

## PHP:

1. Pour pouvoir utiliser PHP, vous devez installer Easy PHP ou un autre environnement de travail pour les scripts,

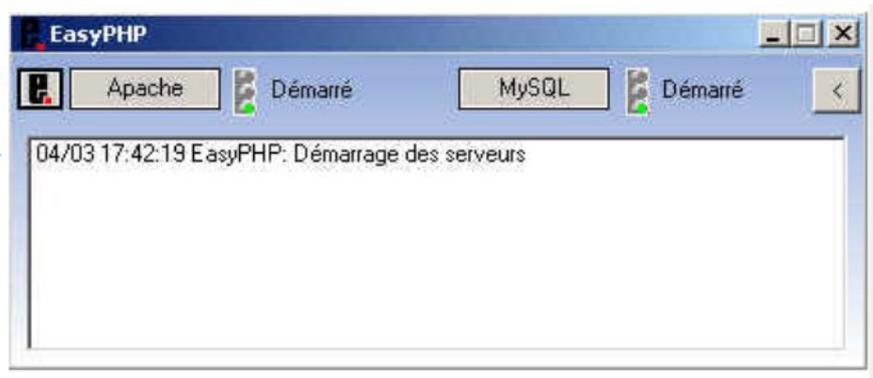
Vous pouvez trouver les fichiers d'installation sur le lien suivant:  
[https://phpsources.net/script/php/easyphp/78-20\\_easyphp,5.3.8.1](https://phpsources.net/script/php/easyphp/78-20_easyphp,5.3.8.1)

2. Suivez les étapes dans le TP PHP pour exécuter vos scripts.

## MySQL:

### 1. Le lancement

1. MySQL est un SGBD pour créer et gérer les bases de données. Une fois le Easy PHP installé, vous pouvez le lancer à partir de:



Sélectionnez: PHPMyAdmin

### 2. Création d'une nouvelle base de données



### 3. Création d'une nouvelle table

Commençons par créer une base de donnée que nous appellerons YBET par exemple (laisser le type en Interclassement par défaut, normalement, cette possibilité sélectionne Latin1\_swedish\_ci, le code de caractère par défaut des européens).

Serveur: localhost ► Base de données: ybet

Structure SQL Exporter Rechercher Requête Opérations Supprimer

**Base de données ybet a été créé(e).**

requête SQL:  
`CREATE DATABASE `ybet` ;`

[Modifier] [Créer source PHP]

Aucune table n'a été trouvée dans cette base.

✚ Créer une nouvelle table sur la base ybet:

Nom:

Champs:  Exécuter

La fenêtre suivante va permettre de personnaliser nos champs. Vous devez obligatoirement rentrer toutes les données

| Champ | Type    | Taille/Valeurs* | Interclassement | Attributs | Null     | Défaut* | Extra |  |  |  |
|-------|---------|-----------------|-----------------|-----------|----------|---------|-------|--|--|--|
|       | VARCHAR |                 |                 |           | not null |         |       |  |  |  |
|       | VARCHAR |                 |                 |           | not null |         |       |  |  |  |

- Le **nom du champ** ne peut être nul, ni comporter de caractères accentués, espaces, -, .... par contre, il accepte \_
- Le type de champ est sélectionné dans une liste de choix. Pour l'instant, sélectionnons VARCHAR qui correspond à du texte.
- la **Taille/valeurs** nous permet de déterminer la taille maximum des données à rentrer dans les champs.
- **Interclassement** représente le jeu de caractères par défaut si vous ne sélectionnez rien (préférable). Sinon vous pouvez sélectionner Latin1\_swedish\_ci. Si le jeu de caractère n'est pas démarré dans PHPMYADMIN, vous recevrez néanmoins un code d'erreur
- L'**attribut** peut-être rien, unsigned ou Unsigned zerofill. Les 2 dernières valeurs s'utilisent uniquement pour des types de champs spéciaux.
- **Null** permet de définir si le champ peut-être null (vide) ou non.
- **Défaut** permet de donner une valeur par défaut au champ.
-  permet de déterminer si le champ est ou non une **clé primaire**. Dans ce cas, la valeur du champ sera unique dans la table. Aucun n'enregistrement n'aura la même valeur dans ce champ, les tris sont également plus rapides
-  **index** permet d'augmenter les vitesses de traitement (tri) dans la table. La clé primaire est également un index.
-  **unique** permet de spécifier que le contenu est unique dans l'ensemble du champ. Contrairement à l'index, elle n'augmente pas la vitesse de traitement.
-  permettent d'indexer des textes complets. Cette possibilité ralentit la table pour de longs champs.

