affectation et d'incrémentation du C sont utilisables

```
$facto=1 ;
for ( $i=1 ; $i <= 10; $i++ )
    $facto *= $i ;
```

- La concaténation de chaînes est notée ‘.’

```
$ch = "Bonjour " ;
$ch .= "tout le monde !" ;
```

- Les opérateurs de comparaison et logiques sont usuels. A noter que && et || peuvent être aussi notés and, or, et que xor est le ou exclusif

4. Structures de contrôle

a) Conditionnelles

Structure if

- Syntaxes

```
if (condition) { bloc1; }
else { bloc2; }
if (condition1) { bloc1; }
elseif (condition2) { bloc2; }
elseif (condition3) { bloc3; }
```

```
else { bloc4;}
```

- Les accolades sont facultatives si le bloc est réduit à une seule instruction.
- Exemple

Code : PHP

```
1 <?php
2 $a = 10; $b = 11;
3 echo "\$a = $a, \$b = $b ";
4 if ($a > $b)
5     echo "\$a est supérieur à \$b";
6 elseif ($a == $b)
7     echo "\$a est égal à \$b";
8 else
9     echo "\$a est inférieur à \$b.";
10 ?>
```

Exécution

```
$a = 10, $b = 11
```

```
$a est inférieur à $b
```

Structure switch

Voici un exemple d'utilisation significatif,

Code : PHP

```
1 <?php
2 $c=2;
3 switch($c)
4 {
5     case 1 : $nombre= 'un' ; break ;
6     case 2 : $nombre= 'deux' ; break ;
7 }
8 echo $nombre ;
9 ?>
```

Exécution

```
deux
```

b) boucles

```
while (condition) {
// bloc d'instructions
}
do {
// bloc d'instructions
} while (condition);
```

```
for($i=0; $i <=$max ; $i++) {
// bloc d'instructions
}
foreach ($tab as $el) {
// instructions portant sur $el, élément du tableau
}
```

```

Construction
while (list(var1, $var2 ..) = each($liste)) {
// bloc d'instructions
}

```

c) les fonctions

- Syntaxe

```

function nomfonction (liste-paramètres) {
....
[return $var]
}

```

Ici, le nom de la fonction n'est pas sensible à la casse.

Il peut n'y avoir qu'une valeur de retour après **return**, qui peut être un tableau.

- Exemple

Code : PHP

```

1 <?
2 fonction somme ($a = 1, $b = 2) {
3 return $a + $b;
4 }
5 echo "somme(3, 5) = ", somme (3,5), "<BR>" ;
6 echo "somme(3) = ", somme (3), "<BR>" ;
7 echo "somme() = ", somme (), "<BR>" ;
8 ?>
9

```

Exécution

```

somme(3, 5) = 8

somme(3) = 5
somme() = 3

```

5. Accès à MySQL avec PHP

a) Connexion MySQL

La connexion MySQL via PHP se fait en 2 étapes. La première étape est la connexion au serveur MySQL avec un identifiant et un mot de passe, et la seconde est le choix d'une base de données de travail. Un exemple de connexion est montré par le script *connexion.php* ci-dessous :

Code : PHP

```

1 <?php
2 // On définit les 4 variables nécessaires à la connexion MySQL :
3 $hostname = "localhost";
4 $user = "root";
5 $password = "password_mysql";
6 $nom_base_donnees = "biblio";
7
8 // Connexion au serveur MySQL
9 $bdd = mysql_connect($hostname, $user, $password) or die(mysql_error());

```

```
10 // Choix de la base sur laquelle travailler
11 mysql_select_db($nom_base_donnees, $conn);
12 ?>
13
```

En faisant un `include('connexion.php')`; dans nos pages PHP, nous aurons ainsi la connexion à MySQL.

b) Lire des données dans une table

Avant de lire des données dans une table, il faut que votre table contienne des enregistrements.

Qu'il s'agisse d'une lecture, d'une écriture ou d'une modification de données ou de paramètres, il nous faudra passer par la fonction `mysql_query()`. Query signifie requête en anglais. C'est cette fonction qui va nous permettre d'interagir avec MySQL.

Code : PHP

```
1 <?php
2 $requete = mysql_query('Ici la requête SQL');
3 ?>
```

Le résultat de la requête sera retourné dans la variable `$requete`.

