

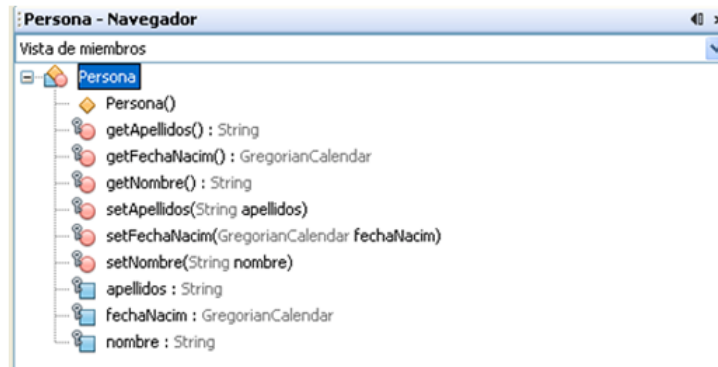
7.C. Herencia.

1. Herencia.

1.3. Utilización de miembros heredados (I). Atributos.

Los **atributos heredados** por una clase son, a efectos prácticos, iguales que aquellos que sean definidos específicamente en la nueva **clase derivada**.

En el ejemplo anterior la clase **Persona** disponía de tres atributos y la clase **Alumno**, que heredaba de ella, añadía dos atributos más. Desde un punto de vista funcional podrías considerar que la clase **Alumno** tiene cinco atributos: tres por ser **Persona** (**nombre**, **apellidos**, **fecha de nacimiento**) y otros dos más por ser **Alumno** (**grupo** y **nota media**).



Ejercicio resuelto

Dadas las clases **Alumno** y **Profesor** que has utilizado anteriormente, implementa métodos **get** y **set** en las clases **Alumno** y **Profesor** para trabajar con sus cinco atributos (tres heredados más dos específicos).

Solución:

Una posible solución sería:

1. Clase **Alumno**.

Se trata de heredar de la clase **Persona** y por tanto utilizar con normalidad sus atributos heredados como si pertenecieran a la propia clase (de hecho se puede considerar que le pertenecen, dado que los ha heredado).

```
public class Alumno extends Persona {  
  
    protected String grupo;  
  
    protected double notaMedia;  
  
  
    // Método getNombre  
    public String getNombre () {  
  
        return nombre;  
  
    }  
  
  
    // Método getApellidos  
    public String getApellidos () {  
  
        return apellidos;  
  
    }  
}
```

```
// Método getFechaNacim
```

```
public GregorianCalendar getFechaNacim (){
```

```
    return this.fechaNacim;
```

```
}
```

```
// Método getGrupo
```

```
public String getGrupo (){
```

```
    return grupo;
```

```
}
```

```
// Método getNotaMedia
```

```
public double getNotaMedia (){
```

```
    return notaMedia;
```

```
}
```

```
// Método setNombre
```

```
public void setNombre (String nombre){
```

```
    this.nombre= nombre;
```

```
}
```

```
// Método setApellidos
```

```
public void setApellidos (String apellidos){
```

```
    this.apellidos= apellidos;
```

```
}
```

```
// Método setFechaNacim
```

```
public void setFechaNacim (GregorianCalendar fechaNacim){
```

```
    this.fechaNacim= fechaNacim;
```

```
}
```

```
// Método setGrupo
```

```
public void setGrupo (String grupo){
```

```
    this.grupo= grupo;
```

```
}
```

```
// Método setNotaMedia
```

```
public void setNotaMedia (double notaMedia){
```

```
    this.notaMedia= notaMedia;
```

```
}
```

```
}
```

Si te fijas, puedes utilizar sin problema la referencia **this** a la propia clase con esos atributos heredados, pues pertenecen a la clase: **this.nombre**, **this.apellidos**, etc.

2. Clase **Profesor**.

Seguimos exactamente el mismo procedimiento que con la clase **Alumno**.

```
public class Profesor extends Profesor {
```

```
    String especialidad;
```

```
    double salario;
```

```
    // Método getNombre
```

```
    public String getNombre (){
```

```
        return nombre;
```

```
    }
```

```
    // Método getApellidos
```

```
    public String getApellidos (){
```

```
        return apellidos;
```

```
    }
```

```
    // Método getFechaNacim
```

```
    public GregorianCalendar getFechaNacim (){
```

```
        return this.fechaNacim;
```

```
    }
```

```
    // Método getEspecialidad
```

```
    public String getEspecialidad (){
```

```
        return especialidad;
```

```
    }
```

```
    // Método getSalario
```

```
    public double getSalario (){
```

```
        return salario;
```

```
    }
```

```
    // Método setNombre
```

```
    public void setNombre (String nombre){
```

```
        this.nombre= nombre;
```

```
    }
```

```
    // Método setApellidos
```

```

public void setApellidos (String apellidos){

    this.apellidos= apellidos;

}

// Método setFechaNacim

public void setFechaNacim (GregorianCalendar fechaNacim){

    this.fechaNacim= fechaNacim;

}

// Método setSalario

public void setSalario (double salario){

    this.salario= salario;

}

// Método setESpecialidad

public void setESpecialidad (String especialidad){

    this.especialidad= especialidad;

}

}

```

Una conclusión que puedes extraer de este código es que has tenido que escribir los métodos **get** y **set** para los tres atributos heredados, pero ¿no habría sido posible definir esos seis métodos en la clase base y así estas dos clases derivadas hubieran también heredado esos métodos? La respuesta es afirmativa y de hecho es como lo vas a hacer a partir de ahora. De esa manera te habrías evitado tener que escribir seis métodos en la clase **Alumno** y otros seis en la clase **Profesor**. Así que recuerda: **se pueden heredar tanto los atributos como los métodos**.

Aquí tienes un ejemplo de cómo podrías haber definido la clase **Persona** para que luego se hubieran podido heredar de ella sus métodos (y no sólo sus atributos):

```

public class Persona {

    protected String nombre;

    protected String apellidos;

    protected GregorianCalendar fechaNacim;

    // Método getNombre

    public String getNombre (){

        return nombre;

    }

    // Método getApellidos

    public String getApellidos (){

        return apellidos;

    }

    // Método getFechaNacim

```

public GregorianCalendar getFechaNacim (){
return this.fechaNacim;
}
// Método setNombre
public void setNombre (String nombre){
this.nombre= nombre;
}
// Método setApellidos
public void setApellidos (String apellidos){
this.apellidos= apellidos;
}
// Método setFechaNacim
public void setFechaNacim (GregorianCalendar fechaNacim){
this.fechaNacim= fechaNacim;
}
}