

HERENCIA DE CLASES

Como su nombre indica, una clase HIJA puede heredar propiedades y funciones de una clase PADRE.

Para probar esto vamos a crear 2 clases nuevas, una llamada Padre y otra llamada Hija.

```
public class Padre {  
  
    protected String nombre;  
    protected int numero;  
  
    public Padre(String nombre, int numero) {  
        this.nombre = nombre;  
        this.numero = numero;  
    }  
}
```

Como vemos, la clase Padre tiene sus propiedades encapsuladas como protected, recordamos que este encapsulamiento se utiliza cuando queremos propiedades privadas que se puedan heredar. Además tiene un constructor como cualquier otro.

```
public class Hijo{  
  
    private boolean bool;  
  
}
```

La clase Hijo de momento solo tiene una propiedad booleana y no está heredando aún.

Si añadimos el parámetro extends y el nombre de la clase padre estaremos haciendo que nuestra clase tenga que heredar todos los parámetros y funciones de la clase padre, en este caso:

```
public class Hijo extends Padre {  
  
}
```

```
7  
8  
9  
10  
11  
12  
13  
public class Hijo extends Padre{  
    protected boolean bool;  
}
```

El encapsulamiento de la propiedad hija la podemos cambiar si tenemos intención de heredar, si no es el caso no es necesario.

Como vemos, al heredar tenemos un error de clase, esto es porque al heredar una clase los hijos están obligados a utilizar los mismos métodos que el padre, en este caso un constructor.

```
public class Hijo extends Padre{  
  
    protected boolean bool;  
  
    public Hijo( String nombre, int numero, boolean bool) {  
        super(nombre, numero);  
        this.bool = bool;  
    }  
}
```