

1. Crea una interfaz llamada `iembalaje` que contenga: a) el dato `margen`, que será el margen entre el objeto y el embalaje (típica gomaespuma alrededor de un producto al enviarlo en una caja). b) Función `embalar`, con parámetros de entradas "unidades". Todos los productos que serán embalados deben crear una variable `volumenEnvoltorio` en tiempo real que sea el producto de "largo x ancho x alto". Para ello calcular: i. El largo: es igual al largo del producto + el margen x 2. ii. El alto: es igual al alto del producto + el margen x 2 iii. El ancho: (Ancho del producto x unidades) + margen x 2.
2. Crear la clase "cliente": a. Variables accesibles desde cliente: i. Nombre ii. Producto b. Variables accesibles por todos: i. Apodo c. Al crear un cliente deben inicializarse todas las variables del punto a, e incrementar un contador de "clientela". El producto por defecto es " " (vacío). Debe mostrarse el mensaje "<nombre> creado". d. Al eliminar un cliente debe actualizarse la "clientela". e. Debe haber una función "comprarProd" que actualice el producto del cliente. f. Debe haber una función "getProducto()" que lo retorne. g. Debe haber una función mostrar, que imprima por pantalla: "<nombre> es conocido por <apodo>".
3. Crear la clase "papel": a. Deberás guardar las variables. i. "páginasGastadas" y "páginasRecicladas". Siendo su valor 0 de inicio. ii. Variable privada `dobleCara`. iii. Variables "alto" y "largo", las cuales NO pueden ser públicas. b. En el constructor deben inicializarse largo, alto y `dobleCara`. c. Debe haber un "método `calcularEspacio()`" no definido, que se escribirá en las subclases. d. Debe haber un método "getDobleCara()", que retorne el valor de esa variable. e. Debe haber un `__toString()` que retorne "Se usa un papel de tamaño (alto x largo): <alto> x <largo>".
4. Crear la clase "fotocopia" que hereda de "papel". Al crear una fotocopia: i. Se inicializan, largo, alto y doble cara (por defecto false). ii. Incrementa las "páginasGastadas" en 1 unidad. c. Al destruir la fotocopia: i. Incrementa las "páginasRecicladas" en 1 unidad. d. Función `__toString()` que muestre la información heredada del `toString` de superclase y también, si la fotocopia es a doble cara o no.
5. Crear la clase "libro", que hereda de papel e implementa la interfaz. a. Variable accesible por todo el programa: título. b. Variable no accesibles: i. Páginas (número total de ellas que tiene el libro) ii. Ancho (la anchura es: `numeroPaginas/100`) c. Datos fijos "dobleCara", que siempre será true. d. Al crear un libro indicar (largo, alto y npáginas) i. Inicializar: largo, ancho, alto, páginas y doblecara) ii. Incrementar el número de páginas gastadas. e. Al eliminar el libro, reciclar su número de páginas. f. Función `__toString()`, que muestre "Libro de <numPag> páginas titulado <título>".

6. Crear un index.php que sirva como programa principal para trabajar con los objetos:

Crear un cliente con nombre "pepe" que no tenga ningún producto asignado.

Crear una fotocopia a doble cara, de tamaño 10 x 20.

Muestra su superficie (calcularEspacio).

Haz que pepe compre la fotocopia.

Muestra los datos de la fotocopia.

Muestra las páginas gastadas.

Muestra las páginas recicladas.

Muestra la clientela total.

Elimina el cliente.

Elimina la fotocopia.

Muestra las páginas gastadas.

Muestra las páginas recicladas.

Muestra la clientela total.

Crear un libro de medidas 12 x 25 y 200 páginas.

Ponle un título sin utilizar funciones.

Embala 2 unidades.

Muestra su "volumenEnvoltorio".

Crear un cliente llamado Juan que tenga como producto el libro creado.

Muestra las páginas gastadas.

Muestra las páginas recicladas.

Muestra la clientela total.