

Secteur Tertiaire Informatique Filière étude - développement

Développer des composants d'interface

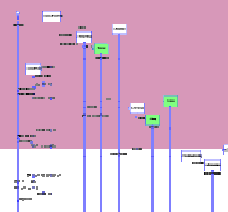
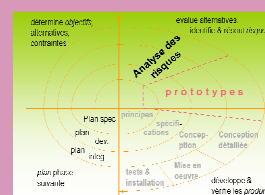
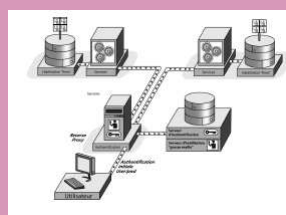
Créer des formulaires : Exercices

Accueil

Apprentissage

Période en
entreprise

Evaluation



Code barre

SOMMAIRE

| | | |
|---------|--|----|
| I | PHASE 1 : PREMIERS PAS..... | 4 |
| I.1 | L'additionneur | 4 |
| I.1.1 | Objectif..... | 4 |
| I.1.2 | But du traitement..... | 4 |
| I.1.3 | Les traitements..... | 4 |
| II | PHASE 2 : LA VALIDATION DE LA SAISIE..... | 5 |
| II.1 | Les contrôles de saisie..... | 5 |
| II.1.1 | Objectif..... | 5 |
| II.1.2 | But du traitement..... | 5 |
| II.1.3 | Les traitements..... | 5 |
| III | PHASE 3 : LES DIFFERENTS OBJETS GRAPHIQUES..... | 7 |
| III.1 | CHECK BOX et BOUTONS RADIOS..... | 7 |
| III.1.1 | Objectif..... | 7 |
| III.1.2 | But du traitement..... | 7 |
| III.1.3 | Les traitements..... | 7 |
| III.2 | Operations de base sur les LISTBOX..... | 9 |
| III.2.1 | Objectifs | 9 |
| III.2.2 | But du traitement..... | 9 |
| III.2.3 | Les traitements..... | 9 |
| III.3 | LISTBOX et COMBOBOX..... | 10 |
| III.3.1 | Objectifs | 10 |
| III.3.2 | But du traitement..... | 10 |
| III.3.3 | Les traitements..... | 10 |
| III.4 | Les composants de défilement..... | 12 |
| III.4.1 | Objectifs | 12 |
| III.4.2 | But du traitement..... | 12 |
| III.4.3 | Les traitements..... | 12 |
| III.5 | Synthese | 14 |
| III.5.1 | Objectifs | 14 |
| III.5.2 | But du traitement..... | 14 |
| III.5.3 | Les traitements..... | 14 |
| IV | PHASE 4 : MENUS, BARRES D'OUTILS ET D'ETAT | 16 |
| IV.1 | Menus, barres d'outils, barres d'état | 16 |
| IV.1.1 | Objectifs | 16 |
| IV.1.2 | But du traitement..... | 16 |
| IV.1.3 | Les traitements..... | 16 |
| IV.2 | Communication entre feuilles..... | 17 |

I PHASE 1 : PREMIERS PAS

I.1 L'ADDITIONNEUR

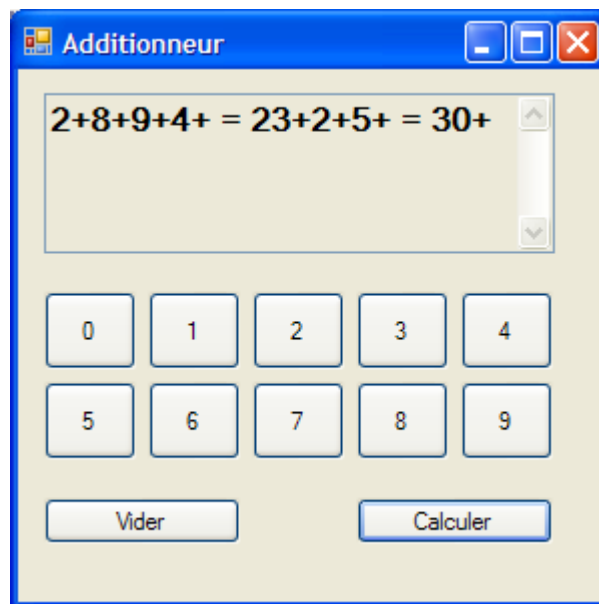
I.1.1 Objectif

Prendre en main l'environnement de développement.

S'initier à la programmation événementielle.

I.1.2 But du traitement

Programmer un additionneur par simple click sur les boutons de l'interface.



I.1.3 Les traitements

L'utilisateur ne peut additionner que des chiffres.

A chaque clic sur un chiffre, le chiffre suivi de " + " est ajouté à la TextBox (multiligne et avec scrollbars).

En cliquant sur le bouton **Calculer**, le résultat final de l'addition est affiché (suivi de " + " lui aussi).

En cliquant sur le bouton **Vider**, on vide la TextBox et remet à zéro le compteur. Le contenu de la TextBox ne peut pas être modifié autrement que par les boutons.

On quitte le programme en cliquant sur la croix de fermeture de fenêtre.

II PHASE 2 : LA VALIDATION DE LA SAISIE

II.1 LES CONTROLES DE SAISIE

II.1.1 Objectif

Savoir gérer la validité d'une zone de saisie alphabétique, numérique ou date.

II.1.2 But du traitement

Gérer la saisie dans une feuille comportant un champ alphabétique, un champ numérique entier, un champ comportant une date et un champ monétaire (voir feuille ci-dessous).

The image displays a Windows application interface. The main window, titled 'Les contrôles', contains four text input fields: 'Nom' (containing 'Duranton'), 'Date' (containing '20/04/2008' with a '(JJ/MM/AAAA)' format hint), 'Montant' (containing '3567.68'), and 'Code Postal' (containing '13013'). Below these fields are two buttons: 'Valider' and 'Effacer'. Overlaid on the right is a 'Validation effectuée' dialog box with a red 'X' icon, displaying the entered data: 'Nom : Duranton', 'Date : 20/04/2008', 'Montant : 3567.68', and 'Code : 13013', with an 'OK' button. Below this is a 'FIN' dialog box with a question mark icon, displaying 'Fin de l'application' and two buttons: 'Oui' and 'Non'.

II.1.3 Les traitements

La feuille n'est pas redimensionnable.

Les 4 champs sont **obligatoires** :

- txtNom.Text n'accepte que des caractères alphabétiques, champ de longueur 30 maxi.
- txtDate.Text accepte, bien sur une date valide, supérieure à la date du jour.
- txtMontant.Text accepte chiffres et séparateur décimal, et refuse les nombres négatifs.
- txtCP.Text n'accepte que 5 caractères numériques.

Le bouton « Effacer » efface les zones de saisie.

Le bouton « Valider » valide la saisie, et envoie un message de bonne fin.

La croix sert à quitter l'application : Envoyez une MessageBox de confirmation de fin d'application, qui permet de revenir dans le traitement.

Si une saisie n'est pas correcte, il faut envoyer un signal sonore, afficher un message d'erreur, positionner le focus sur le champ en erreur, bref, avertir l'utilisateur.

Plusieurs méthodologies différentes sont à tester à travers la réalisation de ce TP :

1. On peut contrôler tous les champs sur le click du bouton Valider. Réaliser ce premier scénario en choisissant les **bonnes** méthodes de contrôle pour chaque champ.
2. On peut également contrôler la saisie champ après champ, le bouton Valider étant clickable pour effectuer les « traitements métier ». Il faut alors choisir les **bons** événements, peut-être aussi les **bons** contrôles graphiques

Quels traitements sur quels événements préconisez-vous ?

