

Trabajo Práctico 6: Colecciones y Sistema de Stock

1) Ejercicio1_TP6 (Main):

```
package SistemaStock;

import java.util.ArrayList;

public class Main {

    public static void main(String[] args) {
        Inventario inv = new Inventario();

        inv.agregarProducto(new Producto("001", "Monitor 140HZ", 4500000, 8, CategoriaProducto.ELECTRONICA));
        inv.agregarProducto(new Producto("002", "Masa de Empanadas x2", 2800, 20, CategoriaProducto.ALIMENTOS));
        inv.agregarProducto(new Producto("003", "Esmoquin Negro", 655000, 6, CategoriaProducto.ROPA));
        inv.agregarProducto(new Producto("004", "Tallarines", 1900, 15, CategoriaProducto.ALIMENTOS));
        inv.agregarProducto(new Producto("005", "Escritorio para PC", 550000, 2, CategoriaProducto.HOGAR));

        inv.listarProductos();

        System.out.println("\nProducto id 001: ");
        Producto productoPorId = inv.buscarProductoPorId("001");
        productoPorId.mostrarInfo();

        System.out.println("\nProductos filtrados");
        ArrayList<Producto> productosPorCategoria = inv.filtrarPorCategoria(CategoriaProducto.ROPA);
        for(Producto p : productosPorCategoria) {
            p.mostrarInfo();
        }

        System.out.println("\nEliminado producto id 005");
        inv.eliminarProducto("005");
        System.out.println("\nInventario actualizado: ");
        inv.listarProductos();

        inv.actualizarStock("003", 10);

        System.out.println("\nStock disponible: " + inv.obtenerTotalStock());

        System.out.println("\nProducto con mayor Stock: ");
        inv.productoConMayorStock().mostrarInfo();
    }
}
```

```
        System.out.println("\nProductos entre $1000 y $3000");
        ArrayList<Producto> productosFiltradosPorPrecio = inv.filtrarProductosPorPrecio(1000, 3000);
        for(Producto p: productosFiltradosPorPrecio){
            p.mostrarInfo();
        }

        System.out.println("\nCategorias disponibles: ");
        inv.mostrarCategoriasDisponibles();
    }
}
```

Producto:

```

public double getPrecio() {
    return precio;
}

public void setPrecio(double precio) {
    this.precio = precio;
}

public int getCantidad() {
    return cantidad;
}

public void setCantidad(int cantidad) {
    this.cantidad = cantidad;
}

public CategoriaProducto getCategoria() {
    return categoria;
}

public void setCategoria(CategoriaProducto categoria) {
    this.categoria = categoria;
}

public void mostrarInfo() {
    System.out.println("Producto{" + "id=" + id + ", nombre=" + nombre + ", precio=" + precio + ", cantidad=" + cantidad + ", categoria=" + categoria + '}');
}
}

```

CategoriaProducto:

```

package SistemaStock;

public enum CategoriaProducto {

    ALIMENTOS("Productos Comestibles"),
    ELECTRONICA("Dispositivos Electronicos"),
    ROPA("Ropa Vestir"),
    HOGAR("Articulos Hogar");
    private final String descripcion;
    CategoriaProducto(String descripcion) {
        this.descripcion = descripcion;
    }
    public String getDescripcion() {
        return descripcion;
    }
}

```

Inventario:

```
package SistemaStock;

import java.util.ArrayList;

public class Inventario {

    private ArrayList<Producto> productos;

    public Inventario() {
        this.productos = new ArrayList();
    }

    public void agregarProducto(Producto p) {
        productos.add(p);
    }

    public void listarProductos() {
        for (Producto p : productos) {
            p.mostrarInfo();
        }
    }

    public Producto buscarProductoPorId(String id) {
        Producto productoEncontrado = null;
        int i = 0;

        while (i < productos.size() && !this.productos.get(i).getId().equalsIgnoreCase(id)) {
            i++;
        }

        if (i < this.productos.size()) {
            productoEncontrado = this.productos.get(i);
        }

        return productoEncontrado;
    }

    public void eliminarProducto(String id) {
        Producto p = buscarProductoPorId(id);

        if (p != null) {
            this.productos.remove(p);
            System.out.println("El producto se ha eliminado");
        } else {
            System.out.println("El ID no se encontro");
        }
    }
}
```

```
public void actualizarStock(String id, int nuevaCantidad) {
    Producto p = buscarProductoPorId(id);

    if (p != null && nuevaCantidad > 0) {
        p.setCantidad(nuevaCantidad);
        System.out.println("Stock actualizado");
    } else if (nuevaCantidad < 0) {
        System.out.println("Debe de ingresar una cantidad positiva");
    } else if (p == null) {
        System.out.println("El ID no se encontro");
    }
}

public ArrayList<Producto> filtrarPorCategoria(CategoriaProducto categoria) {
    ArrayList<Producto> productosFiltrados = new ArrayList();

    for (Producto p : productos) {
        if (p.getCategoria() == categoria) {
            productosFiltrados.add(p);
        }
    }

    return productosFiltrados;
}

public int obtenerTotalStock() {
    int totalStock = 0;
    for (Producto p : productos) {
        totalStock += p.getCantidad();
    }

    return totalStock;
}
```

```
public Producto productoConMayorStock() {
    int maxStock = -1;
    Producto productoConMayorStock = null;
    for (Producto p : productos) {
        if (p.getCantidad() >= maxStock) {
            maxStock = p.getCantidad();
            productoConMayorStock = p;
        }
    }

    return productoConMayorStock;
}

public ArrayList<Producto> filtrarProductosPorPrecio(double min, double max) {
    ArrayList<Producto> productosFiltrados = new ArrayList();

    for (Producto p : productos) {
        if (p.getPrecio() >= min && p.getPrecio() <= max) {
            productosFiltrados.add(p);
        }
    }

    return productosFiltrados;
}

public void mostrarCategoriasDisponibles() {
    System.out.println("Categorias disponibles: ");
    for (CategoriaProducto c : CategoriaProducto.values()) {
        System.out.println("Categoria: " + c);
        System.out.println("Descripcion: " + c.getDescripcion());
    }
}

}
```

