

## naive-ticket-machine

En este proyecto, realiza las siguientes modificaciones:

1. Escribe un método selector llamado `getTotal`. Este método debe devolver el valor del campo **total**.
2. Crea un método llamado `setPrice(int cost)` que le asigne el valor **cost** al campo **price**.
3. Añade un método denominado `prompt`. Este método debe tener un tipo de retorno **void** y ningún parámetro. El cuerpo del método debe imprimir la siguiente línea  
`Please insert the correct amount of money.`  
para informar al cliente de que introduzca la cantidad correcta de dinero.
4. Añade un método `showPrice`. Debe tener un tipo de retorno **void** y ningún parámetro. El cuerpo del método debe imprimir:  
`The price of the ticket is xyz cents.`  
para informar al cliente de cuál es el precio del **billete**. `xyz` debe sustituirse por el valor del campo **price** en el momento de invocar al método.
5. Define un nuevo constructor para la clase. Este no debe admitir ningún parámetro, por lo que se deberá fijar el precio posteriormente llamando a `setPrice`. Comprueba la implementación creando máquinas con ambos constructores.
6. Crea un nuevo método, `empty`, que simule el efecto de extraer todo el dinero acumulado en la máquina. Este método debe tener un tipo de retorno **void** y su cuerpo ha de asignar simplemente el valor **0** al campo **total**. ¿Necesita este método algún tipo de parámetro? Comprueba este método creando una máquina, introduciendo algo de dinero, imprimiendo billetes, comprobando el total y luego vaciando la máquina. ¿Es el método `empty` un mutador o un selector?.