III PHASE 3 : LES DIFFERENTS OBJETS GRAPHIQUES

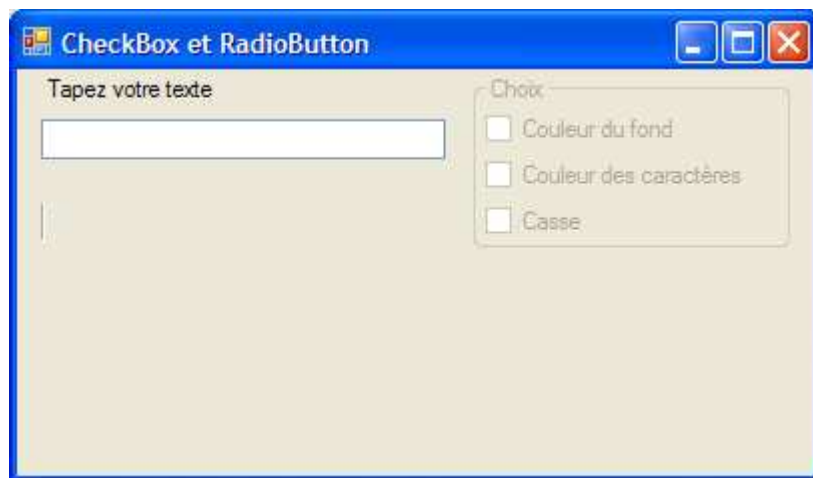
III.1 CHECK BOX ET BOUTONS RADIOS

III.1.1 Objectif

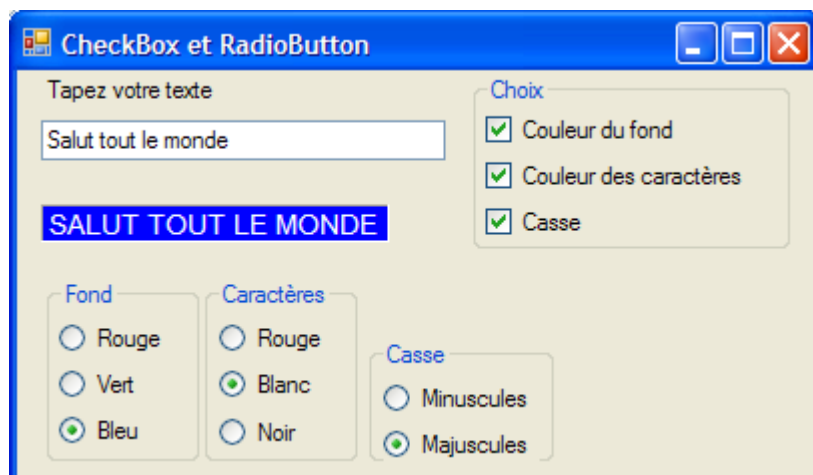
Savoir gérer des cases à cocher et groupes de boutons radios.

III.1.2 But du traitement

Au lancement du programme, la feuille se présente de la façon suivante:
Seule est accessible, la zone de texte où l'utilisateur peut saisir son texte



Puis en cours de programme, en fonction de l'option de choix sélectionnée, on pourra faire varier la couleur de fond, la couleur des caractères et la casse.



III.1.3 Les traitements

La feuille initiale permet uniquement de saisir un texte, copié caractère à caractère dans un label.

Au premier caractère entré, le groupe de choix est rendu accessible et devient inaccessible si le caractère est supprimé.

- La coche de la case à cocher **Couleur du fond** fait apparaître le groupe d'options **Fond**. Un élément coché dans ce groupe d'options modifie la couleur de fond du label (propriété BackColor).
- La coche de la case à cocher **Couleur des caractères** fait apparaître le groupe d'options **Caractères**. Un élément coché dans ce groupe d'options modifie la couleur de la police du texte recopié (propriété ForeColor).
- La coche de la case à cocher **Casse** fait apparaître le groupe d'options **Casse**. Un élément coché dans ce groupe d'options fera passer en minuscules/majuscules le texte recopié.

La décoche de chaque case à cocher provoquera la disparition du groupe d'options correspondant.

On utilisera la structure `System.Drawing.Color`.

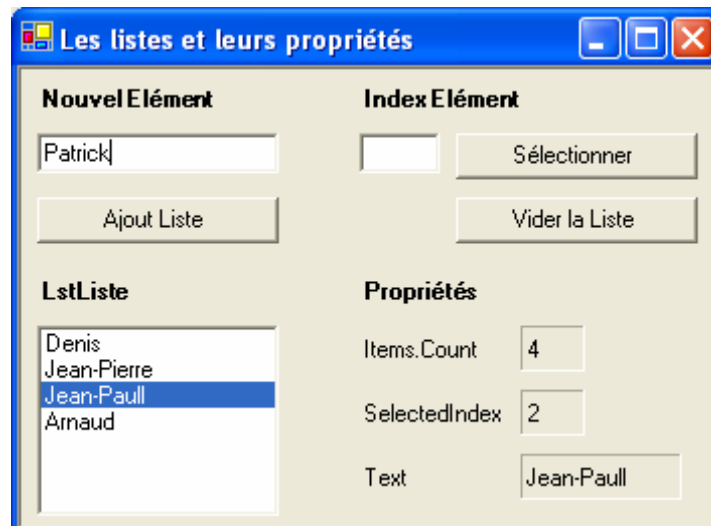
III.2 OPERATIONS DE BASE SUR LES LISTBOX