Build:

```
--- exec:3.1.0:exec (default-cli) @ Ejercicio1 ---
Producto{id=001, nombre=Monitor 140HZ, precio=4500000.0, cantidad=8, categoria=ELECTRONICA}
Producto{id=002, nombre=Masa de Empanadas x2, precio=2800.0, cantidad=20, categoria=ALIMENTOS}
Producto{id=003, nombre=Esmoquin Negro, precio=655000.0, cantidad=6, categoria=ROPA}
Producto{id=004, nombre=Tallarines, precio=1900.0, cantidad=15, categoria=ALIMENTOS}
Producto{id=005, nombre=Escritorio para PC, precio=550000.0, cantidad=2, categoria=HOGAR}

Producto id 001:
Producto{id=001, nombre=Monitor 140HZ, precio=4500000.0, cantidad=8, categoria=ELECTRONICA}

Productos filtrados
Producto{id=003, nombre=Esmoquin Negro, precio=655000.0, cantidad=6, categoria=ROPA}

Eliminado producto id 005
El producto se ha eliminado

Inventario actualizado:
Producto{id=001, nombre=Monitor 140HZ, precio=4500000.0, cantidad=8, categoria=ELECTRONICA}
Producto{id=002, nombre=Masa de Empanadas x2, precio=2800.0, cantidad=20, categoria=ALIMENTOS}
Producto{id=003, nombre=Esmoquin Negro, precio=655000.0, cantidad=6, categoria=ROPA}
Producto{id=004, nombre=Tallarines, precio=1900.0, cantidad=15, categoria=ALIMENTOS}
Stock actualizado

Stock disponible: 53

Producto con mayor Stock:
Producto{id=002, nombre=Masa de Empanadas x2, precio=2800.0, cantidad=20, categoria=ALIMENTOS}

Productos entre $1000 y $3000
Producto{id=002, nombre=Masa de Empanadas x2, precio=2800.0, cantidad=20, categoria=ALIMENTOS}
Producto{id=004, nombre=Tallarines, precio=1900.0, cantidad=15, categoria=ALIMENTOS}

Categorias disponibles:
Categorias disponibles:
Categoria: ALIMENTOS
Descripcion: Productos Comestibles
Categoria: ELECTRONICA
Descripcion: Dispositivos Electronicos
Categoria: ROPA
Descripcion: Ropa Vestir
Categoria: HOGAR
Descripcion: Articulos Hogar
-----
```

2) Ejercicio2_TP6 (Main):

```

Ejercicio2_TP6.java x Autor.java x Biblioteca.java x Libro.java x
Source History
1 package ejercicio_2;
2
3 import java.util.List;
4
5 public class Ejercicio2_TP6 {
6
7     public static void main(String[] args) {
8         Biblioteca biblioteca=new Biblioteca("Biblioteca");
9
10        Autor alberto=new Autor("1","Alberto Gonzalez","Argentina");
11        Autor david= new Autor("2","David Perez","Chile");
12        Autor rosario= new Autor("3","Rosario Fernandez","España");
13
14        biblioteca.agregarLibro("Z678","Biologia Basica", 2017, alberto);
15        biblioteca.agregarLibro("I987", "Hardware de un PC", 2015, david);
16        biblioteca.agregarLibro("J321", "Biologia Intermedia", 2018, alberto );
17        biblioteca.agregarLibro("P667", "Lenguaje C++", 2019, rosario );
18        biblioteca.agregarLibro("V555", "Software de un PC", 2014, david);
19
20        System.out.println("Libros de la "+biblioteca.getNombre()+" : ");
21        biblioteca.listarLibros();
22        System.out.println("-----");
23
24        System.out.println("Se busca un libro por isbn Z678");
25        biblioteca.buscarLibroPorIsbn("Z678").mostrarInfo();
26        System.out.println(".....");
27    }
28 }

