

Andres Serrato

Zayra Gutierrez

Sebastian Cardona

Felipe Martinez

Diego Cardenas

```
public class InvalidCodeException extends Exception{  
    public InvalidCodeException(String message) {  
        super(message);  
    }  
}
```

```

1  import java.util.Date;
2  public class Lector {
3      protected String nombre;
4      protected String direccion;
5      protected String telefono;
6      protected int codigo;
7      public Lector(String nombre, String direccion, String telefono, int codigo) {
8          this.nombre = nombre;
9          this.direccion = direccion;
10         this.telefono = telefono;
11         this.codigo = codigo;
12     }
13     public void reservar(Date date){
14         try {
15             //logica de reservar
16             //verifica si el codigo esta bien:
17             isValidCode(codigo);
18         } catch (InvalidCodeException e) {
19             //logica de enviar mensaje de error
20         }
21         //más logica
22     }
23     public void entregar(){}
24     public boolean isValidCode(int code) throws InvalidCodeException{
25         boolean isValid = false;
26         if(1000000000 > code){
27             return isValid= true;
28         }
29         throw new InvalidCodeException("El código no es válido");
30     }
31 }

```

```


1  public class Libro {
2  }
3

```

```

public class NotAveableMaterialException extends Exception{
    public NotAveableMaterialException(String message) {
        super(message);
    }
}

```

```
2 public class NotMaterialFoundException extends Exception {
3 >  public NotMaterialFoundException(String message) { super(message); }
6 }
7
```

```
1 import java.util.Date;
2
3 public class Pedido {
4     private String idUsuario;
5      private String tituloMaterial;
6     private int codigoMaterial;
7
8     public void reservar(Date fecha){
9         //logica de reservar
10        try {
11            RecursoBiblioteca recurso = new RecursoBiblioteca();
12            recurso.reservar(fecha);
13        } catch (NotMaterialFoundException e) {
14            //logica de enviar mensaje de error
15        } catch (NotAveableMaterialExeception e){
16            //logica de enviar mensaje de error
17        }
18        // logica
19    }
20
21    public void entregar(Date fecha){
22
23    }
24
25 }
26
```

```
1 import java.util.Date;
2
3 public class RecursoBiblioteca {
4     protected String titulo;
5     protected String tipo;
6     protected String autor;
7     protected String editorial;
8
9     public void reservar(Date fecha) throws NotMaterialFoundException, NotAveableMaterialExeption{
10         //logica de reservar
11         //mirar si titulo no está en recursos
12         if (/* condición para verificar si el titulo no está en los recursos */true) {
13             throw new NotMaterialFoundException("No se ha encontrado el material");
14         }
15         //mirar si el material está disponible
16         else if (/* condición para verificar si el material no está disponible */true) {
17             throw new NotAveableMaterialExeption("El material no está disponible");
18         }
19     }
20     public void entregar(Date fecha){
21
22     }
23 }
```