En SQL, lorsque nous souhaitons sélectionner des données provenant d'une table, on utilise tout d'abord le mot `SELECT`, qui veut dire que nous nous apprêtons à récupérer des données. Ensuite, nous devons indiquer à MySQL la liste des champs de la table que nous souhaitons voir dans notre résultat, séparés par des virgules. Nous devons ensuite indiquer à quelle table nous souhaitons prendre les données via le mot clé `FROM` suivi du nom de la table

Si nous souhaitons récupérer les champs `nom` et `prenom` de la table `etudiant`, voici comment nous allons procéder :

Code : PHP

```
1 <?php
2 $requete = mysql_query('SELECT nom, prenom FROM etudiant');
3 ?>
```

Et pour récupérer le résultat d'une requête sous forme textuelle on utilise trois fonctions de PHP (en fait on en utilise une parmi les quatre). Voilà les trois fonctions que nous pouvons utiliser :

- `mysql_fetch_row()` : cette fonction va retourner un tableau avec des indices numériques (l'indice 0 correspond au premier champ de la requête, l'indice 1 au second champ ...).
- `mysql_fetch_array()` : cette fonction va retourner un tableau avec par défaut les indices numériques et associatifs. Elle est plus lente que `mysql_fetch_row()` mais plus simple d'utilisation. Pour la rendre plus rapide, on peut utiliser la fonction `mysql_fetch_assoc()` qui va uniquement retourner un tableau avec les indices associatifs.
- `mysql_fetch_object()` : cette fonction retourne un objet qui aura pour attributs les champs de la requête MySQL.

Voyons voir ce que ça donne pour sélectionner tous les enregistrements d'une table, et les afficher :

Code : PHP

```

1 <?php
2     include('connexion.php') ;
3
4     $requete = mysql_query('SELECT nom, prenom
5     FROM etudiant) OR die('Erreur de la requête MySQL');
6
7     mysql_close();
8
9     /**
10    * On récupère les données
11    * Tant qu'une ligne sera présente, la boucle continuera
12    */
13
14    while($resultat = mysql_fetch_object($requete))
15    {
16        echo '<p>Nom : '.$resultat->nom.'. prénom : '.$resultat->prenom.'</p>';
17    }
18
19 ?>
20

```

Avec la fonction **`mysql_fetch_row()`** :

Code : PHP

```

1 <?php
2     include('connexion.php') ;
3
4     $requete = mysql_query ('SELECT nom, prenom
5     FROM etudiant) OR die('Erreur de la requête MySQL');
6
7     mysql_close();
8
9     /**
10    * On récupère les données
11    * Tant qu'une ligne sera présente, la boucle continuera

```

```

10 */
11
12 while($resultat = mysql_fetch_row($requete))
13 {
14     echo '<p>Nom : '. $resultat[0].'. prénom : '. $resultat[1].</p>';
15 }
16 ?>

```

Et avec la fonction **mysql_fetch_assoc()** :

Code : PHP

```

1 <?php
2 include('connexion.php') ;
3
4 $requete = mysql_query ('SELECT nom, prenom
5 FROM etudiant') OR die('Erreur de la requête MySQL');
6
7 mysql_close();
8
9 /**
10 * On récupère les données
11 * Tant qu'une ligne sera présente, la boucle continuera
12 */
13
14 while($resultat = mysql_fetch_assoc($requete))
15 {
16     echo '<p>Nom : '. $resultat['nom'].'. prénom : '. $resultat['prenom'].</p>';
17 }
18 ?>
19

```

c) Insérer des données

Pour tout ce qui est insertion, suppression, modification de données, le principe est le même que la lecture de données, sauf que là on ne récupère aucun résultat.

Exemple d'insertion de données :

Code : PHP

```

1 <?php
2 include('connexion.php') ;
3
4 $requete = mysql_query ("INSERT INTO ouvrage(code, titre, auteur, specialite)
5 VALUES('4942', 'l'enfant', 'VALLES JULES', 'etudes_fr') "
6 ) OR die('Erreur de la requête MySQL');
7
8 mysql_close();
9
10 ?>

```

Chapitre 4

Application



Introduction

Dans ce chapitre on va présenter le fruit de notre stage durant les deux mois à savoir une application de gestion de la bibliothèque en PHP/MySQL



Figure 20 : page d'accueil de l'application

On voit clairement que la figure ci-dessus représente la page d'accueil de notre projet accompagné par des boutons (Page d'Accueil, Espace Etudiants, Espace bibliothécaire, Espace Administrateur).

