

**Trabajo Practico N° 2**  
**(Recuperatorio):**  
**Laboratorio de Programación**



*Ignacio Martín Canay*

### *Introducción:*

El presente documento detalla y explica las funcionalidades del Trabajo Practico Número 2. Este trabajo abarca todos los temas vistos entre las clases 1 y 15.

Se utilizaron todos los temas mencionados anteriormente y su implementación sera detallada a continuación.

El sistema pedido para este trabajo era a libre elección, se opto por realizar una Guardería de Dragones. En esta guardería se pueden Guardar distintos tipos de Dragones: Dragones Acuáticos, Dragones Terrestres y Dragones Voladores. Aquí mismo también se les proporciona comida y alimento apropiado según sus necesidades.

Para traducir estas funcionalidades al programa se hizo uso de una Clase abstracta Guardería, la cual consta de la lista de todos los dragones y la cantidad de espacio disponible en cada Bioma necesario para que los dragones los habiten.

También se realizo una clase Abstracta Dragón, de la cual heredan otras tres clases, correspondientes a cada tipo de dragón que acepta la guardería: DragonAcuatico, DragonTerrestre y DragonVolador. Los dragones tienen atributos comunes, como lo son el nombre, el tipo de alimentación y tamaño.

Los dragones acuáticos tienen un atributo específico para su clase que indica el tipo de agua que prefieren, y puede ser de dos tipos salada y dulce, ambos encapsulados en el enumerado TipoDeAgua.

Los dragones Terrestres tienen un atributo específico para su clase que indica si viven por debajo de la tierra o sobre ella. Estas opciones estan encapsuladas dentro del enumerado TipoDeSuperficie

Los dragones Voladores tienen un atributo específico para su clase que indica a donde les gusta hacer sus nidos, ya sea en arboles o en acantilados. Estas opciones se encuentran dentro del enumerado TipoDeNido.

La clase Dragon posee varios métodos:

el método Alimentar, es un método abstracto y devuelve un string. Cada una de las clases de dragones que heredan de esta sobrescriben este método para ajustarlo a sus necesidades. Este método es utilizado para alimentar apropiadamente a cada dragón según sus necesidades. Por Ejemplo si es un dragón acuático, prefiere agua salada y es carnívoro, este método devolverá “Se lanza carne al mar”.

Esta clase también posee sobrecarga de los operadores de + para agregar un dragón a una lista de dragones. Y los operadores == y != que comparan una lista con un dragón dado, y retorna si el dragón esta en la lista, en el caso de ==, o no lo esta en el caso de !=.

También en esta clase esta la sobrecarga del metodo ToString(), que retorna una cada indicando las características del dragón. Este método es llamado por cada una de las clases de Dragones para agregar a la descripción las características particulares de cada clase.

La clase Guardería tiene tres métodos que se corresponden con las 3 funcionalidades principales del programa.

El primero es AgregarDragon, que recibe como parámetro un dragón que se agregará a la lista de dragones de la guardería. Este método retorna un bool indicando si el dragón se pudo guardar o no. Este método usa las sobrecargas del operador + informada mas arriba.

El segundo método es AlimentarDragones, que no recibe parametros y devuelve un string con la descripción de como se alimento a cada dragón dependiendo del tipo de dragón y las especificaciones de cada uno.

El tercer método es ListarDragones no recibe parámetro alguno y retorna un string con las especificaciones de cada dragón. Este método hace uso de la sobrecarga del método ToString informada mas arriba.

En lo que a formularios respecta son cuatro en total.

El formulario principal es FormGuarderia, este formulario contiene tres botones, cada uno de ellos

llama a una instancia de uno de los otros tres formularios del programa.

El formulario FormRegistro contiene dos text box, tres combo box, dos botones y cinco labels.

Permite la carga de los datos necesarios para generar un nuevo Dragon. Al seleccionar el combo box bajo el label Tipo, se puede seleccionar el tipo de dragón que se va a registrar, y según cual sea la opción elegida, el combo box cmbxAtributoExtra y el lblAtributo cambian para mostrar las opciones y descripciones pertinentes al tipo de dragon elegido. El botón hace uso del método AgregarDragon enunciado mas arriba para agregar un dragón con los datos proporcionados en los controles.

El formulario FormVerGuarderia posee unicamente un rich Text Box que se inicializa con la lista de los dragones. Este formulario hace uso del método ListarDragones para completar la información.

El formulario FormAlimentar posee un rich Text Box y un botón. Al hacer click en el botón se ejecuta el método AlimentarDragones enunciado mas arriba y el rich text Box se carga con el string devuelto por este método.