```

```

Ejercicio2_TP6.java x Autor.java x Biblioteca.java x Libro.java x
Source History
28     int anioFiltro=2021;
29     List<Libro> librosPorAnio=biblioteca.filtrarLibrosPorAnio(anioFiltro);
30
31     System.out.println("Libros encontrados del año "+anioFiltro+" : ");
32     for(Libro libro:librosPorAnio){
33         libro.mostrarInfo();
34     }
35     System.out.println("////////////////////////////////////////");
36
37
38     biblioteca.eliminarLibro("J321");
39
40     System.out.println("Se imprimen los libros luego de eliminar uno");
41     biblioteca.listarLibros();
42     System.out.println("Cantidad total de libros en la biblioteca " +
43         biblioteca.getNombre()+" : "+biblioteca.obtenerCantidadLibros());
44
45     System.out.println("Autores de los libros: ");
46     biblioteca.mostrarAutoresDisponibles();
47 }
48
49 }
50

```

Autor:



```
1 package ejercicio_2;
2
3 public class Autor {
4     private String id;
5     private String nombre;
6     private String nacionalidad;
7
8     public Autor(String id, String nombre, String nacionalidad) {
9         this.id = id;
10        this.nombre = nombre;
11        this.nacionalidad = nacionalidad;
12    }
13
14    public String getId() {
15        return id;
16    }
17
18    public void setId(String id) {
19        this.id = id;
20    }
21
22    public String getNombre() {
23        return nombre;
24    }
25
26    public void setNombre(String nombre) {
27        this.nombre = nombre;
28    }
29
30    public String getNacionalidad() {
31        return nacionalidad;
32    }
33
34    public void setNacionalidad(String nacionalidad) {
35        this.nacionalidad = nacionalidad;
36    }
37
38    public String mostrarInfo() {
39        return "Autor{id: "+id+
40            "\nNombre: "+nombre+
41            "\nNacionalidad: "+nacionalidad+
42            "}";
43    }
44 }
45
```


Biblioteca:



```
1 package ejercicio_2;
2
3 import java.util.ArrayList;
4 import java.util.Collections;
5 import java.util.List;
6
7 public class Biblioteca {
8
9     private String nombre;
10    private List<Libro> libros;
11
12    public Biblioteca(String nombre) {
13        this.nombre = nombre;
14        this.libros = new ArrayList<>();
15    }
16
17    public List<Libro> getLibros() {
18        return Collections.unmodifiableList(libros);
19    }
20
21    public void agregarLibro(String isbn, String titulo,
22                             int anioPublicacion, Autor autor) {
23        if (!isbn.equals("") && isbn != null && !titulo.equals("")
24            && titulo != null && anioPublicacion > 0 && autor != null) {
25            libros.add(new Libro(isbn, titulo, anioPublicacion, autor));
26        }
27    }
28
29    public void listarLibros() {
30        System.out.println("Lista de libros de la biblioteca: " + nombre);
31        for (Libro libro : libros) {
32            libro.mostrarInfo();
33        }
34    }
35
36    public Libro buscarLibroPorIsbn(String isbn) {
37        for (Libro libro : libros) {
38            if (libro.getIsbn().equals(isbn)) {
39                return libro;
40            }
41        }
42        return null;
43    }
44 }
```



```
Ejercicio2_TP6.java x Autor.java x Biblioteca.java x Libro.java x
Source History
46
47 public void eliminarLibro(String isbn) {
48     if (isbn != null) {
49         Libro libroEliminar = this.buscarLibroPorIsbn(isbn);
50         if (libroEliminar != null) {
51             libros.remove(libroEliminar);
52         }
53     }
54 }
55
56
57 public int obtenerCantidadLibros() {
58     return libros.size();
59 }
60
61 public void mostrarAutoresDisponibles() {
62     for(Libro libro: libros){
63         System.out.println(libro.getAutor().mostrarInfo());
64     }
65 }
66
67 public List<Libro> filtrarLibrosPorAnio(int anio){
68     List<Libro> libroPorAnio=new ArrayList();
69     for(Libro libro:libros){
70         if(libro.getAnioPublicacion()==anio){
71             libroPorAnio.add(libro);
72         }
73     }
74     return Collections.unmodifiableList(libroPorAnio);
75 }
76
77 public String getNombre() {
78     return nombre;
79 }
80
81 public void setNombre(String nombre) {
82     this.nombre = nombre;
83 }
84
85 }
```