- La page d'accueil contient un message d'accueil.
- L'espace étudiant contient deux pages l'un pour la recherche des ouvrages par étudiante et l'autre pour l'appréciation et la demande des ouvrages.
- L'espace bibliothécaire contient aussi deux pages l'un pour l'empreint et l'autre pour la restitution des ouvrages.
- L'espace administrateur est pour gérer les étudiants et les ouvrages telle que l'insertion, la suppression et la modification.

I. Espace étudiant

L'espace étudiant est divisé en deux parties : recherche et appréciation comme le montre la figure suivante :



Figure 21 : Espace étudiant

Dans l'espace étudiant la page par défaut est la page de recherche

Par exemple si on veut chercher un ouvrage on saisie la phrase de recherche et on sélectionne le type de recherche dans la liste déroulante puis on clique sur le bouton Recherche

Quant à la page appréciation ou demande est réservée aux enregistrements des appréciations des étudiants ainsi que les demandes de certains ouvrages selon les besoins des étudiants.

Cette page là est représentée par la figure suivante :

Recherche d'ouvrage

127.0.0.1:8888/fin/etudiant.php?id=2

UNIVERSITE SIDI MOHAMED BEN ABDELLAH
Faculté des Lettres et de Sciences Humaines SAIS FES

Page d'Accueil **Espace Etudiants** Espace bibliothécaire Admin Help

Espace Etudiant

- Recherche
- appréciation**

APPRÉCIATION

Titre

Auteur

Spécialité

Appréciation

Envoyer Rétablir

Figure 22 : Page d'Appréciation

II. Espace Administrateur



Figure 23 : connexion à l'espace administrateur

Donc l'espace administrateur est protégé par un mot de passe, et pour accéder à cet espace il faut saisir un Pseudo et un mot de passe et on clique sur le bouton «Se connecter» on obtient la figure :

