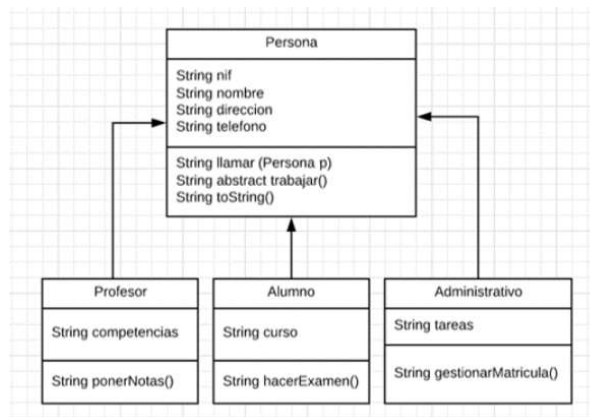


AI-6. Centro educativo

Especificaciones para el ejercicio 1:

Para la gestión de un centro educativo, un programa necesita contemplar las entidades profesor, alumno y administrativo, teniendo en cuenta que todos, a su vez, son personas.

1. Crea la estructura de clases que se corresponda con el siguiente diagrama:



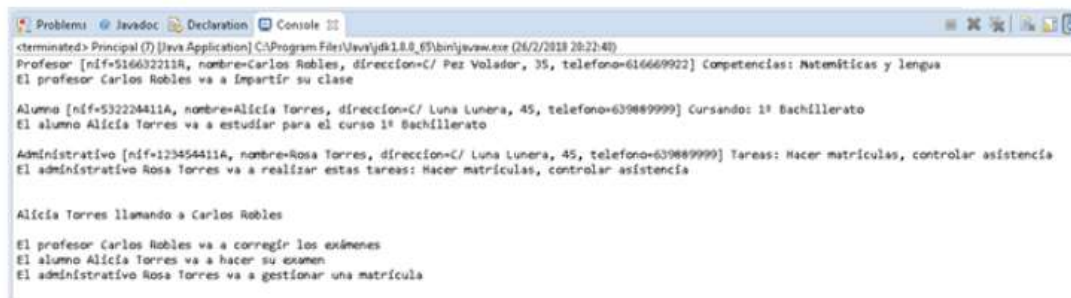
- La clase *Persona* será abstracta.
- Todas las propiedades serán privadas y accesibles mediante métodos *get/set*.
- El método *llamar()* devolverá una cadena similar a esta: “Alicia Torres llamando a Carlos Robles”. Como parte de la cadena se utilizará la propiedad *nombre* del objeto al que se aplica y del objeto pasado como argumento.
- El método *trabajar()* se especializará en cada subclase para devolver una cadena que indique el tipo de tarea que realiza cada perfil de persona. Por ejemplo:
 - *Profesor*: “El profesor Carlos Robles va a impartir su clase”. Como parte de la cadena usarás la propiedad *nombre*.
 - *Alumno*: “El alumno Alicia Torres va a estudiar para el curso 1º Bachillerato”. Como parte de la cadena usarás las propiedades *nombre* y *curso*.
 - *Administrativo*: “El administrativo Rosa Torres va a realizar estas tareas: hacer matrículas, controlar asistencia”. Como parte de la cadena utilizarás las propiedades *nombre* y *tareas*.
 -
- Además, las clases derivadas ampliarán su especialización con métodos propios como *ponerNotas()*, *hacerExamen()* o *gestionarMatricula()*. Cada uno de estos métodos devolverá una cadena de este tipo:
 - *ponerNotas()*: “El profesor Carlos Robles va a corregir los exámenes”. Como parte de la cadena tienes que utilizar la propiedad *nombre*.

AI-6. Centro educativo

- *hacerExamen()*: “El alumno Alicia Torres va a hacer su examen”. Como parte de la cadena debes utilizar la propiedad nombre.
 - *gestionarMatricula()*: “El administrativo Rosa Torres va a gestionar una matrícula”. Como parte de la cadena utilizarás la propiedad nombre.
 - Además de lo especificado en el diagrama de clases, debes sobrescribir el método *toString()* en cada una de las clases para que muestre toda la información posible. Puedes hacerlo de forma automática con ayuda de Eclipse (clic derecho / Source / Generate toString...).
2. Para terminar crea la clase *Principal* con método *main* y realiza las siguientes acciones:

- Crea un objeto *Profesor*, un objeto *Alumno* y un objeto *Administrativo*.
- Ejecuta el método *toString()* sobre cada uno de ellos.
- Ejecuta el método *trabajar()* sobre cada uno de ellos.
- Ejecuta el método *llamar(Persona p)* para que el alumno llame al profesor.
- Ejecuta el método *ponerNotas()* del objeto *Profesor*.
- Ejecuta el método *hacerExamen()* del objeto *Alumno*.
- Ejecuta el método *gestionarMatricula()* del objeto *Administrativo*.