III.2.1 Objectifs

Etre en mesure de réaliser les opérations principales concernant les contrôles de type **ListBox**.

III.2.2 But du traitement

Visualiser les propriétés de la ListBox dans son état vide, lors de l'ajout ou la sélection d'un élément



III.2.3 Les traitements

Lors du lancement de l'application, la liste **LstListe** est vide. Le peuplement de la liste s'effectue en saisissant des valeurs dans la TextBox **Nouvel Elément** puis en pressant le bouton **Ajout Liste**. Une fois un nouvel élément ajouté, le focus se repositionne sur **Nouvel Elément** qui est remise à blanc.

Lorsque l'utilisateur sélectionne un élément, les propriétés affectées s'affichent à droite. Il est également possible de sélectionner un élément en saisissant son N° dans la TextBox **Index Elément** et en pressant le bouton **Sélectionner**.

Enfin, le bouton **Vider la liste** comme son nom le laisse supposer remet la liste à blanc.

Si vous avez du temps

Faites en sorte qu'un utilisateur maladroit ne puisse pas provoquer une erreur d'exécution de votre application. Exemple : s'il saisit un N° d'élément qui n'existe pas ou sous forme de texte.

Assurez-vous également qu'une valeur ne puisse pas être ajoutée deux fois dans la liste.

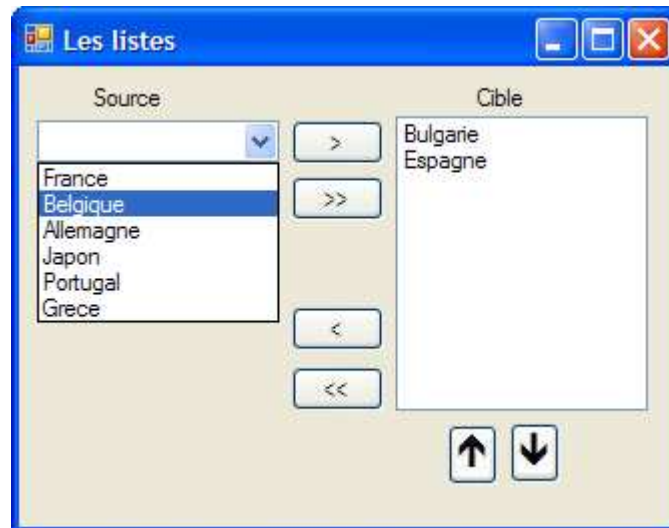
III.3 LISTBOX ET COMBOBOX

III.3.1 Objectifs

Savoir manipuler les contrôles de liste, ListBox et ComboBox.





III.3.2 But du traitement



Gérer le passage d'informations entre une listBox et une comboBox, élément par élément ou l'ensemble des éléments en une seule opération.



III.3.3 Les traitements

Les informations sont chargées dans la liste source au démarrage de l'application, en chargeant la liste, soit ligne après ligne, soit en utilisant un tableau de stockage. Tout élément saisi dans la partie texte de la combo peut être ajouté en cours de traitement sur l'évènement **DropDown**, s'il n'existe pas déjà ni dans la liste, ni dans la combo.

- Un click sur le bouton  ajoute l'élément sélectionné de la combo source dans la liste cible ; l'élément est supprimé de la source. Ce bouton est désactivé si aucun élément de la combo source n'est sélectionné.
- Un click sur le bouton  ajoute tous les éléments de la combo source dans la liste cible ; tous les éléments sont supprimés de la source. Ce bouton est désactivé si la combo source est vide.
- Un click sur le bouton  ajoute l'élément sélectionné de la liste cible dans la combo source ; l'élément est supprimé de la cible. Ce bouton est désactivé si aucun élément de la liste cible n'est sélectionné.
- Un click sur le bouton  ajoute tous les éléments de la liste source dans la combo source ; tous les éléments sont supprimés de la cible. Ce bouton est désactivé si la liste source est vide.