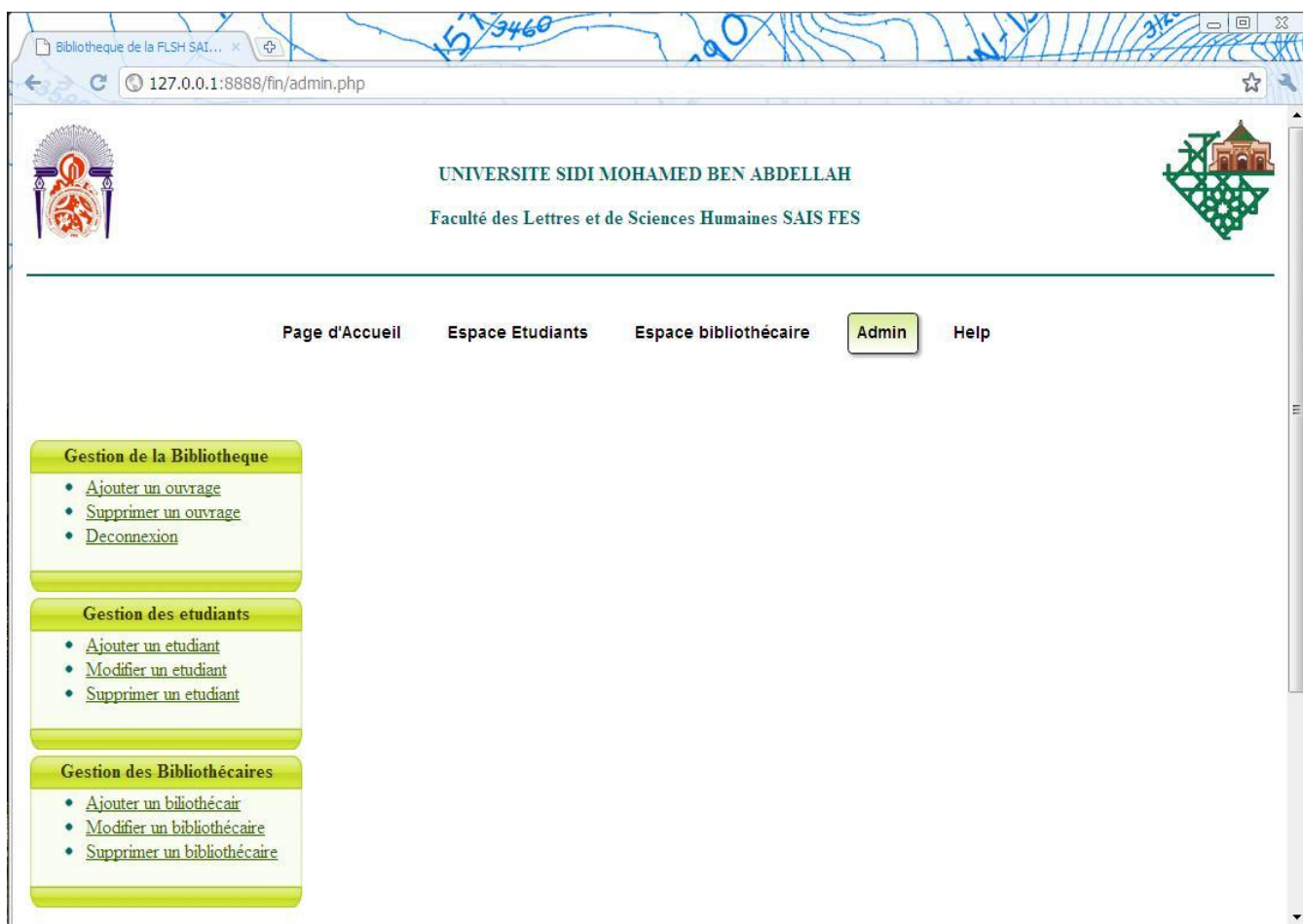


Figure 24 : Espace administrateur

1. L'ajout d'un ouvrage

Bibliothèque de la FLSH SAI...
127.0.0.1:8888/fin/admin.php?id=1

UNIVERSITE SIDI MOHAMED BEN ABDELLAH
Faculté des Lettres et de Sciences Humaines SAIS FES

Page d'Accueil Espace Etudiants Espace bibliothécaire **Admin** Help

INSERTION D'UN OUVRAGE

Code

Titre

Auteur

Edition

Spécialité

Quantité

Gestion de la Bibliothèque

- [Ajouter un ouvrage](#)
- [Supprimer un ouvrage](#)
- [Deconnexion](#)

Gestion des étudiants

- [Ajouter un étudiant](#)
- [Modifier un étudiant](#)
- [Supprimer un étudiant](#)

Gestion des Bibliothécaires

- [Ajouter un bibliothécaire](#)
- [Modifier un bibliothécaire](#)
- [Supprimer un bibliothécaire](#)

Figure 25 : Insertion d'un ouvrage

2. Suppression d'un ouvrage

Pour supprimer un ouvrage il suffit de saisir le code d'ouvrage puis on clique sur le bouton « Envoyer » :



Figure 26 : suppression d'un ouvrage

En ce qui concerne l'insertion, la suppression des étudiants et des bibliothécaires, elles se font de la même manière des ouvrages.

III. Espace bibliothécaire

1. Empreint d'ouvrage par étudiants



Figure 27 : Page d’empreint d’un ouvrage

L’empreint d’ouvrages par étudiants s’effectue si et seulement si les deux contraintes suivantes sont vérifiées :

- La carte d’étudiant n’est pas bloquée.
- L’étudiant a restitué l’ouvrage avant l’empreint d’un autre

Si l’une des contraintes n’est pas vérifiée alors le programme affiche un message d’erreur sinon elle donne la page suivante pour saisir le code d’ouvrage :

