


Relation



Kitbox

#	Nom	Type
1	Reference	varchar(11)
2	CodeBarre	varchar(11)
3	Dimensions	varchar(14)
4	Hauteur	int(11)
5	Profondeur	int(11)
6	Largeur	int(11)
7	Couleur	varchar(9)
8	Enstock	int(11)
9	StockMinimum	int(11)
10	PrixClient	varchar(6)
11	NbPiecesCasier	int(11)
12	PrixFourn1	varchar(5)
13	DelaiFourn1	int(11)
14	NbPiecesComFourn1	int(11)
15	PrixFourn2	varchar(6)
16	DelaiFourn2	int(11)
17	NbPiecesComFourn2	int(11)


Armoire

#	Nom	Type
1	idArmoire 	int(11)
2	Corniere	int(11)
3	PanneauAR	tinytext
4	PanneauGB	tinytext
5	PanneauHB	tinytext
6	Porte	tinytext
7	Tasseau	tinytext
8	TraverseAR	tinytext
9	TraverseAV	tinytext
10	TraverseGD	tinytext


Commande

#	Nom	Type
1	idCommande 	int(11)
2	idClient 	int(11)
3	NbArmoire	int(11)

Client

#	Nom	Type
1	idClient 	int(1)
2	NomClient	tinytext
3	PrenomClient	tinytext

Fournisseur

Nom	Type
idFournisseur 	int(11)
Nom	tinytext
Prenom	tinytext
Adresse	tinytext
CodePostale	tinytext

Importation de la base de données

1. Ouvrir Wampserver
2. Ouvrir navigateur
3. Aller sur localhost/phpmyadmin



4. Connecter vous avec root et sans mot de passe.

A screenshot of the 'Connexion' (Connection) form in phpMyAdmin. The form has a title 'Connexion' with a help icon. It contains three input fields: 'Utilisateur :' (User) with the value 'root', 'Mot de passe :' (Password) which is empty, and 'Choix du serveur :' (Server choice) with a dropdown menu showing 'MySQL'. At the bottom right is a button labeled 'Exécuter' (Execute).

5. Créer une database nommé kitbox_database

A screenshot of the phpMyAdmin interface. On the left, a tree view shows databases: 'Nouvelle base de données', 'csv_db', 'information_schema', 'iothings', and 'kitbox_database'. The 'kitbox_database' is highlighted. Below the tree is a toolbar with buttons: 'Structure', 'SQL', 'Rechercher', 'Requête', 'Exporter', 'Importer', 'Opérations', 'Privilèges', and 'Plus'. Below the toolbar, there is a form to create a new table: 'Nom de table: kitbox_database', 'Ajouter 1 colonne(s)', and an 'Exécuter' button.

6. Aller sur l'onglet importer

Fichier à importer :

Le fichier peut être compressé (gzip, bzip2, zip) ou non.

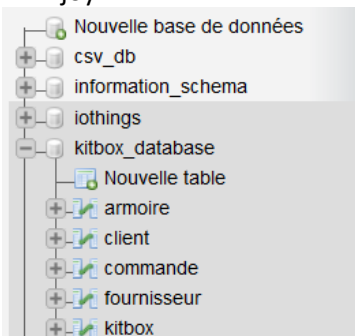
Le nom du fichier compressé doit se terminer par **[format].[compression]**. Exemple : **.sql.zip**

Parcourir les fichiers : kitbox_database.sql (Taille maximale : 128Mio)

Il est également possible de glisser-déposer un fichier sur n'importe quelle page.

Jeu de caractères du fichier :

7. Enjoy



Conception

Armoire

#	Nom	Type
1	idArmoire 🔑	int(11)
2	Corniere	int(11)
3	PanneauAR	tinytext
4	PanneauGB	tinytext
5	PanneauHB	tinytext
6	Porte	tinytext
7	Tasseau	tinytext
8	TraverseAR	tinytext
9	TraverseAV	tinytext
10	TraverseGD	tinytext

Cette table contiendra toutes les références pour la conception de l'armoire

Client

#	Nom	Type
1	idClient 🔑	int(1)
2	NomClient	tinytext
3	PrenomClient	tinytext

Cette table contiendra les informations concernant le client


Commande

#	Nom	Type
1	idCommande 🔑	int(11)
2	idClient 🔑	int(11)
3	NbArmoire	int(11)

Cette table représente le lien entre une armoire et un client.

Un client peut acheter plusieurs fois la même armoire lors d'une même commande.

Fournisseur

Nom	Type
idFournisseur 	int(11)
Nom	tinytext
Prenom	tinytext
Adresse	tinytext
CodePostale	tinytext

Cette table contiendra les informations concernant le fournisseur

Kitbox

#	Nom	Type
1	Reference	varchar(11)
2	CodeBarre	varchar(11)
3	Dimensions	varchar(14)
4	Hauteur	int(11)
5	Profondeur	int(11)
6	Largeur	int(11)
7	Couleur	varchar(9)
8	Enstock	int(11)
9	StockMinimum	int(11)
10	PrixClient	varchar(6)
11	NbPiecesCasier	int(11)
12	PrixFourn1	varchar(5)
13	DelaiFourn1	int(11)
14	NbPiecesComFourn1	int(11)
15	PrixFourn2	varchar(6)
16	DelaiFourn2	int(11)
17	NbPiecesComFourn2	int(11)

Cette table contient le détail du stock et des informations spécifiques concernant les pièces de l'armoire.