Libro:



```
1 package ejercicio_2;
2
3 public class Libro {
4     private String isbn;
5     private String titulo;
6     private int anioPublicacion;
7     private Autor autor;
8
9     public Libro(String isbn, String titulo, int anioPublicacion, Autor autor) {
10         this.isbn = isbn;
11         this.titulo = titulo;
12         this.anioPublicacion = anioPublicacion;
13         this.autor = autor;
14     }
15
16     public String getIsbn() {
17         return isbn;
18     }
19
20     public void setIsbn(String isbn) {
21         this.isbn = isbn;
22     }
23
24     public String getTitulo() {
25         return titulo;
26     }
27
28     public void setTitulo(String titulo) {
29         this.titulo = titulo;
30     }
31
32     public int getAnioPublicacion() {
33         return anioPublicacion;
34     }
35
36     public void setAnioPublicacion(int anioPublicacion) {
37         this.anioPublicacion = anioPublicacion;
38     }
39
40     public Autor getAutor() {
41         return autor;
42     }
43 }
```

```

public void mostrarInfo() {
    System.out.println("Libro {isbn: "+isbn+
        "\nTitulo: "+titulo+
        "\n Año de Publicacion: "+anioPublicacion+
        "\n "+autor.mostrarInfo()+
        "}");
}
}

```

Build:

```

Output - Ejercicio2TP6 (run) x
run:
Libros de la Biblioteca Universitaria:
Lista de libros de la biblioteca: Biblioteca Universitaria
Libro {isbn: Z678
Titulo: Biologia Basica
Año de Publicacion: 2017
Autor{id: 1
Nombre: Alberto Gonzalez
Nacionalidad: Argentina}}
Libro {isbn: I987
Titulo: Hardware de un PC
Año de Publicacion: 2015
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}}
Libro {isbn: J321
Titulo: Biologia Intermedia
Año de Publicacion: 2018
Autor{id: 1
Nombre: Alberto Gonzalez
Nacionalidad: Argentina}}
Libro {isbn: P667
Titulo: Lenguaje C++
Año de Publicacion: 2019
Autor{id: 3
Nombre: Rosario Fernandez
Nacionalidad: España}}
Libro {isbn: V555
Titulo: Software de un PC
Año de Publicacion: 2014
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}}

