

Documentation technique de Célia Baylet : Projet contexte karaté

Mission 3 : Célia, David

Créer un 'projet' puis pour les formulaires en haut à gauche 'Projet'-> 'Ajouter un formulaire (Windows Form)'

Il faut une page appeler : Program.cs qui sert à lancer le projet (ici lance le Form2) :

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Form2());
}
```

Une page pour la base de données ici 'basedd.cs' :

```
public static MySqlConnection ConnectBD()
{
    // Création de la chaîne de connexion
    string connectionString = "SERVER=localhost; DATABASE=karate; UID=root;
PASSWORD=";
    MySqlConnection conn = new MySqlConnection(connectionString);
    return conn;
}
```

La page 'Mission3.cs' sert à relier les 3 missions entre elles :

```
{
    Accueil frm = new Accueil();
    frm.Show();
}
```

Avec ce code qui est relier au bouton qui va aller vers la mission 1, le même code est effectué pour les 2 autres missions.

Avant de faire le code, on effectue le visuel. Donc des 'label' pour du texte, des 'Button' pour des boutons qui renvoie vers une autre page par exemple. Les 'textBox' pour rentrer un nom, prénom... Et les DataGridView pour avoir une vue sur certaines données ou sélectionner un entraineur par exemple. Lorsque l'on a effectué le visuel en faisant un double clic sur un bouton ou sur le 'load' l'arrière-plan, on est renvoyé dans la partie code à l'endroit souhaité. Il faut bien penser à ajouter un 'Name' cohérent aux boutons et DataGridView



Tache 1.3 : Gestion des entraineurs

Page 'Accueil' ou 'Form1.cs' est la première page

Sur chaque bouton on a créé un lien qui renvoie sur une autre page. Les boutons de modifications et suppression amène tt les 2 sur la même page. Ici ce code amène sur la page de Création.

```
{
    Création frm = new Création();
    frm.Show();
}
```

Avec la Création page 'Création.cs'

Ici dans le 'Load' il y a le code de connexion à la bdd. Puis une requête de type select pour récupérer les données à afficher dans la DataGridView (Rows pour afficher en ligne les données).

Lorsque le bouton ajouter est cliquer le code est effectué. Il faut penser à la connexion à la bdd. La requête est de type insert car la mission est de pouvoir ajouter un entraineur, il rentre son nom et prénom et choisit son club grâce à la DataGridView. Dans le code des paramètres il faut faire référence aux noms donnés lors de la création du design. On exécute la requête et le messageBox est juste là pour vérifier que le reste à bien fonctionner. Penser à fermer la connexion.

Modification et Suppression page 'Modifier'

Dans le load on se connecte à la bdd. La requête est de type select pour la DataGridView.

Sur le bouton modifier, on a un lien qui renvoie vers la page de modification :

```
Page_Modification frm = new
Page_Modification(dataGridViewEntraîneur.CurrentRow.Cells["NUM_ENTRAINEUR"].Value.ToString());
frm.Show();
```

Sur le bouton supprimer : La requête est de type Delete, il faut supprimer la clé primaire donc 'NUM_ENTRAINEUR' pour supprimer toute la ligne.

Et modification se fait grâce à la page 'Page Modification'

Elle contient une DataGridView de la table club. Le nom et prénom sont par défaut mis dans les cases pour la modification. Dans le 'load' il y a 2 connexions, la première pour la DataGridView. La 2nde pour le nom et prénom. L'entraîneur sélectionner à modifier sera par défaut choisis pour le modifier.

Il vérifie si la condition est remplie avec des données provenant de reader2. Si la condition est vraie, il parcourt chaque ligne de la DataGridView et cherche une correspondance dans la première colonne. Si une correspondance est trouvée, la cellule correspondante est définie comme cellule 'par défaut' dans le contrôle DataGridView.

Dans le bouton 'conclure', une requête type update pour la modification. Le numéro du club est changé avec la DataGridView. Le numéro de l'entraîneur n'est pas modifié car c'est une clé primaire.

Tache 2.3 : Affectation des entraineurs comme membre du jury

Page 'Form2.cs' est la première page.

Sur chaque bouton on a créé un lien qui renvoie sur une autre page. Ici 2 boutons un pour la création et un pour la suppression.

Avec la Création page 'creationcompet.cs'

Sur le bouton ajouter : On fait une requête du type select max. elle sélectionne la valeur maximale de la colonne NUM_JURY de la table juge où la valeur de la colonne NUM_COMPETITION correspond à une valeur spécifiée par le paramètre.

Cette ligne récupère la valeur de la colonne "maxjury" (qui est le résultat de la fonction MAX() appliquée à la colonne NUM_JURY) de la ligne lue précédemment et la convertit en un entier pour stockage dans la variable NumJury :

```
int NumJury = Convert.ToInt32(reader2["maxjury"]);
```

La requête du type insert sert à ce que l'entraîneur avec son numéro choisisse sa compétition. Le numéro du jury lui sera attribué en fonction du plus gd nbr déjà dans la base de données ; il aura le nbr supérieur à celui-ci.

Dans le load le code avec des requêtes select pour récupérer les informations dont ont à besoin dans la bdd pour afficher les DataGridView.

Suppression page 'modifcompet'

2 DataGridView. (Les mêmes que pour la création).

Un bouton pour afficher les compétitions lorsque l'on clique dessus les compétitions de l'entraîneur choisis s'affiche. Le bouton suppression si l'on souhaite supprimer l'une des compétitions affecter à un entraîneur.

Dans le load le code pour la DataGridView.

Dans le bouton afficher les compétitions, il y a les 2 requêtes de types select pour les DataGridView, il y a une boucle while pour afficher les numéros de compétition avec leur date seulement pour l'entraîneur choisit.

Dans le bouton supprimer, une requête de type DELETE, qui supprimer le num entraineur et le num compétition (clé composée de juge). Pour pouvoir supprimer l'affectation de l'entraineur à la compétition il faut passer par la table juge qui les relie.

Tache 2.6 : Affichage d'un bilan global sur une saison pour un compétiteur donné

Page 'Bilan.cs' est la première page. Puis suite

Un espace ('textBox') pour pouvoir rentrer la saison souhaitée. Le bouton suite pour poursuivre.

Le seul code est le code du bouton suite qui renvoie vers la page 'ListeDesParticipants'.

ListeDesParticipants.cs

Cette page est composée d'une DataGridView avec les informations de la table membre. Et d'un bouton suite.

```
String date_valeur;  
public ListeDesParticipants(string date_valeur)  
{  
    InitializeComponent();  
    this.date_valeur = date_valeur;  
}
```

Ces lignes de code permettent d'initialiser une variable. Cela peut être utile pour stocker des informations spécifiques. Cette variable est entrée en paramètre dans le code du load ensuite.

Dans le load il y a le code pour la DataGridView. La requête retourne toutes les informations de 'membre' et on utilise un INNER JOIN pour relier aux tables inscription (pour le numéro de licence), compétition (pour le num compétition), et l'on cherche l'année des compétitions. Avec le while on récupère les informations du membre relié à la saison/l'année rentrer dans le premier 'questionnaire'.

Le bouton suite renvoie à la page suivante 'Bilan_saison'.

Bilan_saison.cs

Cette page contient une DataGridView pour afficher le bilan du membre choisit avec son année sélectionner auparavant.

Comme dans la page précédente une variable est initialisée au début du code, qui sera le paramètre rentré.

On récupère toutes les informations souhaitées du membre choisit avec la clé primaire de membre soit un 'WHERE membre.NUM_LICENCE'