Pour l'instant nous allons créer 2 champs de type VARCHAR (caractères de taille variables)

- code, de taille 10 et non null. Unique
- titre, de taille 30 et non null.

N'oubliez pas de sauvegarder la table. La fenêtre suivante donne ceci

| Champ                          | Type        | Interclassement   | Attributs | Null | Défaut | Extra | Action |
|--------------------------------|-------------|-------------------|-----------|------|--------|-------|--------|
| <input type="checkbox"/> code  | varchar(10) | latin1_swedish_ci |           | Non  |        |       |        |
| <input type="checkbox"/> TITRE | varchar(30) | latin1_swedish_ci |           | Non  |        |       |        |

- permet d'éditer (modifier) la table.
- Le bouton permet de supprimer le champ.

### 3.1. Les champs de types caractères

|                   |  |  |
|-------------------|--|--|
| <b>CHAR</b>       | Chaîne de caractères fixe, nombre de caractères obligatoires. Les caractères manquants sont remplacés par des espaces. | 255 char. max., taille obligatoire         |
| <b>VARCHAR</b>    | Chaîne de caractères de longueur variable  | 255 char. max., taille maximum obligatoire |
| <b>TINYBLOB</b>   | Petite zone de texte, sensible à la casse  | 255 char. max.                             |
| <b>TINYTEXT</b>   | Petite zone de texte, insensible à la casse  |  |
| <b>BLOB</b>       | Zone de texte standard (mais accepte toute sortes de données comme des images), sensibles aux majuscules / minuscules  | 65K char. max.                             |
| <b>TEXT</b>       | Zone de texte standard (mais accepte toute sortes de données comme des images), insensible à la casse                  |  |
| <b>MEDIUMBLOB</b> | Zone de texte moyenne, différentie majuscule /minuscules   | 16 millions char. max.                     |
| <b>MEDIUMTEXT</b> | Zone de texte moyenne, insensible à la casse   | 16 millions char. max.                     |
| <b>LOBLOB</b>     | Grande zone de texte, différentie majuscule / minuscule  | 4 milliards char. max.                     |
| <b>LONGTEXT</b>   | Grande zone de texte, ne différentie pas majuscule / minuscule   | 4 milliards char. max.                     |
| <b>ENUM</b>       | Liste de choix   | 65535 valeurs max.                         |
| <b>SET</b>        | Liste de choix multiple  | 64 valeurs max.                            |

### 3.2. Les champs numériques

| Champs numériques |                             |  |
|-------------------|-----------------------------|--|
| TINYINT           | Entier très petit           | 256 valeurs différentes, de -128 à +127 ou 0 à 255 (1 octet) |
| SMALLINT          | Entier petit                | 65.536 valeurs différentes (2 octets)                        |
| MEDIUMINT         | Entier moyen                | 16.777.216 valeurs différentes (3 octets)                    |
| INT               | Entier standard             | 4.294.967.296 valeurs différentes (4 octets)                 |
| BIGINT            | Entier grand                | 8 octets   |
| FLOAT             | Décimal de simple précision | 4 octets   |
| DOUBLE, REAL      | Décimal de double précision | 8 octets   |
| DECIMAL           | Décimal sous forme de       |  |

### 3.3. Dates et heures

|                  |   |          |
|------------------|---|----------|
| <b>DATE</b>      | Date. Par exemple 2017-03-06, format anglophone   | 3 octets |
| <b>TIME</b>      | Heure. Par exemple 10:25:59   | 3 octets |
| <b>DATETIME</b>  | Date et heure. Par exemple 2017-03-06 10:25:59  | 8 octets |
| <b>TIMESTAMP</b> | Date et heure sans les séparations, par exemple 20060306102559.<br>Attention, le format a évolué avec les versions d'où un risque d'erreur en changeant d'hébergement Internet. | 4 octets |
| <b>YEAR</b>      | Année, exemple 2017   | 1 octet  |