

**Universidad Tecnológica Nacional  
Facultad Regional Avellaneda**



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

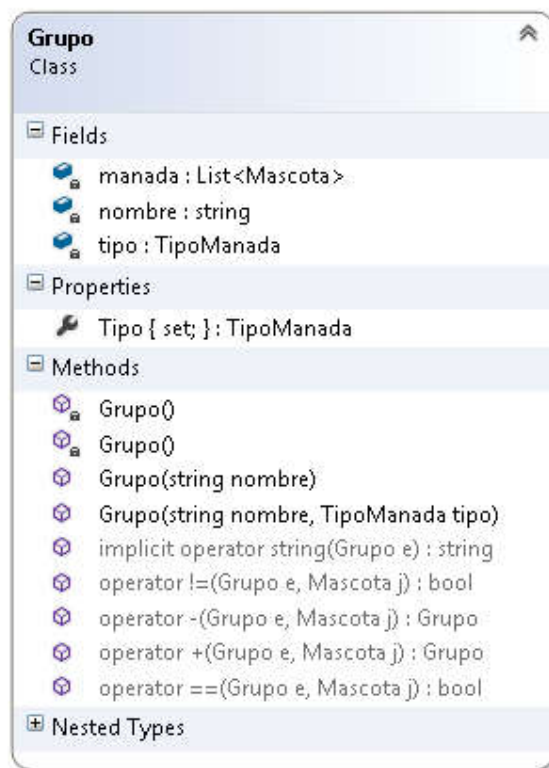
**Materia: Laboratorio de Computación II**

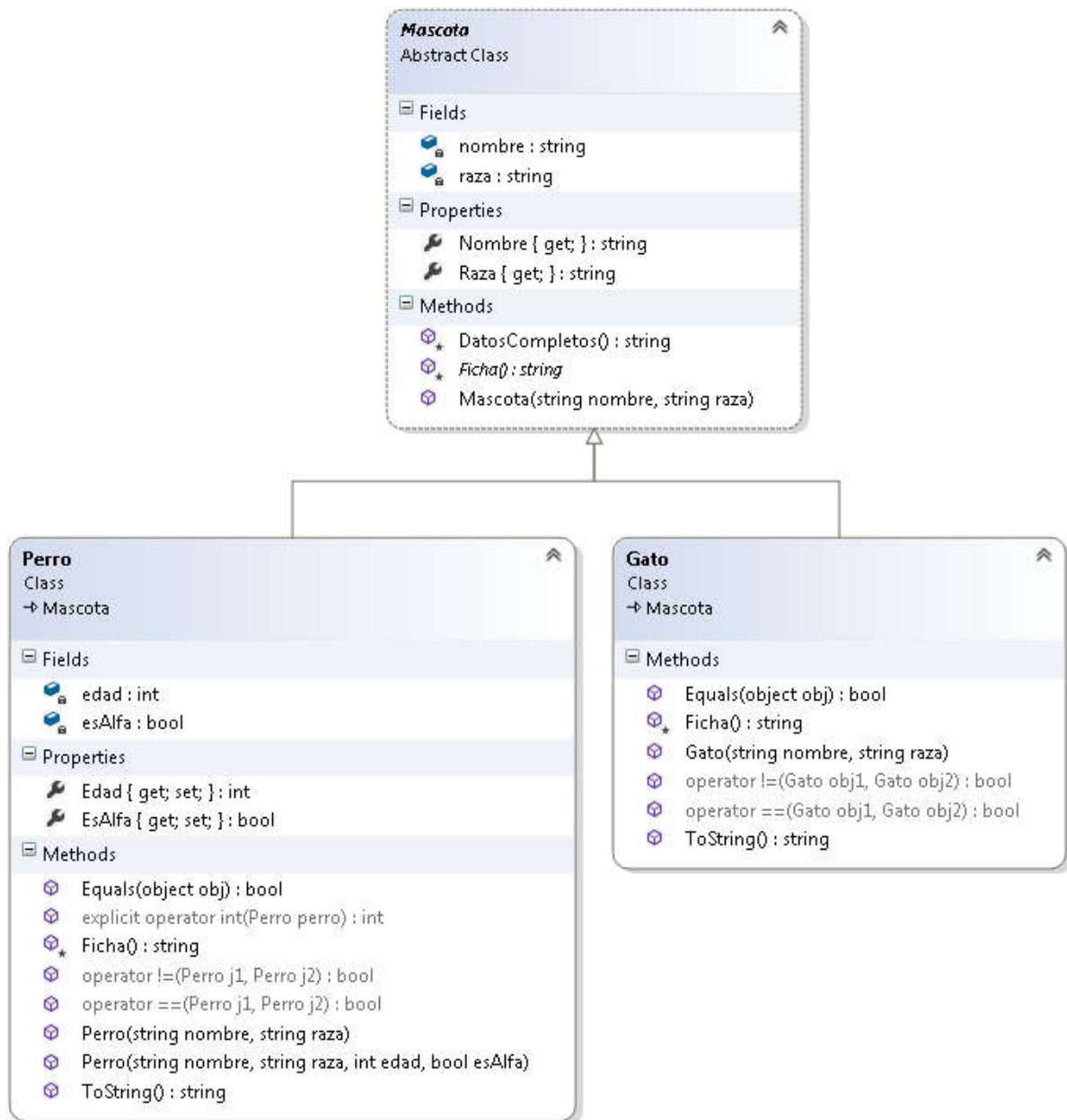
Apellido:		Fecha:	
Nombre:		Docente <sup>(2)</sup> :	
División:		Nota <sup>(2)</sup> :	
Legajo:		Firma <sup>(2)</sup> :	
Instancia <sup>(1)</sup> :	<div style="display: flex; justify-content: space-around;"> <div>PP</div> <div>X</div> <div>RPP</div> <div></div> <div>SP</div> <div></div> <div>RSP</div> <div></div> <div>FIN</div> <div></div> </div>		


