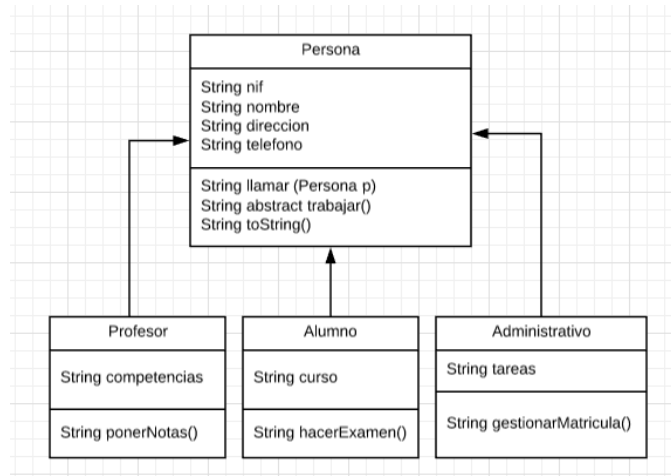


Ejercicio 1

Se nos pide que pasemos este diagrama a java.



Sera unos 5 archivos en total ya que tendremos que crear un main para poder utilizar las otras clases.

La clase Persona: Tiene que ser abstracta ya que vamos a crear el método trabajar que lo utilizaremos en todas las clases.

```

1 package act1;
2
3 public abstract class Persona {
4
5     private String dni;
6     private String Nombre;
7     private String Direccion;
8     private String Telefono;
9
10    public Persona() {
11        this.dni = "";
12        this.Nombre = "";
13        this.Direccion = "";
14        this.Telefono = "";
15    }
16
17    public Persona(String dni, String nom, String dir, String tel) {
18        this.dni = dni;
19        this.Nombre = nom;
20        this.Direccion = dir;
21        this.Telefono = tel;
22    }
23
24
25    public String llamar(Persona p) {
26        return(this.Nombre + " llama a " + p.getNombre());
27    }
28
29    abstract public String trabajar();
30    @Override
31
32
33    public String toString(){
34        return(" Persona | " +this.dni+ " " +this.Nombre+ " " +this.Telefono+ " ");
35    }
36
37
38
39    public String getDni() {
40        return dni;
41    }
42
43
44
45
46
47
48
49
50
51

```

El método llamar lo que hace es llamar una persona a otra es decir que alumno puede llamar a profesor.

La clase Profesor: Para poder utilizar la clase persona para “crear” tenemos que hacer que sea un extends de esta manera utilizando el super() podrás utilizar lo que esta arriba de esta clase (en este caso es Persona)

```

1 package acti;
2
3 public class Profesor extends Persona{
4     private String competencias;
5
6
7
8     public Profesor(String dni, String nom, String dir, String tel, String com) {
9         super(dni, nom, dir, tel);
10        this.competencias = com;
11    }
12
13    public Profesor() {
14        super();
15        this.competencias = "";
16    }
17
18    public String ponerNotas() {
19        return("El profesor " + super.getNombre() + " va a corregir exámenes");
20    }
21
22    public String toString(){
23        return(" Persona | "+super.getDni()+ " " +super.getNombre()+ " " +super.getTelefono()+ " ");
24    }
25
26    @Override
27    public String trabajar() {
28        return("El profesor "+super.getNombre()+ " va a dar clase");
29    }
30
31    public String getCompetencias() {
32        return competencias;
33    }
34
35    public void setCompetencias(String competencias) {
36        this.competencias = competencias;
37    }
38
39
40 }

```

Ademas de eso tenemos el método ponerNotas lo que hace es llamar al nombre del profesor usando el super.getNombre().

ToString es una salida simple de datos en la que dices el DNI, Nombre ,Teléfono.

Y aquí llamamos al método trabajar de la clase persona que como se dijo antes es un método abstracto.

Y el resto de clases no varían en nada solo en el nombre de los métodos y la salida de texto.

```

1 package acti;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         Persona p;
8         Alumno al = new Alumno("12345678H", "Alicia Torres", "Islas filipinas", "789456", "1ª Daw");
9         Profesor pf = new Profesor("98765432F", "Raul S", "Canal", "741258963", "BBDO y program");
10        Administrativo ad = new Administrativo("36925814T", "Maria dolore nieve", "La Viña", "78952461", " matriculas y asistencia");
11
12        System.out.println("Lista de personas");
13        System.out.println(al.toString());
14        System.out.println(pf.toString());
15        System.out.println(ad.toString());
16        System.out.println(" ");
17        System.out.println("-----llamada-----");
18        System.out.println(al.llamar(pf));
19        System.out.println(" ");
20        System.out.println("-----hacer examen-----");
21        System.out.println(al.hacerExamen());
22        System.out.println(" ");
23        System.out.println("-----poner notas de exámenes-----");
24        System.out.println(pf.ponerNotas());
25        System.out.println(" ");
26        System.out.println("-----papeleo-----");
27        System.out.println(ad.gestionarMatricula());
28        System.out.println(" ");
29        System.out.println("-----Trabajo de cada persona-----");
30        System.out.println(al.trabajar());
31        System.out.println(pf.trabajar());
32        System.out.println(ad.trabajar());
33    }
34
35 }
36
37

