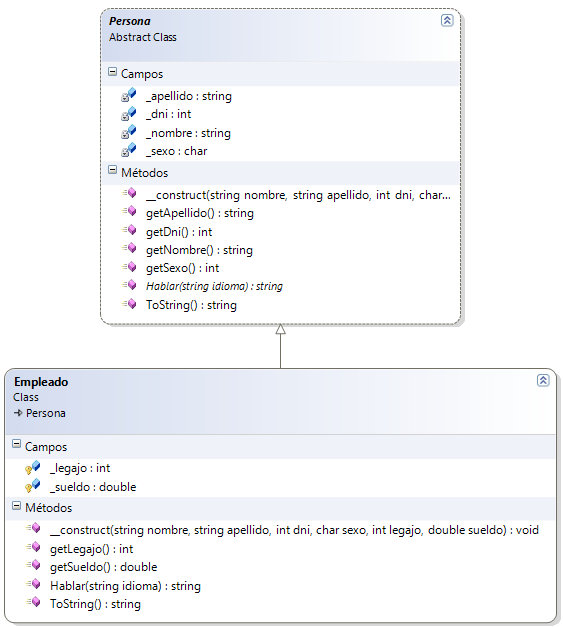
Trabajo Práctico (parte 1).- Programación III (PHP) - 2017

Profesores:

Neiner Maximiliano, Villegas Octavio

# 1.- Creación de jerarquía de clases

Codificar el siguiente diagrama de clases en PHP. Cada clase deberá ser codificada en un archivo distinto. *Ejemplo: persona.php, empleado.php, etc.*



Detalles técnicos

Clase Persona

Atributos (todos privados)

Constructor (inicializa los atributos de la clase)

Métodos (de instancia)

* **Hablar** (abstracto). Retorna un string.
* **ToString**. Retorna un string mostrando todos los datos de la persona, separados por un guión medio (-).
* **getters** para cada uno de los atributos.

Clase Empleado

Atributos (todos protegidos)

Constructor (inicializa los atributos de la clase)

Métodos (de instancia)

* **Hablar** (polimorfismo). Retorna un string con el formato “El empleado habla *Español*”, siendo Español, el valor recibido por parámetro.
* **ToString** (polimorfismo). Retorna un string mostrando todos los datos del empleado, separados por un guión medio (-).
* **getters** para cada uno de los atributos.

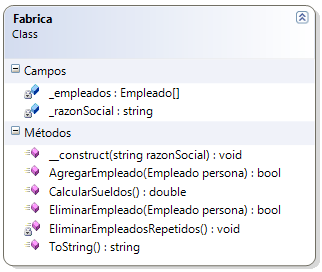
# 2.- Testear la jerarquía de clases

Para probar el buen funcionamiento de la jerarquía anteriormente descrita, en el archivo **index.php**, se deberá instanciar y probar todos los métodos.

***Nota:*** *Recordar incluir los archivos a ser utilizados desde index.php.*

# 3.- Modificar el diagrama de clases

Agregar al diagrama de clases la siguiente clase:



Detalles técnicos

Clase Fabrica

Atributos (todos privados)

Constructor (inicializa los atributos de la clase, incluido el array)

Métodos (de instancia)

* **AgregarEmpleado**. Recibe un objeto de tipo Empleado y lo agrega al final del array (buscar información de la función *array\_push()*)
* **CalcularSueldos**. Retornará el total acumulado que la fábrica abonará en concepto de sueldos.
* **EliminarEmpleado**. Recibe un objeto de tipo Empleado y lo elimina del array (buscar información de la función *unset()*)
* **EliminarEmpleadosRepetidos** (privado). Sacará del array aquellos elementos que se encuentren en más de una ocasión. Invocar al método cada vez que se agregue un empleado (buscar información de la función *array\_unique()*)
* **ToString**. Retorna un string mostrando todos los datos de la fábrica (incluidos sus empleados), separados por un guión medio (-).

# 4.- Testear la clase Fabrica

Testear el funcionamiento de la fábrica (en index.php), agregando y quitando empleados y mostrando sus características a partir de su método ToString().