(1) Las instancias validas son: 1<sup>er</sup> Parcial (**PP**), Recuperatorio 1<sup>er</sup> Parcial (**RPP**), 2<sup>do</sup> Parcial (**SP**), Recuperatorio 2<sup>do</sup> Parcial (**RSP**), Final (**FIN**). Marque con una cruz.

(2) Campos a ser completados por el docente.

Diagrama de clases:





- Generar un proyecto con el nombre: apellido.nombre.división. Por ejemplo Pérez.Juan.2E.
- Los proyectos que no sean identificables, no serán corregidos.
- Sólo se corregirá lo que el alumno entregue de la siguiente forma:
  - Al finalizar, colocar la carpeta de la Solución completa en un archivo ZIP y dejar este último en el Escritorio de la máquina. Luego presionar el botón  de la barra superior, cargar un mensaje y presionar Aceptar. La barra superior deberá cambiar de color.
- En todos los casos que sea posible, reutilizar código.

### *Mascota:*

Clase abstracta con dos atributos privados (nombre y raza). El único constructor recibirá dos parámetros. Las propiedades Nombre y Raza serán de sólo lectura.

Método abstracto y protegido `Ficha()`.

Método virtual y protegido `DatosCompleto()`. Retornará el nombre y el raza con el formato "Nombre Pitbull". Se deberá utilizar el método `Format` de la clase `String`.

### *Perro:*

Clase pública que hereda de Mascota con dos atributos propios (edad y esAlfa). Contará con dos constructores, en caso de no recibir número y es capitán deberán inicializarlos como 0 (cero) y false respectivamente.

Implementar el método `Ficha()`. Retornará toda la información del Perro con el siguiente formato:

- Si es alfa: Ramón Salchicha, alfa de la manada, edad 2
- Si no lo es: Julio Cruza edad 13

#### Operadores:

- Dos Perros serán iguales si comparten nombre, raza y edad.
- Conversión explícita de Perro a entero, retornando su edad.

#### Sobreescribir:

- Método `ToString()` para que publique la información del Perro.
- Método `Equals` para que reutilice `==`.

### *Gato:*

Clase pública que hereda de Mascota.

Implementar el método `Ficha()`. Retornará toda la información del Gato con un formato similar al de Perro.

#### Operadores:

- Dos Gatos serán iguales si comparten nombre y raza.

#### Sobreescribir:

- Método `ToString()` para que publique la información del Gato.
- Método `Equals` para que reutilice `==`.

### *Grupo:*

Enumerado `TipoManada` contendrá `Única`, `Mixta`.

Contendrá una lista de Mascotas, un nombre y atributo de clase tipo.

#### Constructores:

- De clase que inicializará el tipo como `Única`.
- Por defecto privado, será el único lugar donde se inicialice la lista.
- Otro recibirá nombre.
- El último recibirá nombre y tipo.

Una sola propiedad de sólo escritura para el tipo.

#### Operadores:

- Un Grupo será igual a una Mascota si esta última forma parte de la lista.
- Si una mascota no forma parte de la lista, se podrá agregar con el `+`.
- Si una mascota forma parte de la lista, se podrá quitar con el `-`.
- Conversión implícita a `String`, debiendo quedar la información con el siguiente formato:

**\*\*Río Unica\*\***

Integrantes:

Moro Pitbull edad 0

Julio Cruza edad 13

Ramón Salchicha, alfa de la manada, edad 2

José Angora

Hernán Cruza

Fer Siames