Argumentación

# Ejercicio facturas

El diagrama de clases se ha planteado para una empresa pequeña, que puede ser una tienda cualquiera donde solo haya una o dos personas trabajando y la administración la lleven ellos mismos. Por lo tanto los campos que corresponden a la empresa que vende el producto siempre van a ser los mismos y por ello no se ha creado la clase Empresa donde aparezcan sus atributos propios como podría ser “sucursal”, “dirección”, etc.

En la clase factura se especifica un atributo boolean efectivo. Si este boolean es verdadero el pago se ha realizado en efectivo si el boolean es falso el pago se ha efectuado a través de una entidad bancaria.

Nuestra empresa solo le vende a particulares, no está habilitada para venderle a otras empresas por lo que los datos del cliente son los que se les pediría a un cliente normal.

En la clase Producto aparecerá un método abstracto, “tipoIVA”. Cada producto tiene un tipo de IVA diferente se marcara como predeterminado el 21% pero si el producto en cuestión pertenece a una categoría que le corresponde un IVA menos como por ejemplo puede ser alimentos o libros se implementara métodos distintos para ellos. Aunque en el diagrama no haya creado los distintos tipos de productos entendemos que estos heredan de la clase productos todos sus atributos y métodos.

Cuando en el diagrama se pone como método “gets/sets” nos referimos así a todos los métodos que se generarían de los atributos de cada clase.

# Ejercicio mochila

En la clase mochila se considera que esta tendrá un peso mínimo, por lo que el método peso es un campo calculado entre ese peso mínimo y el de los objetos que lleva. Además el espacio se distribuye en slot.

Hay dos posibilidades de que la mochila no pueda usarse para meter más objetos la primera que el espacio se agote o que llegue al peso máximo. Consideramos que el método “DestruirItem” se da al arrojar un objeto fuera de la mochila que no es lo mismo que equipárselo al personaje.

El método “DañoTotal” de un arma a distancia es la suma del daño que hace ese arma de por sí o por la mejoras que pueda tener a parte del que haga la munición que se use junto a él. También hemos tenido en cuenta que se pueda romper por ejemplo una flecha al ser lanzada.

El método “ImpactoCritico” es calculado con el atributo de municiones y con el propio atributo del arma. A parte tenemos que el arco puede realizar tiros especiales por el hecho de ser un arco por lo que existe un atributo “tipoTiro”.

El uso de un arma cuerpo a cuerpo puede ser con una o dos manos, para ello se ha implementado el método “UsarDosManos”, tipo boolean. Si es falso usaría solo una mano si es verdadero usa las dos para sostener el arma.

Cuando en el diagrama se pone como método “gets/sets” nos referimos así a todos los métodos que se generarían de los atributos de cada clase.