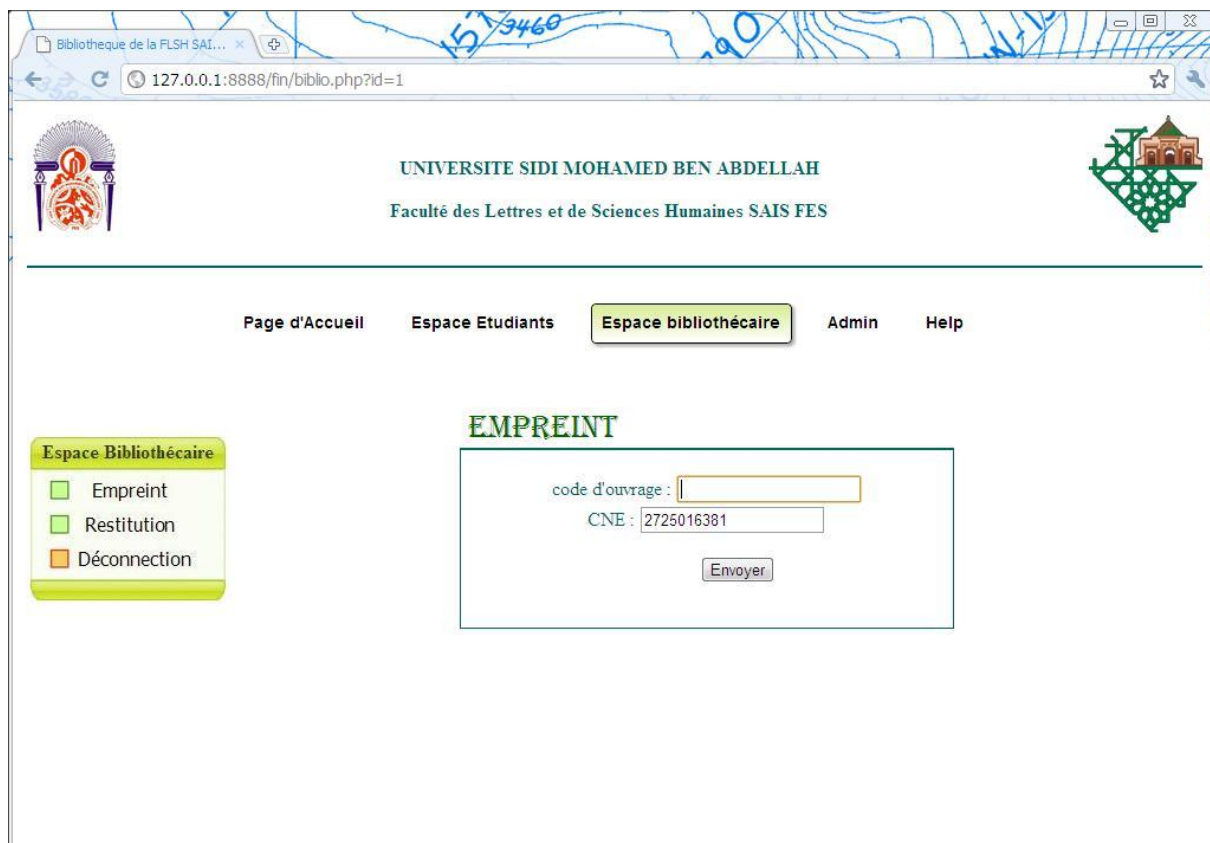
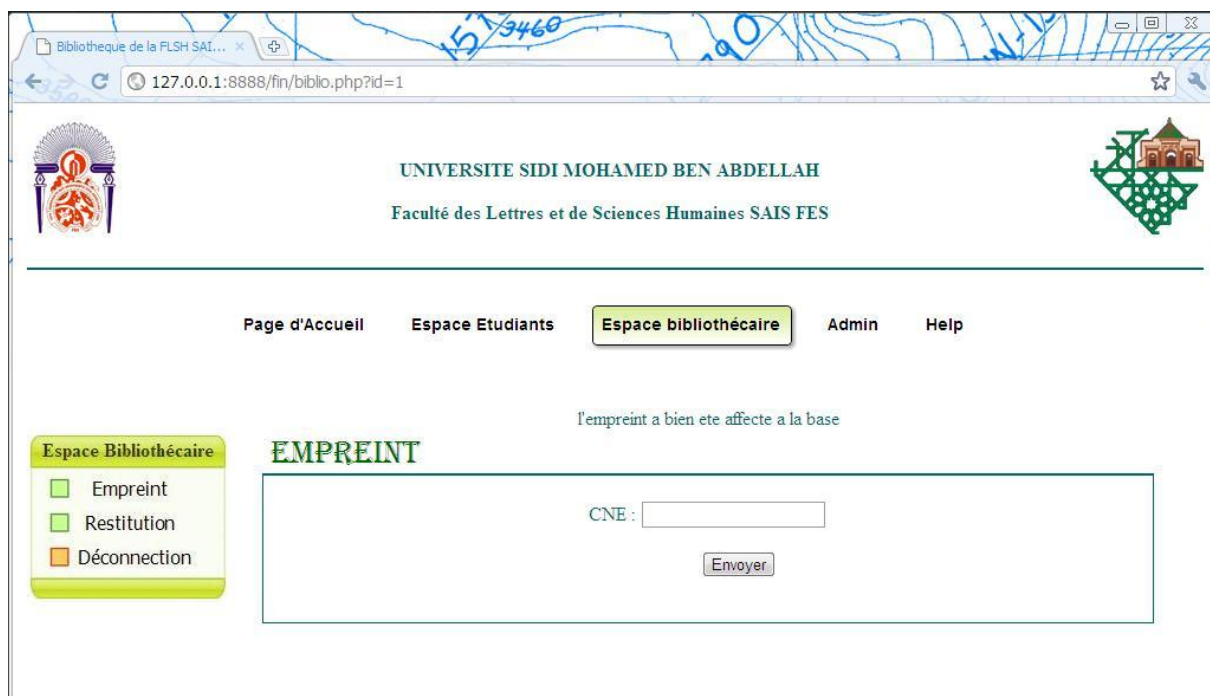


