Ejercicio 1:

Explicacion de los principios de diseño: SOLID:

--> Principio de Sustitución de Liskov

Se cumple este principio ya que en la clase ElementosBillete que es nuestra clase abstracta tenemos nuestros metodos abstractos, y además tenemos metodos no abstractos en la clase que no van a ser sobreescritos en ninguna de las clases que heredan, como por ejemplo los metodos "mostrar_billete", "mostrar totales" y "mostrar disponibles".

--> Principio de inversión de la dependencia Está presente ya que dependemos de una clase ElementosBillete que es abstracta en vez de depender de clases o funciones concretas.

Explicación del patron usado:

--> Patron Composición

Hemos utilizado el patron composicion ya que observamos que era el que mas se adecuaria a este ejercicio. Esto se debe a que en este ejercicio de gestion de billetes, cada billete va tener una serie de criterios para la posterior seleccion de los billetes, lo que es decir, cada billete va a contener esos criterios. Además se nos pide que se puedan introducir nuevos criterios en un futuro, y el patron composicion facilita la introduccion de nuevos componentes.

Diagramas UML:



