



En la realización de esta práctica, dado que el trabajo a realizar era más extenso que en las anteriores, se han tomado diversas decisiones (siempre siguiendo lo descrito en el enunciado de la práctica) a la hora de implementar la biblioteca.

El apartado 1 nos llevó más tiempo del pensado en realizarlo. Tras leer detenidamente el enunciado varias veces, fuimos implementando poco a poco todas las clases necesarias y los métodos especificados. El enunciado especifica que hay algunos atributos que no deben ser modificados, como los atributos comunes a las obras o el nombre de los usuarios de la biblioteca, y pensamos en que debíamos declarar esos atributos como `static`, pero luego decidimos que era más simple no implementar los *setters* de esos atributos, de forma que no los modificaríamos en ningún caso. Además, para ejecutar el primer tester y comprobar si todo funcionaba como debía, debíamos definir clases como Ejemplar y Prestamo, aunque de los mismos sólo teníamos, en ese momento, que implementar muy pocos aspectos. Tuvimos problemas con los números referentes a ppb y ps de la clase Publico pues no actuaban como debían, aunque tras repasar el código pudimos solucionarlo sin mayores problemas.

En el apartado 2, no teníamos claro cómo podíamos modificar una función declarada e implementada en una librería, como lo era el método *equals*, pero tras investigar un poco comprobamos que, implementando nosotros la función *equals* con los mismos parámetros de entrada y salida que el método de la librería, al llamar a ese método, se utilizaría nuestra implementación.

En el apartado 3, para asignarle a cada ejemplar un identificador, utilizamos una variable de tipo `static int` declarada como variable de la clase Ejemplar que, al crearse un nuevo ejemplar, aumentara en uno el valor de esta variable, para así, de forma automática, al crearse un ejemplar, tuviera un identificador único. Nos resultó algo costoso el implementar correctamente los métodos de prestar y devolver para que hiciera lo que debía hacer, pues a veces no se actualizaban correctamente los valores de ppb de los objetos de la clase Publico. Para llevar la cuenta del número de préstamos realizados y del número de préstamos pendientes, declaramos en la clase Prestamos dos variables de tipo `static int`, prestados y devueltos, ambas inicializadas a 0, donde prestados aumentaba (en uno) cada vez que se prestaba un ejemplar y devueltos aumentaba (en uno) cada vez que se devolvía un ejemplar, para así poder calcular tanto el número de préstamos realizados (prestados) como el número de préstamos pendientes (prestados-devueltos).

En el apartado 4, el último realizado (pues el 5 era realizar el diagrama de clases, la documentación javadoc y guardar una imagen de la salida de los testers), utilizamos la información dada en el enunciado para trabajar con las fechas y poder calcular los días de retraso en las devoluciones. El valor de los días de retraso podía ser positivo, 0 o negativo. En el caso de que fuera negativo, significaba que aun no había vencido, por lo que no aplicamos ninguna sanción. El método que aplicaba las sanciones a los usuarios recibía como parámetro de entrada una variable de tipo `int`, por lo que hicimos un casting.

En lo que respecta a los problemas pendientes por resolver, sólo hay uno en el apartado 3. El método `getPrestamos()` en la clase Usuario, dado que su valor de retorno es un `ArrayList` de objetos de la clase Prestamo, debíamos implementar dicho método para que no lo devolviera tal cual, pero no hemos sido capaces de entender del todo qué tipo de dato tiene que devolver, pues el enunciado dice *Este método debe devolver algo así como Collections.unmodifiableList(misPrestamos).*