

EJERCICIOS POO PHP

1º) Probar la herencia.

Crea las clases:

Figura

Propiedades: nombre, color, esRellena

métodos: getNombre(), get/setColor(), estaRellena(), estaVacía(), showInfo()(muestra la información de la figura).

constructor: recibe el nombre de la figura y lo pone en nombre.

Circulo descendiente de Figura

propiedades: radio

métodos: get/setRadio, getArea()

Cuadrado descendiente de Figura

propiedades: lado

métodos: get/setLado getArea()

2º) Sobreescritura de métodos:

- En Circulo reescribir el método getNombre, para que nos devuelva "circulo".

3º) Métodos blindados (no se tocan):

- Uso de final.

Renombrar el método setNombre() de Figura para que no lo puedan redefinir en las clases hijas.

Comprobarlo.

4º) Ampliar la funcionalidad de los métodos:

Utilizando parent::, ampliar showInfo() para que muestre las propiedades de la clase hija, tanto en Circulo como en Cuadrado.

Crear un constructor para las clases hijas que reutilice el constructor del padre y lo amplie.

Probarlo.

5º) Métodos y clases abstractas.

Transformar Figura en una clase abstracta.

Las clases abstractas no pueden tener objetos.

Pueden tener métodos concretos y abstractos.

Todos los métodos abstractos de una clase abstracta hay que definirlos en las hijas.

Definir `getArea()` como método abstracto.

Crear la clase Rectángulo como hija de Figura, comprobar que me obliga a reescribir `getArea()`.

(De esta forma, no se nos puede pasar por alto definir los métodos que considero obligatorios para todas las figuras).

6º) Interfaces

Funcionan de manera similar a las clases Abstractas, pero solo definen métodos que las clases que implementan

estas interfaces deberán redefinir.

Para crearla se necesita la palabra clave: `interface`.

Ejemplo

```
interface MyInterface{  
  
    public function myMethod1($par1, $par2);  
  
    public function myMethod2($par1);  
  
}
```

Uso

```
class MyClass implements MyInterface {  
  
    ...  
  
}
```

Crea una interfaz que sea Imprimible con el método imprime(), que la implemente el método Cuadrado.

Cuando se imprime un cuadrado tendrás que comprobar si es relleno o no.

Si es relleno lo pintará completo y si no vacío.