```

```

Se busca un libro por isbn 2678
Libro {isbn: 2678
Titulo: Biologia Basica
Año de Publicacion: 2017
Autor{id: 1
Nombre: Alberto Gonzalez
Nacionalidad: Argentina}}
.....
Libros encontrados del año 2021 :
////////////////////////////////////
Se imprimen los libros luego de eliminar uno
Lista de libros de la biblioteca: Biblioteca Universitaria
Libro {isbn: 2678
Titulo: Biologia Basica
Año de Publicacion: 2017
Autor{id: 1
Nombre: Alberto Gonzalez
Nacionalidad: Argentina}}
Libro {isbn: 1987
Titulo: Hardware de un PC
Año de Publicacion: 2015
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}}
Libro {isbn: P667
Titulo: Lenguaje C++

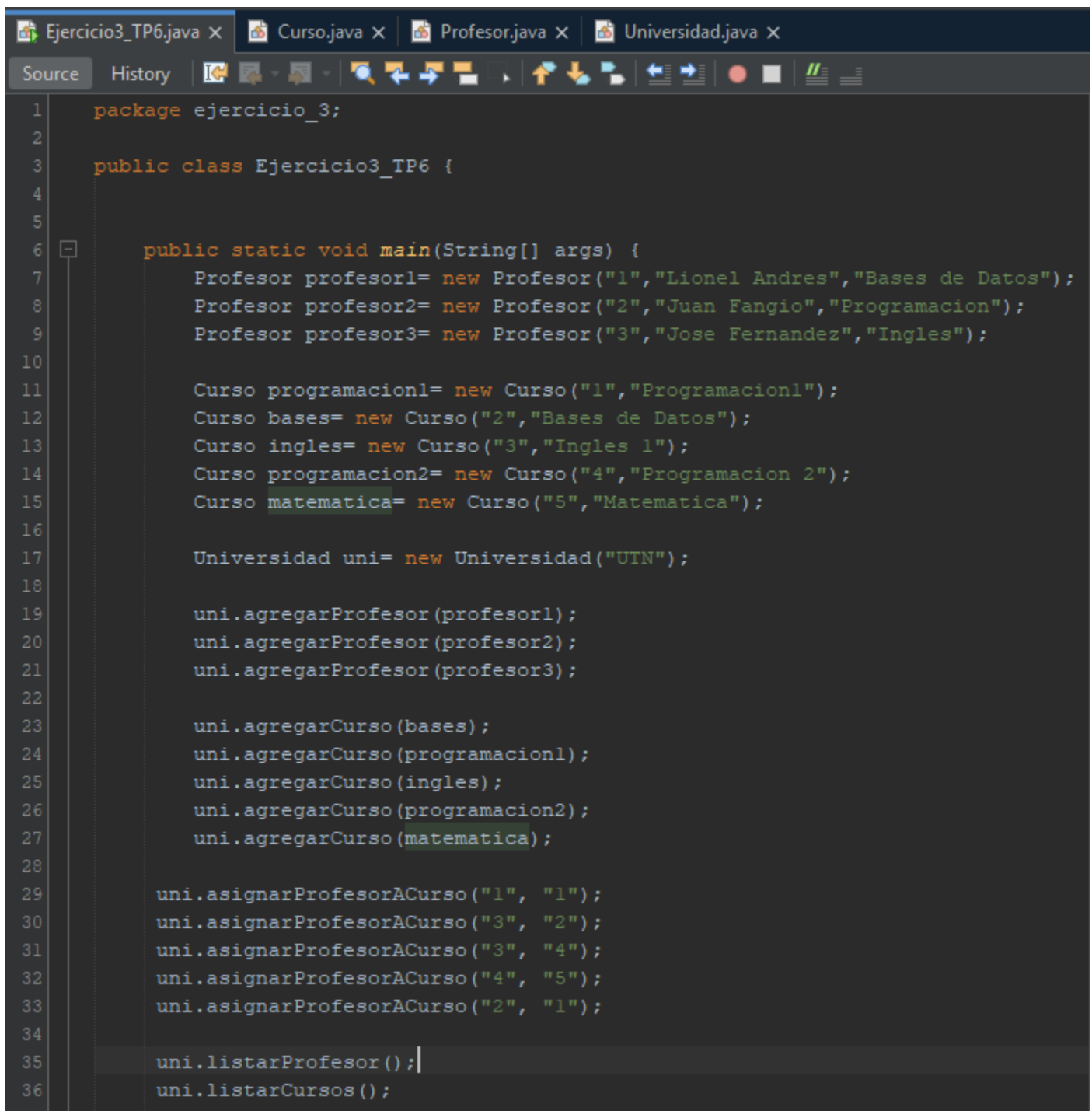
```

```

Año de Publicacion: 2019
Autor{id: 3
Nombre: Rosario Fernandez
Nacionalidad: España}}
Libro {isbn: V555
Titulo: Software de un PC
Año de Publicacion: 2014
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}}
Cantidad total de libros en la biblioteca Biblioteca Universitaria: 4
Autores de los libros:
Autor{id: 1
Nombre: Alberto Gonzalez
Nacionalidad: Argentina}
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}
Autor{id: 3
Nombre: Rosario Fernandez
Nacionalidad: España}
Autor{id: 2
Nombre: David Perez
Nacionalidad: Chile}
BUILD SUCCESSFUL (total time: 0 seconds)