Requêtes SQL

Connection

```
using MySql.Data.MySqlClient;

namespace WinForms_MySQL
{
    public partial class frmWM : Form
    {
        private MySqlConnection connection;
        public frmWM()
        {
            this.InitConnexion();
            InitializeComponent();
        }
        private void InitConnexion()
        {
            // Création de la chaîne de connexion
            string connectionString = "SERVER=127.0.0.1; DATABASE=test; UID=root; PASSWORD=";
            this.connection = new MySqlConnection(connectionString);
        }
    }
}
```

Cela permet de créer la chaîne de connexion :

```
using MySql.Data.MySqlClient;

private MySqlConnection connection;
public frmWM()
{
    this.InitConnexion();
    InitializeComponent();
}
private void InitConnexion()
{
    // Création de la chaîne de connexion
    string connectionString = "SERVER=127.0.0.1; DATABASE=kitbox_database; UID=root;
PASSWORD=";
    this.connection = new MySqlConnection(connectionString);
}
```

Insertion

```
private void SaveDate(object sender, EventArgs e)
{
    this.connection.Open();
    // Création d'une commande SQL en fonction de l'objet connection
    MySqlCommand cmd = this.connection.CreateCommand();
    // Requête SQL
    cmd.CommandText = "INSERT INTO `winforms` (`Date`, `Temperature`, `Humidity`) VALUES('"
    // Exécution de la commande SQL
    cmd.ExecuteNonQuery();
    // Fermeture de la connexion
    this.connection.Close();
}
```

Cela permet d'insérer une nouvelle ligne dans la BDD

```
private void SaveDate(object sender, EventArgs e)
{
    this.connection.Open();
    // Création d'une commande SQL en fonction de l'objet connection
    MySqlCommand cmd = this.connection.CreateCommand();
    // Requête SQL
    cmd.CommandText = "INSERT INTO `winforms` (`Date`, `Temperature`, `Humidity`) VALUES('"
+ Convert.ToString(DateTime.Now) + "', '" + tbxTemperature.Text + "', '" + tbxHumidity.Text + "')";
    // Exécution de la commande SQL
    cmd.ExecuteNonQuery();
    // Fermeture de la connexion
    this.connection.Close();
}
```

INSERT dans la table ____ dans les colonnes ____, ____ ... les valeurs suivants ____, ____ .

On remplacera le winforms par le nom de la table.

Date, temperature ... par les noms de colonnes souhaité.

tbxTemperature.Text ... par la valeur souhaitée

/*! ne pas oublier de fermer la connexion.

Selection

```
private void LoadData(object sender, EventArgs e)
{
    string ConnectionString = "server=localhost;uid=root;pwd=;database=test";
    MySql.Data.MySqlClient.MySqlConnection Connection;
    Connection = new MySql.Data.MySqlClient.MySqlConnection(ConnectionString);
    Connection.Open();

    string CommandString = "SELECT * FROM winforms";
    MySql.Data.MySqlClient.MySqlCommand Command;
    Command = new MySql.Data.MySqlClient.MySqlCommand(CommandString, Connection);
    MySql.Data.MySqlClient.MySqlDataReader Reader;
    Reader = Command.ExecuteReader();

    while (Reader.Read()) {
        lblDate.Text = Reader.GetString("Date");
        lblTemperature.Text = Reader.GetString("Temperature");
        lblHumidity.Text = Reader.GetString("Humidity");
        string sDate = Reader.GetString("Date");
        cbxChoose.Items.Add(Reader.GetString("sDate"));
        //cbxChoose.DataSource = ds.Tables[0];
    }
}
```

Cela permet de sélectionner la DERNIERE ligne

```
private void LoadData(object sender, EventArgs e)
{
    string ConnectionString = "server=localhost;uid=root;pwd=;database=test";
    MySql.Data.MySqlClient.MySqlConnection Connection;
    Connection = new MySql.Data.MySqlClient.MySqlConnection(ConnectionString);
    Connection.Open();

    string CommandString = "SELECT * FROM winforms";
    MySql.Data.MySqlClient.MySqlCommand Command;
    Command = new MySql.Data.MySqlClient.MySqlCommand(CommandString, Connection);
    MySql.Data.MySqlClient.MySqlDataReader Reader;
    Reader = Command.ExecuteReader();

    while (Reader.Read()) {
        lblDate.Text = Reader.GetString("Date");
        lblTemperature.Text = Reader.GetString("Temperature");
        lblHumidity.Text = Reader.GetString("Humidity");
        string sDate = Reader.GetString("Date");
        cbxChoose.Items.Add(Reader.GetString("sDate"));
    }
}
```

Sélectionne toutes (*) les colonnes de la table ____ .

```
string sDate = Reader.GetString("Date");
```

La variable sDate contient toutes les valeurs de la colonne nommé Date

```
cbxChoose.Items.Add(Reader.GetString("sDate"));
```

On ajoute les données de la variable sDate dans la combobox.

Update

Permet de modifier la valeur d'une entrée.