

1. Dado el siguiente código que recorre un conjunto de elementos y los añade a una lista, añade el código correspondiente a continuación que permita eliminar los elementos del conjunto que empiezan por “Barcelona”:

```
for (String entrada : conjuntoJugadores) {  
    jugador = entrada.substring(entrada.indexOf(" - ") + 3).trim();  
    listaJugadores.add(jugador);  
}
```

2. Tomando como referencia el ejercicio 1, añade el código correspondiente para poder pasar recorrer todos los elementos del conjunto y pasar el contenido de cada uno de ellos a mayúsculas.

3. Dados dos conjuntos:

```
conjuntoNumeros1 = [3, 5, 7, 9, 11, 13, 15]
```

```
conjuntoNumeros2 = [5, 7, 10, 12, 15, 17]
```

- a) Declara **conjuntoResultado** de números enteros de manera que contenga los elementos de **conjuntoNumeros1** ordenados de menor a mayor.

-
- b) Indica el contenido final que contendrá **conjuntoResultado** después de que se realicen las siguientes operaciones:

```
conjuntoResultado.retainAll(conjuntoNumeros2);
```

```
conjuntoResultado.addAll(conjuntoNumeros2);
```

```
conjuntoResultado.remove(15);
```

4. Considerando el siguiente fragmento de código, indica el orden en el que se deben ejecutar las líneas para que se pueda leer una línea del fichero.

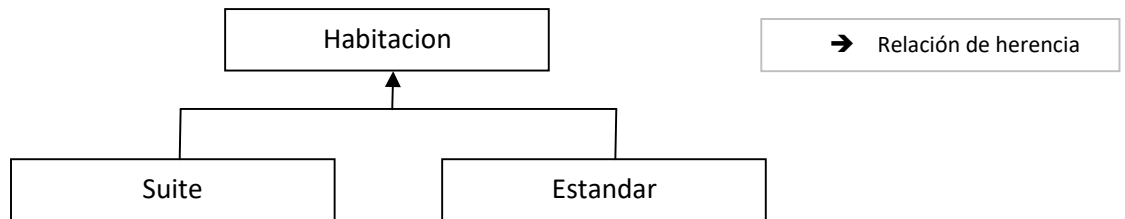
```

1. String linea = br.readLine();
2. FileReader fr = new FileReader(archivo);
3. File archivo = new File(System.getProperty("user.dir") + "/recursos/fichero.txt");
4. BufferedReader br = new BufferedReader(fr);

```

Orden de ejecución correcto: _____

5. Una cadena hotelera quiere modelar su sistema de reservas. Para ello, su departamento de informática crea la siguiente jerarquía de clases:



Descripción de atributos de Habitación:

Atributo	Tipo	Descripción
CADENA_HOTELERA	texto	Nombre de la cadena hotelera (Hotel Aguadulce). Nunca cambia y es compartido por todas las habitaciones
MAX_OCUPANTES	entero	Número máximo de ocupantes permitido (4). Nunca cambia y es compartido por todas
totalHabitaciones	entero	Contador de habitaciones registradas. Compartido por todas pero puede cambiar
idHabitacion	entero	Identificador único de la habitación. Exclusivo de cada objeto y nunca cambia tras su asignación
precioNoche	real	Precio por noche. Exclusivo de cada objeto y puede modificarse
planta	entero	Planta donde se ubica. Accesible desde subclases y puede cambiar
categoria	texto	Categoría de la habitación. Accesible desde subclases pero no puede cambiar una vez asignada

a) Declara todos los atributos de la clase **Habitacion** con los modificadores correctos.

b) Indica qué atributos de **Habitacion** son accesibles desde Suite sin usar getters y por qué.

c) ¿Qué ocurriría si intentamos modificar **CADENA_HOTELERA** desde algún método? Explícalo.