Figure 28 : L'empreint d'un ouvrage

On clique sur envoyer pour affecter l'empreint dans base et diminuer le nombre d'ouvrage existant dans la bibliothèque.



2. Restitution d'ouvrage

En ce qui concerne la restitution d'un ouvrage par étudiant, il suffit de saisir le CNE d'étudiant concerné puis cliquer sur le bouton Envoyer. Le rôle de ce dernier est d'augmenter le nombre d'ouvrages existants et de supprimer l'enregistrement pour donner à l'étudiant la permission d'emprunter un autre ouvrage.



The screenshot shows a web browser window with the URL `127.0.0.1:8888/fin/biblio.php?id=2`. The page header includes the logo of Université Sidi Mohamed Ben Abdellah and the text "UNIVERSITE SIDI MOHAMED BEN ABDELLAH" and "Faculté des Lettres et de Sciences Humaines SAIS FES". The navigation menu contains "Page d'Accueil", "Espace Etudiants", "Espace bibliothécaire" (highlighted), "Admin", and "Help". On the left, a sidebar menu for "Espace Bibliothécaire" includes "Empreint", "Restitution" (highlighted), and "Déconnection". The main content area is titled "RESTITUTION" and features a form with a "CNE:" label and an input field. Below the input field is an "Envoyer" button. The browser's address bar at the bottom shows the same URL.

Figure 30 : Page de restitution d'un ouvrage

Conclusion

L'objectif de notre stage a été réalisé une base de données permettant de gérer les différentes organisations de la bibliothèque de la faculté des lettres et des sciences humaines SAIS-Fès telle que la recherche , l'ajout, l'empreint, et la restitution des ouvrages afin de bien faciliter les taches aux bibliothécaires. Donc on a réalisé une application web de gestion de la bibliothèque, ce programme gère :

- Gestion d'ouvrage telle que la recherche, la suppression et l'ajout.
- L'empreint et la restitution des ouvrages par chacun des étudiants.
- Une gestion personnelle de service bibliothèque.

Pendant ce stage j'ai pu suivre et comprendre la manière dont un projet prend forme et comment il est mené a bien, un autre terme gérer les difficultés que l'on peu rencontrer. Ce stage a été aussi une opportunité pour moi d'appréhender des nouvelles notions et de développer et enrichir mes connaissance en informatiques en particulier sur la programmation (langage C, HTML, PHP, MySQL), ainsi que d'appliquer les connaissances que j'ai acquise au cours de mon cursus universitaire.

Références :

Livres :

- 1) Christain Soutou, "Apprendre SQL avec MySQL ", EDITION EYROLLES 2006
- 2) Philippe Rigaux, "Pratique de MySQL et PHP ", Dunod, Paris, 2009
- 3) Mathieu NEBRA, "Concevez Votre Site Web Avec PHP et MySQL ", Livre du zéro 2009
- 4) Damien Seguy & Philippe Gamache, "Sécurité PHP5 et MySQL", EYROLLES 2007

Sites web :

- 1) <http://fr.wikipedia.org>
- 2) <http://www.siteduzero.com>
- 3) <http://www.asp-php.net>
- 4) <http://www.commentcamarche.net>
- 5) <http://www.lephpfacile.com>