

“Programación .Net – Listas y Colecciones”

Actividades:

Manejo de ArrayList	1
Creación y manejo de ArrayList	1
Agregar elementos al ArrayList.....	1
Eliminar elementos del ArrayList.....	2
Manejo de Listas Genéricas	3
Creación y manejo de List<T>	3
Agregar elementos a List<T>	3
Eliminar elementos del List<T>	3
Manejo de Hashtable.	5
Creación de la clase.....	5
Creación y manejo de Hashtable	5
Agregar elementos al Hashtable.....	6
Eliminar elementos del Hashtable	6

Manejo de ArrayList

Crearemos una aplicación de consola en el cual mostraremos el contenido de la creación de colecciones con **ArrayList**, **List<T>** y **Hashtable**.

Elementos iniciales:

- a) Cree una solución de nombre **“Solucion Colecciones”** con un proyecto de consola de nombre **MisColecciones**.
- b) Agregue en esta nueva consola un menú que permita trabajar con las siguientes opciones:
 1. Manejo ArrayList
 2. Manejo List
 3. Manejo Hashtable
 4. Salir
- c) El detalle de acciones para cada opción se describe a continuación.

Creación y manejo de ArrayList

- a) En la Carga de ArrayList, debe solicitar el número de elementos que se cargarán, el que debe ser numérico mayor igual a 5 y menor igual a 10.
- b) Luego de capturar la cantidad de elementos, cree un ArrayList en base a la cantidad de elementos indicados. Cada elemento debe ser agregado como un string genérico de la forma “Elemento n”, donde “n” vendrá dado por un índice de 0 hasta la cantidad de elementos solicitados.
- c) Una vez creados los elementos, recorra la colección con un ciclo foreach y despliegue cada string por consola en líneas independientes.
- d) Ejecute la aplicación y genere una colección de elementos.

Agregar elementos al ArrayList

- a) En la misma opción deberá permitir agregar un nuevo elemento al ArrayList generado. Por lo que deberá generarse un submenú con las opciones de Carga Arraylist, Agregar Elemento, Insertar Elemento y Volver.
- b) La opción **Agregar Elementos** deberá solicitar el valor para agregar un nuevo elemento a la colección. Una vez agregado, se debe mostrar el contenido modificado de la colección.
- c) La opción **Insertar Elementos** deberá solicitar el valor para agregar un nuevo elemento a la colección y la posición donde se desea agregar. Se debe validar que la posición sea válida en el rango de posiciones de la colección. Una vez insertado el nuevo elemento, se debe mostrar el contenido modificado de la colección.

- d) Implemente un método que permita mostrar la colección, para evitar repetir este código cuando sea requerido.
- e) Ejecute la aplicación para generar y modificar la colección.

Eliminar elementos del ArrayList

- a) En la misma opción deberá permitir eliminar elementos del ArrayList generado. Por lo que deberá incluir estas nuevas opciones: Eliminar Elemento y Eliminar por Posición.
- b) La opción **Eliminar Elemento** deberá desplegar los elementos de la colección y solicitar al usuario el elemento a eliminar. Luego del proceso debe informar si el elemento pudo ser eliminado.
- c) La opción **Eliminar por Posición** debe desplegar los elementos de la colección informando su posición y solicitar al usuario la posición del elemento a eliminar. Luego del proceso debe informar si el elemento pudo ser eliminado o si la posición era incorrecta.
- d) Ejecute la aplicación para generar y eliminar elementos de la colección.

Manejo de Listas Genéricas

El ejercicio anterior mostró el trabajo con elementos agrupados mediante un ArrayList. Como se ha comentado, el problema que se puede presentar en el manejo con ArrayList, es que no se tiene control sobre el tipo de dato que se está almacenando en la lista, pudiendo ocasionar algún problema de casting al recuperar el objeto. El siguiente ejercicio demuestra como declarar y manejar las listas genéricas.

Creación y manejo de List<T>

- a) Cree de inmediato un submenú con las siguientes opciones:
 1. Carga List
 2. Agregar Elemento
 3. Insertar Elemento
 4. Eliminar Elemento
 5. Eliminar por Posición
 6. Volver
- b) Debe solicitar el número de elementos que se cargarán, el que debe ser numérico mayor igual a 5 y menor igual a 10, cada elemento debe ser agregado como un número múltiplo de 10 a partir de 1 (10, 20, 30, etc.)
- c) Considere la creación de un método para el despliegue de los valores de la colección.
- d) Una vez creados los elementos, recorra la colección con un ciclo foreach y despliegue cada elemento por consola en líneas independientes.
- e) Ejecute la aplicación y genere una lista de elementos.