Ejemplo de ejecución:



```
<terminated> Principal (7) [Java Application] C:\Program Files\Java\jdk1.8.0_65\bin\javaw.exe (26/2/2018 20:22:40)
Profesor [nif=5166322118, nombre=Carlos Robles, direccion=C/ Pez Volador, 35, telefono=616669922] Competencias: Matemáticas y lengua
El profesor Carlos Robles va a impartir su clase

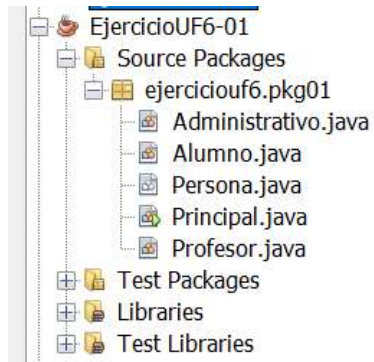
Alumno [nif=532220411A, nombre=Alicia Torres, direccion=C/ Luna Lunera, 45, telefono=639889999] Cursando: 1º Bachillerato
El alumno Alicia Torres va a estudiar para el curso 1º Bachillerato

Administrativo [nif=123454411A, nombre=Rosa Torres, direccion=C/ Luna Lunera, 45, telefono=639889999] Tareas: Hacer matrículas, controlar asistencia
El administrativo Rosa Torres va a realizar estas tareas: Hacer matrículas, controlar asistencia

Alicia Torres llamando a Carlos Robles
El profesor Carlos Robles va a corregir los exámenes
El alumno Alicia Torres va a hacer su examen
El administrativo Rosa Torres va a gestionar una matrícula
```

AI-6. Centro educativo

Se han guardado las clases con esta estructura:



Mostramos la clases que se han creado:

// Empezamos con la clase persona que es la clase abstracta y nos sirve de “modelo” para el resto de clases.

```
package ejerciciouf6.pkg01;
```

```
/**
```

```
 * @author Silvia Piñel Fañanás
```

```
 */
```

```
public abstract class Persona {
```

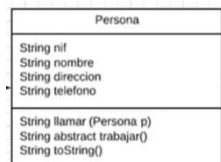
```
    private String nif;
```

```
    private String nombre;
```

```
    private String direccion;
```

```
    private String telefono;
```

// Definimos los atributos de la clase Persona.



//Declaramos el constructor sin parámetros

```
public Persona(){
```

```
    nif="";
```

```
    nombre="";
```

```
    direccion ="";
```

```
    telefono ="";
```

```
}
```

AI-6. Centro educativo

//Declaramos el constructor con parámetros.

```
public Persona(String nif,String nombre,String direccion,String telefono){  
    this.nif=nif;  
    this.nombre=nombre;  
    this.direccion=direccion;  
    this.telefono=telefono;  
}
```

//Definimos los getters y los setters.

```
public String getNif() {  
    return nif;  
}  
public void setNif(String nif) {  
    this.nif = nif;  
}  
public String getNombre() {  
    return nombre;  
}  
public void setNombre(String nombre) {  
    this.nombre = nombre;  
}  
public String getDireccion() {  
    return direccion;  
}  
public void setDireccion(String direccion) {  
    this.direccion = direccion;  
}  
public String getTelefono() {  
    return telefono;  
}  
public void setTelefono(String telefono) {  
    this.telefono = telefono;    }
```

AI-6. Centro educativo

// La clase persona tiene dos métodos, llamar y trabajar que no definimos ya que esto se hará en cada una de las subclases.

```
public abstract String llamar(Persona p);  
  
public abstract String trabajar ();
```

// Sobrescribimos el método toString para muestre los datos que nos interesa.

```
@Override  
  
public String toString() {  
  
    return "Persona{" + "nif=" + nif + ", nombre=" + nombre + ", direccion=" + direccion + ",  
telefono=" + telefono + '}';  
  
}  
}
```

//Definimos la clase Administrativo a partir de la clase abstracta Persona.

```
package ejerciciouf6.pkg01;  
  
/**  
  
 * @author Silvia Piñel Fañanás  
 */  
  
public class Administrativo extends Persona {  
  
    private String tareas;
```

//Declaramos el constructor sin parámetros.

```
public Administrativo() {  
  
    super();  
  
    this.tareas=""; }  

```

//Declaramos el constructor con parámetros.

```
public Administrativo(String nif, String nombre, String direccion, String telefono, String  
tareas) {  
  
    super(nif, nombre, direccion, telefono);  
  
    this.tareas=tareas; }  

```

//Declaramos los getters y los setters.

```
public String getTareas() {  
  
    return tareas; }  
  
public void setTareas(String tareas) {  
  
    this.tareas = tareas; }  

```

AI-6. Centro educativo

// Sobrescribimos el método trabajar que lo definimos para esta clase particular.

