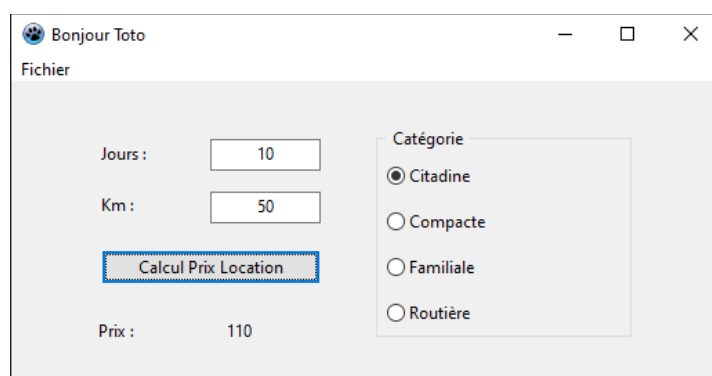


Nous travaillons avec les interfaces graphiques (formulaires et boîtes de dialogues) à partir de maintenant.

1. Location de véhicules

On vous demande d'élaborer une application qui permet de calculer le montant de la location d'un véhicule à partir du nombre de jours (entier), du kilométrage parcouru (entier), et de sa catégorie (4 valeurs possibles : « Citadine », « Compacte », « Familiale », « Routière »).

Voici l'interface graphique de votre formulaire principal (il n'y a qu'un seul formulaire de toute manière dans l'application) :



Nous distinguons tour à tour : des étiquettes (TLabel ; « caption » permet de modifier l'affichage), des zones d'édition (TEdit ; « text » permet de gérer les valeurs saisies par l'utilisateur), un bouton de calcul (TButton ; on se servira de l'évènement « OnClick » pour déclencher les calculs, « caption » modifie l'étiquette de l'objet) et un groupe de boutons radio (TRadioGroup ; « Items » permet d'énumérer les différentes chaînes à faire apparaître dans la structure, « ItemIndex » fournit le numéro de l'élément sélectionné [négatif = pas de sélection, zéro = premier élément], initialisez le composant pour être sûr qu'au moins une des catégories soit sélectionnée).

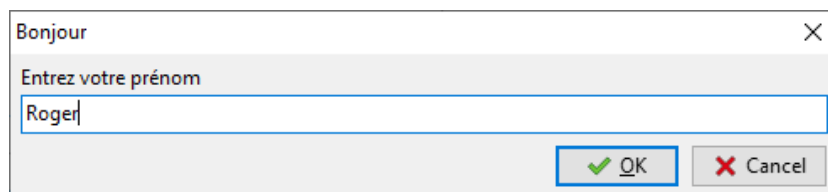
Lors du « click » sur l'objet bouton « Calcul Prix Location », le montant de la location est affiché dans le label en face de l'étiquette « Prix ». Voici le mode de calcul :

- « Citadine » : $11.0 * \text{jour}$;
- « Compacte » : $10.0 * \text{jour} + 0.3 * \text{km}$;
- « Familiale » : $6.0 * \text{jour} + 0.5 * \text{km}$;
- « Routière » : $5.0 * \text{jour} + 1.0 * \text{km}$.

Remarque : Si les valeurs saisies dans les zones d'édition de « jour » et « km » ne sont pas des entiers, un message d'erreur doit être affiché. Pour ce faire, plutôt que d'utiliser un StrToInt() pour la conversion des chaînes de caractères en numérique, utilisez plutôt [TryStrToInt\(\)](#).

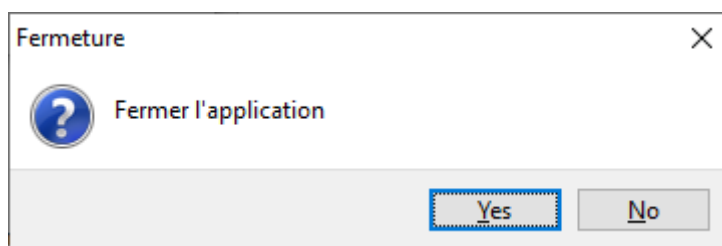
2. Démarrage et Fermeture du formulaire principal

Au démarrage de l'application, lors de l'affichage du formulaire principal (`OnShow`), on aimerait qu'une boîte de dialogue invitant l'utilisateur à saisir son prénom apparaisse (`InputBox`). La valeur par défaut « Toto » est utilisée si l'utilisateur ignore cette étape.