- Un click sur le bouton  déplace l'élément sélectionné de la cible d'une ligne vers le haut. Ce bouton est désactivé si aucun élément de la liste cible n'est sélectionné, ou si le premier élément est sélectionné.
- Un click sur le bouton  déplace l'élément sélectionné de la cible d'une ligne vers le bas. Ce bouton est désactivé si aucun élément de la liste cible n'est sélectionné, ou si le premier élément est sélectionné.

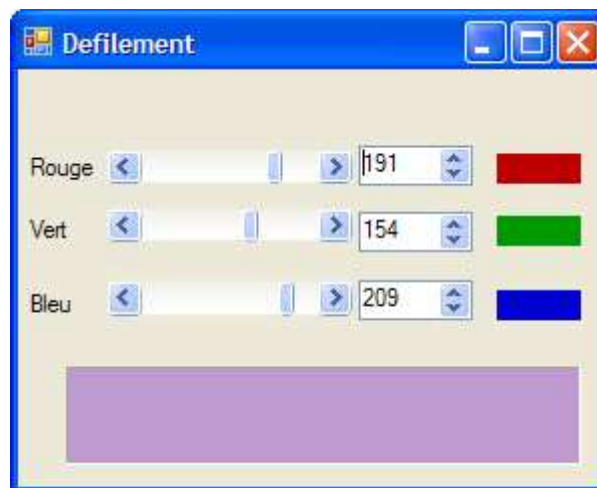
III.4 LES COMPOSANTS DE DEFILEMENT

III.4.1 Objectifs

Savoir utiliser les propriétés et méthodes des composants de défilement.

III.4.2 But du traitement

Chaque barre de défilement (HScrollBar) permet de faire varier la couleur de chaque fondamentale, Rouge, Vert ou Bleu obtenant la couleur de l'étiquette positionnée en fin de ligne. La couleur résultant du mélange des trois est affichée à l'aide d'une quatrième étiquette au bas de la feuille.



III.4.3 Les traitements

Chaque barre de défilement (HScrollBar) permet de faire varier la couleur de chaque fondamentale, Rouge, Vert ou Bleu obtenant la couleur de l'étiquette positionnée en fin de ligne.

La couleur résultant du mélange des trois est affichée à l'aide d'une étiquette au bas de la feuille.

L'utilisateur peut également obtenir la couleur désirée en positionnant dans les NumericUpDown de chaque ligne la valeur souhaitée.

Les évènements utilisés :

- L'évènement Scroll du contrôle HScrollBar
- L'évènement ValueChanged du contrôle HScrollBar
- L'évènement ValueChanged du contrôle NumericUpDown

Les méthodes utilisées :

- La méthode `Color.FromArgb(red, green, blue)`

Où *red*, *green*, *blue* sont des valeurs comprises entre 0 et 255.

Les propriétés utilisées :

- Les propriétés Minimum, Maximum , SmallChange, LargeChange et Value du contrôle HsScrollBar.
Attention : La valeur maximale pouvant être atteinte à l'exécution équivaut à la valeur de la propriété Maximum moins la valeur de la propriété LargeChange plus 1.
- Les propriétés Minimum, Maximum et Value du contrôle NumericUpDown.

Cet exercice peut être fait de la même façon avec des contrôles **TrackBar**.

III.5 SYNTHÈSE

III.5.1 Objectifs

Savoir utiliser propriétés et méthodes des contrôles de base dans une application.

III.5.2 But du traitement

Nous devons réaliser un petit programme avec une seule fenêtre pour calculer le montant périodique d'un prêt en fonction de son taux, de la périodicité de remboursement, de la durée du prêt et bien sûr de la somme empruntée.

The screenshot shows a Windows application window titled "Emprunts". It has a blue title bar with standard window controls. The main area contains several input fields and controls:

- Nom:** An empty text box.
- Capital Emprunté:** A text box containing "150000".
- Durée en mois du remboursement:** A text box containing "120" next to a slider control.
- Périodicité de remboursement:** A list box with options: Mensuelle, Bimestrielle, Trimestrielle (highlighted), Semestrielle, and Annuelle.
- Taux d'intérêt:** A group box containing three radio buttons: 7%, 8% (selected), and 9%.
- Buttons:** "OK" and "Annuler" buttons are located to the right of the interest rate group.
- Results:** At the bottom right, the text "40 Remboursements" and "5 483,36 €" are displayed in red.