```

Problems | Javadoc | Declaration | Console

terminated: Main [J] [Java Application] C:\Program Files\AdoptOpenJDK\jdk-15.0.1-hotspot\bin\javaw.exe (4 mar. 2021 15:52:23 - 15:52:23)

```

Lista de personas
Persona | 12345678H Alicia Torres 789456
Persona | 98765432F Raul S 741258963
Persona | 36925814T Maria dolore nieve 78952461

-----llamada-----
Alicia Torres llama a Raul S

-----hacer examen-----
El alumno Alicia Torres va a hacer un examen

-----poner notas de exámenes-----
El profesor Raul S va a corregir exámenes

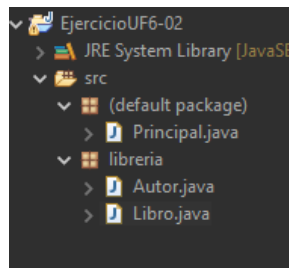
```

En el main lo que se hace es crear un Alumno un Profesor y un Administrado.

Una vez creados se saca una lista de todos los que se han creado utilizando el método toString.
Y luego se utiliza uno a uno el resto de métodos de cada clase

Ejercicio 2

En esta actividad se nos pide crear un main que se localiza en un paquete diferente al que se encuentran las clases Libro y Autor



La clase Libro:Hace una llamada a la clase autor para utilizar los datos nombre y bio.

```
1 package libreria;
2
3 public class Libro {
4
5     private String titulo;
6     private String genero;
7     private Autor autor;
8
9
10    public Libro(String tit, String gen, Autor aut) {
11        this.titulo = tit;
12        this.genero = gen;
13        this.autor = aut;
14    }
15
16    public Libro(String tit, String gen, String aut, String bio) {
17        this.titulo=tit;
18        this.genero=gen;
19        this.autor = new Autor(aut, bio);
20    }
21
22    public String toString() {
23        return("El libro " +this.getTitulo()+ " "+"del genero " + this.getGenero()+ " "+"escrito por " + this.getAutor());
24    }
25
26
27
28
29
30    public Autor getAutor() {
31        return autor;
32    }
33
34    public void setAutor(Autor autor) {
35        this.autor = autor;
36    }
37
38    public String getTitulo() {
39        return titulo;
40    }
41 }
```

Para eso el primer public Libro es de agregación débil esto significa que si se borra libro autor sigue existiendo, pero en cambio, el segundo es de agregación fuerte ya que si borras el libro el autor deja de existir. Ya que como se puede ver en el segundo caso se esta creando un autor.

La clase Autor: Es una clase simple en la que se asignan los datos nombre y biografia tiene un toString por si queremos hacer una salida de texto para saber que hizo el autor.

```

1 package libreria;
2
3 public class Autor {
4     private String nombre;
5     private String biografia;
6
7     public Autor() {
8         this.nombre = "";
9         this.biografia = "";
10    }
11
12
13
14    public Autor (String nom, String bio) {
15        this.nombre = nom;
16        this.biografia= bio;
17    }
18
19    public String toString() {
20        return "El autor " + this.getNombre() + ", con la biografia " + this.getBiografia();
21    }
22
23
24
25    public String getNombre() {
26        return nombre;
27    }
28
29
30    public void setNombre(String nombre) {
31        this.nombre = nombre;
32    }
33
34
35
36
37    public String getBiografia() {
38        return biografia;

```

En el main: Creamos un autor y un libro (linea 8 y 10) esta es la manera de agregación débil que mencionamos anteriormente hacemos unas salidas de texto para ver el libro y para ver el autor.

```

1 import libreria.Autor;
2
3 public class Principal {
4
5     public static void main(String[] args) {
6
7         Autor aut = new Autor("Julio Cortázar", "Escritor y traductor argentino.");
8
9         Libro lib = new Libro("Rayuela", "Novela", aut);
10        System.out.println("-----Autor-----");
11        System.out.println(aut.toString());
12        System.out.println(" ");
13        System.out.println("-----Libro-----");
14        System.out.println(lib.toString());
15
16
17        System.out.println(" ");
18        Libro lib2 = new Libro("Rayuela", "Novela", "Julio Cortázar", "Escritor y traductor argentino.");
19        System.out.println(lib2.toString());
20        lib2.getAutor().setBiografia("Protesto contra el regimen militar");
21        System.out.println(lib2.toString());
22        lib2.setGenero("Poesia");
23        System.out.println(lib2.toString());
24    }
25 }
26
27
28

```

Problems Javadoc Declaration Console

```

<terminated> Principal [Java Application] C:\Program Files\AdoptOpenDK\jdk-15.0.1.9-hotspot\bin\javaw.exe (4 mar. 2021 17:16:15 - 17:16:15)

-----Autor-----
El autor Julio Cortázar, con la biografia Escritor y traductor argentino.

-----Libro-----
El libro Rayuela del genero Novela escrito por El autor Julio Cortázar, con la biografia Escritor y traductor argentino.
El libro Rayuela del genero Novela escrito por El autor Julio Cortázar, con la biografia Protesto contra el regimen militar
El libro Rayuela del genero Poesia escrito por El autor Julio Cortázar, con la biografia Protesto contra el regimen militar

```

Luego creamos un nuevo libro que lo llamamos lib2 esta manera de crear el libro es la que se monstruo antes como agregación fuerte lo mostramos con el toString.

Y luego cambiamos un dato del autor desde libro (linea 21) lo que se hace es llamar a lib2 pides el (usando el get) autor y le cambias el dato de la biografia(usando el set).