On doit alors voir apparaître la chaîne « Bonjour 'prénom' » dans l'en-tête (`Caption`) du formulaire principal (cf. la copie d'écran dans la première page du présent TD).

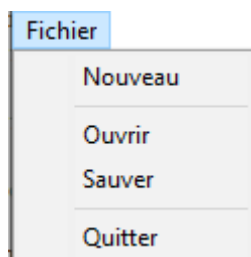
Une confirmation doit être demandée lorsque l'utilisateur souhaite fermer le formulaire principal (`OnCloseQuery`). Utilisez une boîte de dialogue standard `MessageDlg()` à cet effet. Voici un exemple d'affichage.



Si l'utilisateur clique sur « Yes », la fenêtre principale (et donc l'application) doit être effectivement fermée ; dans le cas contraire, il clique sur « No », on sursoit à sa fermeture.

3. Menu « Fichier »

Nous souhaitons ajouter un menu « Fichier » (`TMainMenu`) pour piloter l'application. Il apparaît comme suit :

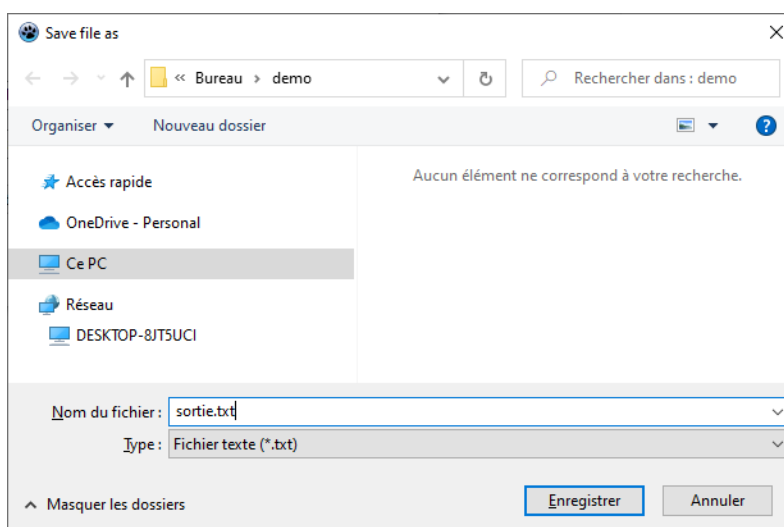


- « Nouveau » vise à remettre à zéro le contenu des zones d'édition « Jour » et « Km ».
- « Quitter » provoque la fermeture du formulaire principal, et donc de l'application.

- « Sauver » permet de sauvegarder dans un fichier texte le contenu des zones d'édition, dans un format qui sera explicité plus loin.
- « Ouvrir » permet de récupérer les valeurs de « Jour » et « Km » dans un fichier, puis de les faire apparaître dans les zones d'édition dédiées.

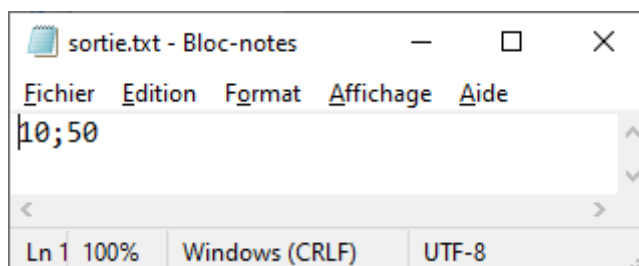
Fichier / Sauver

Lorsque l'utilisateur clique sur l'item de menu Fichier / Sauver, une boîte de dialogue (TSaveDialog) l'invitant à saisir le nom de fichier (extension « .txt ») par défaut doit apparaître.



S'il introduit un nom de fichier valide (l'extension doit être complétée automatiquement si elle est omise) et s'il clique sur Ok, le fichier doit être créé à l'emplacement indiqué. Il doit comporter respectivement les contenus des zones d'édition « jour » et « km », séparés par le caractère « ; ».

Il s'agit d'un fichier texte que l'on peut ouvrir avec n'importe quel éditeur de texte. Voici un exemple édité avec NotePad de Windows.



Fichier / Ouvrir

Le menu « Fichier / Ouvrir » agit à l'inverse de la sauvegarde : il doit demander à l'utilisateur de sélectionner un fichier, le parser en s'appuyant sur le séparateur « ; », extraire les valeurs de « jours » et « km » pour les faire afficher dans les zones d'édition dédiées.