Agregar elementos a List<T>

- a) La opción **Agregar Elementos** deberá solicitar el valor para agregar un nuevo elemento a la colección, como se trata de una lista de números, deberá validar el ingreso. Una vez agregado, se debe mostrar el contenido modificado de la colección.
- b) La opción **Insertar Elementos** deberá solicitar el valor para agregar un nuevo elemento a la colección y la posición donde se desea agregar. Se debe validar que el valor sea numérico y que la posición sea válida en el rango de posiciones de la colección. Una vez insertado el nuevo elemento, se debe mostrar el contenido modificado de la colección.
- c) Ejecute la aplicación para generar y modificar la colección.

Eliminar elementos del List<T>

- a) La opción **Eliminar Elemento** deberá desplegar los elementos de la colección y solicitar al usuario el elemento a eliminar, deberá validar que sea un elemento numérico. Luego del proceso debe informar si el elemento pudo ser eliminado.

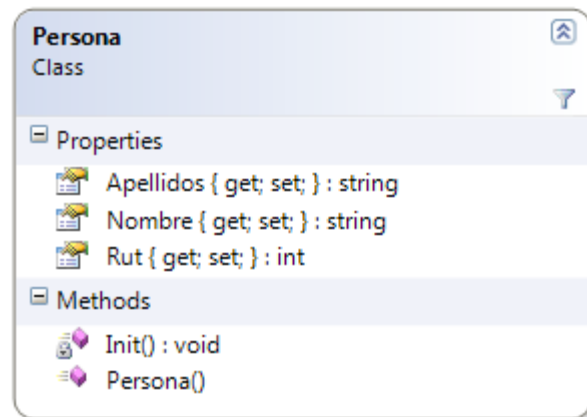
- b) La opción **Eliminar por Posición** debe desplegar los elementos de la colección informando su posición y solicitar al usuario la posición del elemento a eliminar. Luego del proceso debe informar si el elemento pudo ser eliminado o si la posición era incorrecta.
- c) Ejecute la aplicación para generar y eliminar elementos de la colección

Manejo de Hashtable.

Para este ejercicio se construirá una clase ejemplo para agrupar instancias de ellas en el Hashtable.

Creación de la clase

Se debe crear una clase Persona en una biblioteca de clases, en base a la siguiente definición.



Creación y manejo de Hashtable

- a) Cree un submenú con las siguientes opciones:
 1. Carga Personas
 2. Agregar Persona
 3. Insertar Persona
 4. Eliminar Persona
 5. Volver
- b) Debe solicitar el número de personas que se cargarán, el que debe ser numérico mayor a 0 y menor igual a 5.
- c) La carga creará un Hashtable en base a la cantidad de personas indicadas. Cada elemento debe ser creado de manera genérica de la forma: Rut=n, Nombre="Nombre n" y Apellidos="Apellido n". Para la llave del elemento utilice el valor de la propiedad Rut.
- d) Considere la creación de un método para el despliegue de los valores de la colección.
- e) Una vez creados los elementos, recorra el Hashtable con un ciclo foreach y despliegue cada elemento por consola en líneas independientes, utilizando las propiedades Rut, Nombre y Apellidos para desplegar.
- f) Ejecute la aplicación y genere una lista de personas.

Agregar elementos al Hashtable

- a) La opción **Agregar Personas** deberá solicitar los datos válidos requeridos para agregar una nueva persona a la colección. Una vez agregada, se debe mostrar el contenido modificado de la colección.
- b) La opción **Insertar Persona** deberá solicitar los datos válidos requeridos para agregar una nueva persona a la colección y la posición donde se desea agregar. Se debe validar que la posición sea válida en el rango de posiciones de la colección. Una vez insertada la nueva persona, se debe mostrar el contenido modificado de la colección.
- c) Ejecute la aplicación para generar y modificar la colección.

Eliminar elementos del Hashtable

- a) La opción **Eliminar Persona** deberá desplegar los elementos de la colección y solicitar al usuario el rut de la persona a eliminar, deberá validar que sea un valor numérico. Deberá buscar y comparar las personas en la colección, para encontrar la instancia que posee el número de Rut buscado; teniendo la instancia que se desea eliminar se debe ejecutar el método **Remove()** con esta instancia. Luego del proceso debe informar si la persona pudo ser eliminada.
- b) Ejecute la aplicación para generar y eliminar personas de la colección.