```

3) Ejercicio3_TP6 (Main):



```
1 package ejercicio_3;
2
3 public class Ejercicio3_TP6 {
4
5
6     public static void main(String[] args) {
7         Profesor profesor1= new Profesor("1","Lionel Andres","Bases de Datos");
8         Profesor profesor2= new Profesor("2","Juan Fangio","Programacion");
9         Profesor profesor3= new Profesor("3","Jose Fernandez","Ingles");
10
11         Curso programacion1= new Curso("1","Programacion1");
12         Curso bases= new Curso("2","Bases de Datos");
13         Curso ingles= new Curso("3","Ingles 1");
14         Curso programacion2= new Curso("4","Programacion 2");
15         Curso matematica= new Curso("5","Matematica");
16
17         Universidad uni= new Universidad("UTN");
18
19         uni.agregarProfesor(profesor1);
20         uni.agregarProfesor(profesor2);
21         uni.agregarProfesor(profesor3);
22
23         uni.agregarCurso(bases);
24         uni.agregarCurso(programacion1);
25         uni.agregarCurso(ingles);
26         uni.agregarCurso(programacion2);
27         uni.agregarCurso(matematica);
28
29         uni.asignarProfesorACurso("1", "1");
30         uni.asignarProfesorACurso("3", "2");
31         uni.asignarProfesorACurso("3", "4");
32         uni.asignarProfesorACurso("4", "5");
33         uni.asignarProfesorACurso("2", "1");
34
35         uni.listarProfesor();
36         uni.listarCursos();
37     }
```

```

        System.out.println("////////////////////////////////////////");
        System.out.println("Se reasigna el curso de Programacion 1 al docente"
            + "Leandro De Angelis");
        uni.asignarProfesorACurso("1", "1");
        uni.listarCursos();
        uni.listarProfesor();

        System.out.println("////////////////////////////////////////");
        System.out.println("Se quita el curso de id=2 y se observan los profesores");
        uni.eliminarCurso(matematica);
        uni.listarProfesor();

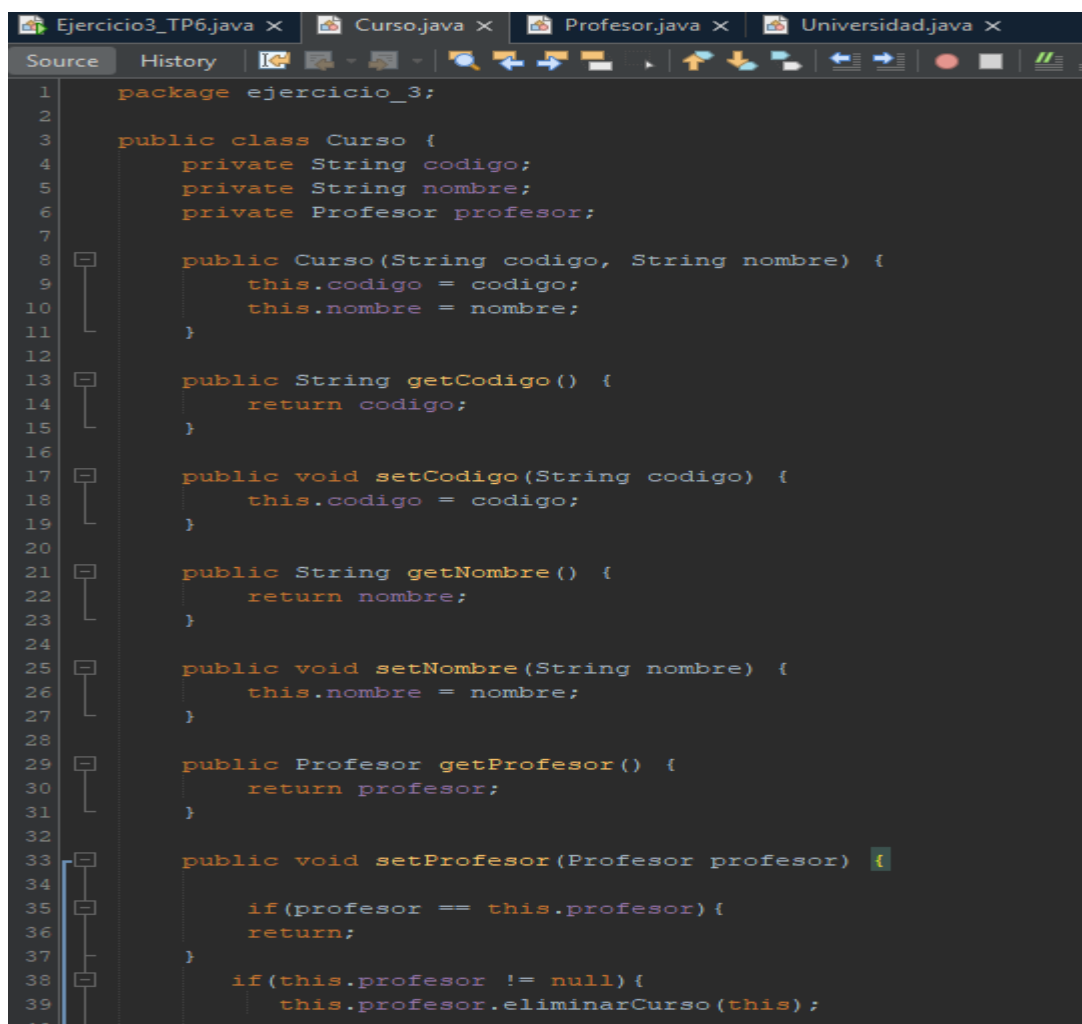
        uni.listarCursos();

        System.out.println("////////////////////////////////////////");
        System.out.println("Se quita al profesor de id=1");
        uni.eliminarProfesor(profesor1);

        uni.listarProfesor();
    }
}

```

Curso:



```

Ejercicio3_TP6.java × Curso.java × Profesor.java × Universidad.java ×
Source History
1 package ejercicio_3;
2
3 public class Curso {
4     private String codigo;
5     private String nombre;
6     private Profesor profesor;
7
8     public Curso(String codigo, String nombre) {
9         this.codigo = codigo;
10        this.nombre = nombre;
11    }
12
13    public String getCodigo() {
14        return codigo;
15    }
16
17    public void setCodigo(String codigo) {
18        this.codigo = codigo;
19    }
20
21    public String getNombre() {
22        return nombre;
23    }
24
25    public void setNombre(String nombre) {
26        this.nombre = nombre;
27    }
28
29    public Profesor getProfesor() {
30        return profesor;
31    }
32
33    public void setProfesor(Profesor profesor) {
34
35        if(profesor == this.profesor){
36            return;
37        }
38
39        if(this.profesor != null){
40            this.profesor.eliminarCurso(this);