III.5.3 Les traitements

La formule de calcul est la suivante :

Montant de remboursement = $K * (t / (1 - (1 + t)^{-n}))$

Où

- K : est le capital emprunté,
- t : est le taux annuel,
- n : est le nombre de remboursements.

Attention

Le taux donné par les boutons radio est un taux annuel, il faudra le diviser par 12 si l'on veut un taux mensuel.

Il peut y avoir des remboursements mensuels, bimestriel, trimestriels, semestriels ou annuels.

Le nom de la personne est saisi de manière facultative, par contre toutes les autres zones sont obligatoires. Il faut contrôler leur présence et leur validité.

La durée en mois nombre du remboursement, donné par l'ascenseur doit s'afficher en clair ; si pour des remboursements mensuels, ce chiffre évolue de un en un, pour des remboursements trimestriels, par exemple, il évoluera de trois en trois.

Dès le chargement de la feuille, positionner le curseur sur le « nom », le taux d'intérêt sélectionné doit être « 7% », la durée en mois de remboursement doit être égale à « 1 », ainsi que le nombre de remboursements en Euros, et la sélection doit être « Mensuelle » dans la périodicité de remboursement.

A chaque changement de capital, de taux de périodicité et de durée de remboursement, ou de périodicité, le montant du remboursement sera réinitialisé.

Sur le clic du bouton OK, afficher un message d'erreur si le capital n'est pas saisi.

Dans la zone *capital emprunté*, il ne peut y avoir que des chiffres, on limite à 10 chiffres sans décimale cadrés à droite. Le champ Nom s'il est saisi, ne doit comporter que des caractères alphabétiques.

Un contrôle d'erreur peut être positionné en cas de saisie incorrecte.

On fait grossir le curseur de la barre horizontale en fonction de la périodicité.

Vous avez en votre possession l'exécutable du programme réalisé. Faites le *tourner* et *retourner*.

Etudiez tout d'abord l'interaction entre la liste Périodicité de remboursement et la scrollbar Durée du remboursement :

- Quel est l'effet de la sélection d'un item de la liste sur la ScrollBar ? : quelles sont les propriétés affectées ?
- Comment évolue le label qui contient la durée en mois lorsque le curseur de la ScrollBar est actionné ? Comment évolue-t-il lorsqu'on change la périodicité ?
- A quel moment est calculé le nombre de remboursements ?

Mettez en place les contrôles, le calcul du taux d'intérêt, puis le calcul du montant du remboursement.

Contrôlez vos résultats par rapport aux résultats donné par le programme.

IV PHASE 4 : MENUS, BARRES D'OUTILS ET D'ETAT

IV.1 MENUS, BARRES D'OUTILS, BARRES D'ETAT

IV.1.1 Objectifs

Utiliser les contrôles de menus

IV.1.2 But du traitement

Il s'agit de rassembler sous une feuille Menu, par exemple par phase, tous les exercices réalisés pendant cette séance de formation.

On ajoutera également une barre d'outils et une barre d'état.

Chaque click sur une option de menu ou dans la barre d'outil appelle la feuille concernée. A l'initialisation, seul le menu Connection et le bouton s'identifier de la barre d'outils sont accessibles.



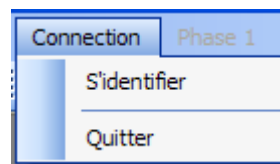
IV.1.3 Les traitements

Si tous vos exercices ont été réalisés dans des projets différents, démarrer un nouveau projet de nom Menu.

Renommez le formulaire Form1 en frmMenu, et positionnez ce formulaire comme conteneur MDI.

Importez chacune des feuilles des exercices réalisés, en faisant attention de renommer l'espace de nom sur toutes les feuilles d'extension .cs et .designer.cs.

