

7.- Se debe crear a clase *VentaBuilder* de xeito similar a *ClienteBuilder*. En principio o chequeo dos campos sempre será exitoso, pero nun futuro se comprobará a existencia do cliente do artigo e que se dispón de stock suficiente. Se debe verificar que a data sexa anterior ou igual a hoxe, que as unidades son >0 e que o prezo é maior a 0.

8.- No desenvolvemento de *VentaBuilder* podemos observar que pode utilizarse código común con *ClienteBuilder*. **Se pide desenvolver a clase *BuilderAdapter*** que permita non duplicar código.

9.- Desenvolver a clase *Factura* que teña os seguintes atributos so accesibles dende o propio package: ***String codigo***, ***LocalDate dataFactura***, ***String dniCliente***, ***Collection<Venta> ventas***, ***double importe***, ***double iva***, ***double total***, ***boolean pechada***.

- O construtor de *Factura* recibe a data da factura e o dni do cliente.
- O valor do atributo ***codigo*** da factura inicialmente é ***null*** e o do atributo ***pechada*** é ***false***
- Non será posible construír obxectos *Factura* dende fora do package
- Os atributos *dataFactura*, *dniCliente* e *ventas* non deben poder modificarse unha vez establecido o seu valor (o contido da colección *ventas*, si)
- Debes programar un método accesible dende calquera parte ***addVenta(Venta v)*** que engade unha venda ao atributo *ventas*. Si a venda non corresponde co cliente da factura ou non se pode engadir a venda se debe lanzar unha *VerboseException* coa mensaxe de erro apropiada.
- Debe dispoñer dos getter e setter axeitados.
- A representación como *String* dunha *Factura* será ***dataFactura codigo dniCliente total€***