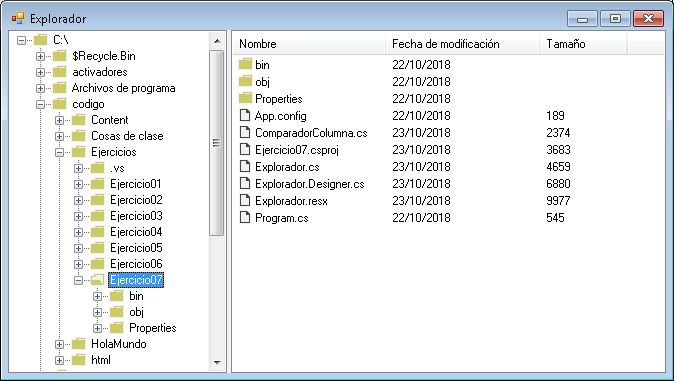
### UT2E7. Vista dividida, control árbol y lista ordenada.

Construye una aplicación de tipo explorador de archivos. Este tipo de aplicaciones consta en primer lugar de un SplitContainer para dividir el formulario en dos partes.

La parte izquierda está compuesta por un TreeView o árbol en el que se muestran los directorios de manera jerárquica.

La parte derecha está compuesta por un ListView compuesto por 3 columnas: nombre, fecha de modificación y tamaño. Su contenido será la lista de directorios y archivos del nodo seleccionado en el árbol.

El aspecto será el siguiente.



Para obtener información sobre un directorio y sus archivos emplearemos la clase “DirectoryInfo”, a modo de ejemplo las siguientes líneas obtienen la información de una ruta específica, sus subdirectorios y sus archivos.

DirectoryInfo directorio = new DirectoryInfo(@"C:\");

DirectoryInfo[] directorios = directorio.GetDirectories();

FileInfo[] archivos = directorio.GetFiles();

Tanto en el árbol como en la lista emplearemos imágenes, por lo que podremos **compartir en los dos casos el control ImageList.** Usaremos 3 iconos, el 0 para la imagen del directorio cerrado, el 1 para la imagen del directorio abierto o seleccionado, el 2 para la imagen del archivo.

Consideraciones SplitContainer

* El panel fijado será el izquierdo, el correspondiente al árbol.

Consideraciones TreeView

* La colección de nodos estará inicialmente vacía, se cargará el nodo raíz en el constructor.
* El árbol tiene asociado un control ImageList de uso común con el ListView.
* **Para añadir al nodo el símbolo “+” que permite expandirlo sin cargar los datos por adelantado deberemos añadirle un nodo “dummy” a su colección de nodos hijos.**
* **Al añadir un nodo al árbol siempre se guardar en la propiedad Tag del nodo el objeto DirectoryInfo asociado con la ruta.**
* Al crear un nuevo nodo deberemos asignarle el índice de imagen para los dos estados posibles, nodo no seleccionado y nodo seleccionado.
* El evento BeforeExpand leerá la propiedad tag del nodo que dispara el evento para recuperar el objeto DirectoryInfo asociado al nodo, si en la colección de nodos hijos únicamente está el nodo “dummy” deberemos limpiar la lista de nodos hijos y añadirle un nodo hijo por cada subdirectorio del directorio asociado al nodo.
* El evento AfterSelect leerá la propiedad tag del nodo que dispara el evento para recuperar el objeto DirectoryInfo asociado al nodo, limpiará las filas del listview y primero añadirá la información de los directorios y segundo añadirá la información de los archivos.

Consideraciones ListView

* La propiedad ListViewItemSorter permite definir un objeto encargado de las operaciones de ordenación de la lista, este objeto debe implementar la interfaz no genérica IComparer. Este comparador debe ser asignado a la lista en el constructor del formulario.
* Se definirán 3 columnas: nombre, fecha de modificación y tamaño, al pulsar sobre alguna de ellas en ejecución se ordenará la lista según los valores de la columna indicada, tanto en orden ascendente como descendente.
* El evento ColumnClick gestionara los criterios de ordenación que son: el número de columna por la que ordenar, el orden ascendente o descendente y finalmente invocara a la función del listview .Sort() para que se ordenen las filas. Tanto el número de columna como el orden son propiedades del objeto comparador que deberás implementar.
* Al añadir un nodo a la lista deberemos asignarle el índice de imagen según el tipo de elemento que represente, directorio o archivo.

Consideraciones comparador listview

* Implementa la interfaz IComparer
* Deberá tener las propiedades “NumeroColumna” de tipo entero y “Orden” de tipo SortOrder (es un enumerado de sistema), los valores por defecto serán columna 0 y orden “None”.
* El método Compare tendrá en cuenta el valor de las propiedades “NumeroColumna” y “Orden” para ordenar de forma correcta.
* Puede resultar de gran ayuda buscar el ejemplo de ordenación de un listview de la documentación oficial. (<https://support.microsoft.com/en-us/help/319401/how-to-sort-a-listview-control-by-a-column-in-visual-c>)

Al terminar sube a la plataforma un documento con formato zip y el nombre “UT2E7\_NOMBREAPELLIDOS” con el proyecto de código. **RECUERDA limpiar la solución antes de comprimirla**.