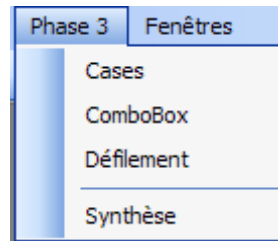
Le menu Connection



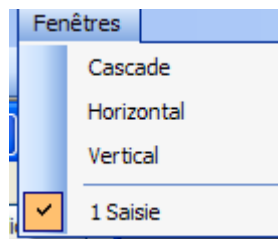
- Un click sur **S'identifier**, enverra dans un premier temps un message de bienvenue, et permettra l'accessibilité à tous les menus de l'application.
- **Quitter** permettra de quitter l'application, après confirmation par une boîte de dialogue.

Les **menus de phase** présentent les titres des différents exercices réalisés pendant cette séance de formation.

Exemple : Phase 3



Le menu **Fenêtres** présente les différentes façons d'arranger les fenêtres dans un environnement MDI, et la liste des fenêtres ouvertes.



La barre d'outils présente :

- Un **ToolStripButton** permettant de s'identifier
- Un **ToolStripSplitButton** reprenant les items du menu Phase 3.

La barre d'état présente :

- Un **ToolStripStatusLabel** initialisé avec la date du jour
- Un **ToolStripStatusLabel** indiquant la dernière opération effectuée.

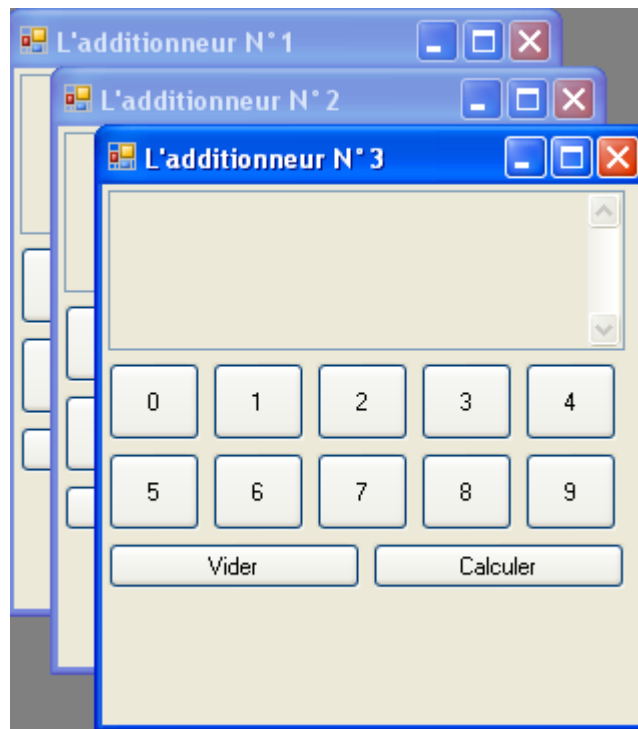
A l'affichage de la feuille, seuls le menu **Connection** et le bouton **S'identifier** de la barre d'outils sont accessibles : L'utilisateur doit s'identifier pour travailler sur l'application.

IV.2 COMMUNICATION ENTRE FEUILLES

Dissocier la saisie du texte de la feuille **Cases à cocher et boutons radios** en créant une feuille de saisie qui transmettra en paramètre le texte saisi dans la TextBox au label de la feuille codée dans la phase 3.



Vous pouvez également numéroter les feuilles de l'additionneur.



S'il vous reste du temps (et du courage !), vous pouvez créer une nouvelle feuille qui permettra à l'utilisateur de s'identifier : Vous mettrez en place une règle de validation (par exemple l'identification pourrait être bonne, si le login saisi est égal au mot de passe).



Un click sur le bouton OK validera la saisie et rendra accessible les menus de l'application. Une piste possible :: Rendez public (en utilisant les propriétés get/set de C#) les menus de la feuille MDI, de manière à ce que leur propriété **enabled** soit accessible par cette feuille de login.

Etablissement référent

Marseille Saint Jérôme

Equipe de conception

Elisabeth Cattaneo

Remerciements :

Aux formateurs de St Brieuc, Caen et Clermont

Reproduction interdite

Article L 122-4 du code de la propriété intellectuelle.

« toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droits ou ayants cause est illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction par un art ou un procédé quelconques. »

Date de mise à jour 24/03/2009

afpa © Date de dépôt légal mars 09

