

# Fiche savoirs : programmation événementielle en C# partie 2

## Limiter les accès aux objets graphiques

Il est possible de rendre un objet graphique inaccessible. Cela peut permettre d'éviter certains tests et rend l'interface plus parlante pour l'utilisateur : il voit qu'il ne peut pas accéder à l'objet.

```
btnSupprimer.Enabled = false;
```

## Gérer le focus

Il est possible de définir l'objet qui doit avoir le focus.

```
txtTitre.Focus();
```

## Optimiser le code avec des fonctions

Les optimisations sont pour éviter des répétitions de code en plaçant un code utilisable plusieurs fois dans une fonction.

Exemple pour gérer la reposition du focus qui doit se faire plusieurs fois sur la même zone de texte.

Il est possible de créer automatiquement une fonction non événementielle à partir de ligne(s) de code sélectionnée(s).

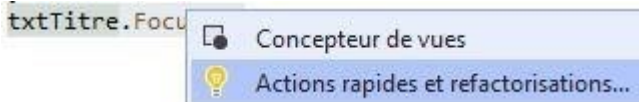
### Sélectionner le code à isoler :

À n'importe quel endroit, sélectionnez la ligne de code qui gère la position du focus sur la zone de texte.

```
txtTitre.Focus();
```

### Demander de refactorisation :

Clic droit sur la sélection et, dans le menu qui s'affiche, sélectionnez **Actions rapides et refactorisations...**



Demander de création de la procédure : Dans le nouveau menu, sélectionnez "Extraire une méthode". L'aperçu de droite montre ce qui va être fait : Création d'une méthode "NewMethod" contenant le code sélectionné. Remplacement de la sélection par l'appel de la méthode.

Renommer la méthode : Il vous est directement proposé de renommer la méthode. Directement sur le nom actuel qui est sur fond vert, tapez un nouveau nom (par exemple "GestionZoneDeSaisie") : remarquez que le nom de l'appel se change en même temps. Validez ou cliquez sur "Appliquer". La méthode est créée ainsi que le premier appel.

```
private void frmFilmsLoad(object sender, EventArgs e) { lblTitre.Text = "";
```

```
GestionZoneDeSaisie(); } Appeler la méthode : Partout où se trouve la ligne de code qui permet de remettre le focus sur la zone de texte, remplacez la ligne par l'appel de la méthode. private void GestionZoneDeSaisie() { txtTitre.Text = "";
```

```
txtTitre.Focus(); } Par exemple, au chargement du formulaire. Optimiser la méthode : Logiquement, lorsque le focus est replacé sur la zone de saisie, celle-ci doit être préalablement vidée. Ajouter la ligne correspondante. private void btnAjouterClick(object sender, EventArgs e) { string titre = txtTitre.Text.ToUpper(); if (!titre.Equals("")) { lstTitres.Items.Add(titre); lblTitre.Text = titre; txtTitre.Text = "";
```

```
}  
GestionZoneDeSaisie();
```

} Nettoyer le code : Dans le reste du code, enlevez les lignes qui vidant la zone de texte (attention, n'enlevez pas les lignes qui vident le label). A priori, ce n'est que sur le clic de btnAjouter.

From:

/ - **Les cours du BTS SIO**

Permanent link:

</doku.php/bloc1/evenementiel2?rev=1674163559>

Last update: **2023/01/19 22:25**