```

```

    }

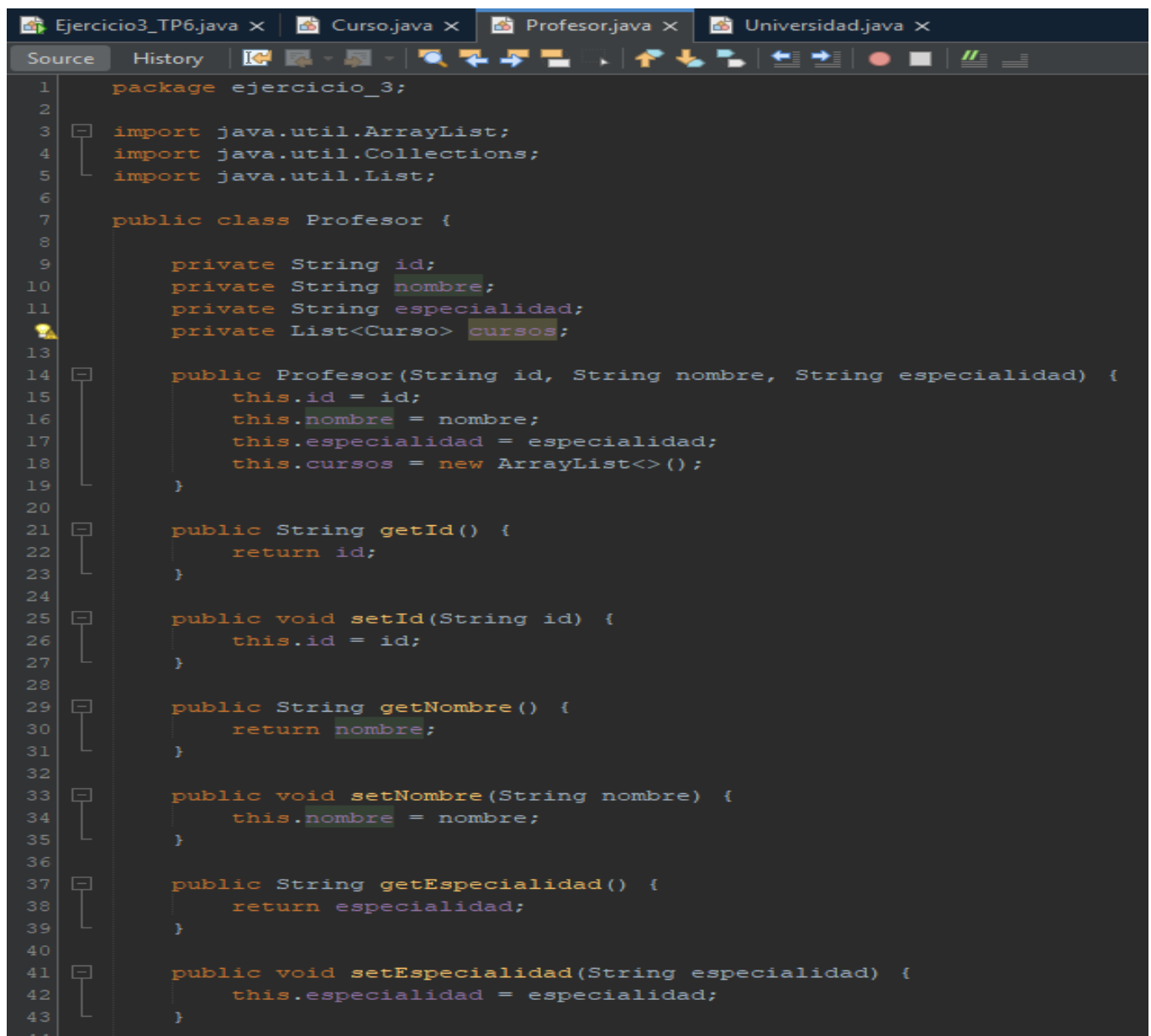
    this.profesor=profesor;

    if(profesor!=null && !profesor.getCursos().contains(this)){
        profesor.agregarCurso(this);
    }
}

public void mostrarInfo(){
    System.out.println("Codigo del curso: "+codigo+
        "\nNombre del curso es: "+nombre+
        "\nProfesor asignado al curso: "+profesor.getNombre() );
}
}

```

Profesor:



```

package ejercicio_3;

import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class Profesor {

    private String id;
    private String nombre;
    private String especialidad;
    private List<Curso> cursos;

    public Profesor(String id, String nombre, String especialidad) {
        this.id = id;
        this.nombre = nombre;
        this.especialidad = especialidad;
        this.cursos = new ArrayList<>();
    }

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getEspecialidad() {
        return especialidad;
    }

    public void setEspecialidad(String especialidad) {
        this.especialidad = especialidad;
    }
}

