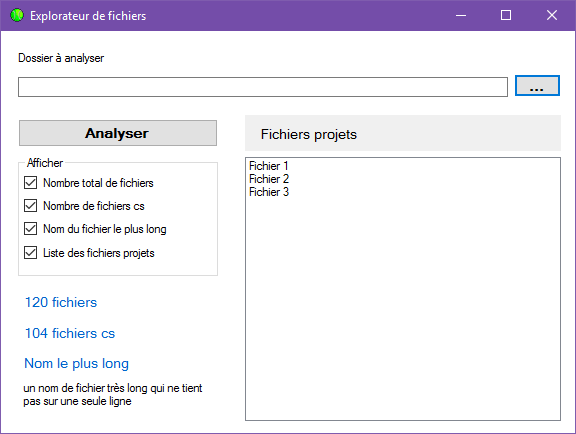
# Construction d’une interface visuelle

**Etape 1 :** Construire l’interface visuelle suivante :



**Etape 2** : Branchement de gestionnaires d’évènements

Lors du clic sur le bouton « … », afficher une boîte de dialogue de sélection de dossier, et affecter le fichier sélectionné à la zone de saisie (utiliser la classe FolderBrowserDialog)

Lors du clic sur les cases à cocher, afficher ou masquer les éléments visuels correspondants

**Etape 3** : Utilisation de paramètres d’application

Lors de la fermeture de l’application, faire en sorte que le dossier sélectionné, et l’état des cases à cocher soient mémorisé dans les paramètres utilisateur

**Etape 4** : message de confirmation

Lors de la fermeture de la fenêtre, afficher un message demandant si on veut enregistrer le chemin sélectionné et l’état des cases à cocher.

**Etape 5** : Utilisation de ressources

Rendre la fenêtre localisable en anglais. Tester en forçant la langue anglaise dans la méthode Main

Ajouter une icône dans la barre de titre de la fenêtre

**Etape 6** : logique applicative

Implémenter l’analyse du dossier sélectionner en utilisant le code de l’exercice de la veille (la séparation du visuel et de la logique applicative permet normalement de réutiliser les classes Analyseur et Explorateur sans modification)

# Récupération et affichage de données

Cet exercice utilise la base de données Northwind

**Etape 1** : Exécution d’une première requête

* Dans cotre solution Visual Studio, ajouter le projet ADO fourni
* Dans une classe DAL, créer une méthode pour récupérer la liste des régions. On se contentera dans un premier temps d’afficher les valeurs dans la fenêtre de sortie.
* Exécuter la méthode en debug pour vérifier les valeurs.

**Etape 2 :** Remplissage d’une DataGridView

* Créer une fenêtre fille nommée FormFournisseurs qui sera ouverte par clic sur un menu.
* Au chargement de cette fenêtre, exécuter une requête pour récupérer la liste des fournisseurs (tous les champs, sauf les 3 derniers)
* Afficher le résultat dans une DataGridView.

**Etape 3** : Remplissage d’une liste déroulante

Dans la même fenêtre, ajouter une liste déroulante Pays, et la charger à partir du résultat d’une requête qui récupère les différents pays des fournisseurs triés par ordre alphabétique

**Etape 4** : Requête paramétrable

Rendre la requête de récupération des fournisseurs paramétrable selon le pays

Dans la fenêtre précédente, lors de la sélection d’un pays dans la liste déroulante, exécuter cette requête en lui passant la valeur sélectionnée, puis afficher son résultat dans la DataGridView

NB/ La DataGridView doit afficher une liste filtrée dès l’ouverture de la fenêtre.

**Etape 5** : Requête scalaire

Dans la fenêtre précédente, ajouter un libellé qui affiche le nombre de produits fournis par l’ensemble des fournisseurs du pays sélectionné

**Etape 6** : Exécution d’une fonction

Dans la base de données, créer une fonction qui renvoie la liste des commandes passées par un client. Cette fonction prend l’Id client en paramètre, et renvoie pour chaque commande :   
Id, Date, Date d’envoi, nombre d’articles, Montant total, Frais d’envoi (tri selon la date)

Dans l’application, créer une nouvelle fenêtre FormCommandes, ouverte par clic sur un nouveau menu, et ajouter une zone de saisie d’un code client, un bouton « Voir les commandes » et une DataGridview.

Faire en sorte que lorsqu’on clique sur le bouton, la DatagridView affiche la liste des commandes du client dont on a saisi le code

**Etape 7** : Maître / détail

Supprimer le bouton et remplacer la zone de saisie par une ListBox affichant la liste complète des clients (code, nom et prénom).

Lorsqu’on double-clique sur un client, afficher sa liste de commandes.