```



```
public List<Curso> getCursos() {  
    return Collections.unmodifiableList(cursos);  
}  
  
public void agregarCurso(Curso curso) {  
    if (curso != null && !cursos.contains(curso)) {  
        cursos.add(curso);  
        if (curso.getProfesor() != this) {  
            curso.setProfesor(this);  
        }  
    }  
}  
  
public void eliminarCurso(Curso curso) {  
    if (curso != null && cursos.contains(curso)) {  
        cursos.remove(curso);  
        if (curso.getProfesor() == this) {  
            curso.setProfesor(null);  
        }  
    }  
}  
  
public void listarCursos() {  
    System.out.println("Cursos del profesor: " + nombre);  
    for (Curso curso: cursos) {  
        curso.mostrarInfo();  
    }  
}  
  
public void mostrarInfo() {  
    System.out.println("Profesor con id: "+id+  
        "\nNombre: "+nombre+  
        "\nEspecialidad: "+especialidad+  
        "\nEnseña en "+cursos.size()+" cursos.");  
}
```

Universidad:



```
1 package ejercicio_3;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class Universidad {
7     private String nombre;
8     private List<Curso> cursos;
9     private List<Profesor> profesores;
10
11
12     public Universidad(String nombre) {
13         this.nombre = nombre;
14         this.cursos = new ArrayList<>();
15         this.profesores = new ArrayList<>();
16     }
17
18     public void agregarProfesor(Profesor profesor) {
19         if (profesor != null && !profesores.contains(profesor)) {
20             profesores.add(profesor);
21         }
22     }
23
24     public void agregarCurso(Curso curso) {
25         if (curso != null && !cursos.contains(curso)) {
26             cursos.add(curso);
27         }
28     }
29
30     public void eliminarProfesor(Profesor profesor) {
31         if (profesor != null && profesores.contains(profesor)) {
32             for (Curso curso : new ArrayList<>(profesor.getCursos())) {
33                 curso.setProfesor(null);
34             }
35             profesores.remove(profesor);
36         }
37     }
38
39     public void eliminarCurso(Curso curso) {
40         if (curso != null && cursos.contains(curso)) {
41             curso.setProfesor(null);
42             cursos.remove(curso);
43         }
44     }
45 }
```

```
48 public void listarProfesor() {  
49     System.out.println("La cantidad de profesores de la universidad " + nombre  
50         + " son " + profesores.size());  
51     for (Profesor profesor : profesores) {  
52         profesor.mostrarInfo();  
53         System.out.println("////////////////");  
54     }  
55 }  
56
```

```
57 public void listarCursos() {  
58     System.out.println("La universidad " + nombre  
59         + " tiene " + cursos.size() + " cursos y son: ");  
60     for (Curso curso : cursos) {  
61         curso.mostrarInfo();  
62     }  
63 }  
64  
65  
66  
67 public Profesor buscarProfesorPorId(String id) {  
68     if (id != null) {  
69         for (Profesor profesor : profesores) {  
70             if (profesor.getId().equals(id)) {  
71                 return profesor;  
72             }  
73         }  
74     }  
75     return null;  
76 }  
77  
78  
79  
80 public Curso buscarCursoPorCodigo(String codigo) {  
81     if (codigo != null) {  
82         for (Curso curso : cursos) {  
83             if (curso.getCodigo().equals(codigo)) {  
84                 return curso;  
85             }  
86         }  
87     }  
88     return null;  
89 }  
90  
91 public void asignarProfesorACurso(String codigoCurso, String idProfesor) {  
92     Profesor profesor = this.buscarProfesorPorId(idProfesor);  
93     Curso curso = this.buscarCursoPorCodigo(codigoCurso);  
94     if (profesor != null && curso != null) {  
95         curso.setProfesor(profesor);  
96     } else {  
97         System.out.println("El docente/curso no existe "  
98             + "en la universidad "+ nombre);  
99     }  
100 }
```

Build:

```
run:
El docente/curso no existe en la universidad UTN
El docente/curso no existe en la universidad UTN
La cantidad de profesores de la universidad UTN son 3
Profesor con id: 1
Nombre: Lionel Andres
Especialidad: Bases de Datos
Enseña en 2 cursos.
////////////////
Profesor con id: 2
Nombre: Juan Fangio
Especialidad: Programacion
Enseña en 1 cursos.
////////////////
Profesor con id: 3
Nombre: Jose Fernandez
Especialidad: Ingles
Enseña en 0 cursos.
////////////////
La universidad UTN tiene 5 cursos y son:
Codigo del curso: 2
Nombre del curso es: Bases de Datos
Profesor asignado al curso: Lionel Andres
Codigo del curso: 1
Nombre del curso es: Programacion1
Profesor asignado al curso: Lionel Andres
Codigo del curso: 3
Nombre del curso es: Ingles 1
Profesor asignado al curso: Juan Fangio
```