@Override

public String trabajar () {

return "El Administrativo "+getNombre() + " va a realizar estas tareas: hacer matrículas,
controlar asistencia ";

}

// Sobrescribimos el método llamar que lo definimos para esta clase particular.

public String llamar(Persona p){

return getNombre() + " llamando a " + p.getNombre();

};

// Método creado para la clase Administrativo.

public String gestionarMatricula () {

return "El Administrativo "+getNombre() + " va a gestionar una matrícula";

};

// Sobrescribimos el método toString para que muestre todos los datos disponibles de Adminitrativo.

@Override

public String toString() {

return "Los datos del adminitrativo "+getNombre()+ " son: "+" NIF= " + getNif()
+ ",Direccion= " + getDireccion()

+ ",Teléfono = " + getTelefono() + ",tareas= " + getTareas();

}

}

AI-6. Centro educativo

//Definimos la clase Alumno a partir de la clase abstracta Persona.

```
package ejerciciouf6.pkg01;
```

```
/**
```

```
* @author Silvia Piñel Fañanás
```

```
*/
```

```
public class Alumno extends Persona {
```

```
    private String curso;
```

//Definimos el Constructor sin parámetros

```
    public Alumno() {
```

```
        super();
```

```
        this.curso="";
```

```
    }
```

/*Definimos el Constructor con parámetros, a este constructor le hemos añadido el parámetro curso que es específico de la clases Alumno*/

```
    public Alumno(String nif, String nombre, String direccion, String telefono, String curso) {
```

```
        super(nif, nombre, direccion, telefono);
```

```
        this.curso=curso;
```

```
    }
```

// Definimos los getters y los setters

```
    public String getCurso() {
```

```
        return curso;
```

```
    }
```

```
    public void setCurso(String curso) {
```

```
        this.curso = curso;
```

```
    }
```

//Sobrescribimos los métodos trabajar, llamar y las definimos para esta clase en concreto.

```
    @Override
```

```
    public String trabajar () {
```

```
        return "El alumno "+getNombre() + " va a estudiar "+ getCurso();
```

```
    }
```

AI-6. Centro educativo

```
public String llamar(Persona p){  
    return getNombre() + " llamando a " + p.getNombre();  
};
```

// El método hacerExamen es específico de la clase Alumno.

```
public String hacerExamen (){  
    return "El alumno "+getNombre() + " va a hacer su examen";  
}
```

/* Sobrescribimos el método toString() para definirlo con toda la información que podemos mostrar.*/

@Override

```
public String toString() {  
    return "Los datos del alumno "+getNombre()+ " son: "+" NIF= " + getNif() +  
    ",Direccion= " + getDireccion() + ",Teléfono = " + getTelefono() + ",Curso= " +  
    getCurso();  
}  
}
```

//Definimos la clase Profesor a partir de la clase abstracta Persona.

```
package ejerciciouf6.pkg01;
```

```
/**
```

```
* @author Silvia Piñel Fañanás
```

```
*/
```

```
public class Profesor extends Persona {
```

```
    private String competencias;
```

//Declaramos el constructor por defecto.

```
public Profesor() {  
    super();  
    this.competencias="";  
}
```


AI-6. Centro educativo

/*Declaramos el constructor con parámetros, el parámetro competencias es específico de la clase Profesor.*/

```
public Profesor(String nif, String nombre, String direccion, String telefono, String
competencias) {
    super(nif, nombre, direccion, telefono);
    this.competencias=competencias;
}
```

// Declaramos los getters y los setters.

```
public String getCompetencias() {
    return competencias;
}

public void setCompetencias(String competencias) {
    this.competencias = competencias;
}
```

//Sobrescribimos los métodos trabajar y llamar.

```
@Override
public String trabajar (){
    return "El profesor "+getNombre() + " va a impartir su clase";
}

public String llamar(Persona p){
    return getNombre() + " llamando a " + p.getNombre();
};

public String ponerNotas (){
    return "El profesor "+getNombre() + " va a corregir los exámenes";
}
```

// Sobrescribimos el método toString() que muestra los distintos datos del Profesor.

```
@Override
public String toString() {
    return "Los datos del profesor "+getNombre()+ " son: "+" NIF= " + getNif() + ",Direccion= " +
getDireccion() + ",Teléfono = " + getTelefono() + ",competencias= " + getCompetencias();
}
}
```

AI-6. Centro educativo

```
package ejerciciouf6.pkg01;
```

```
/**
```

```
* @author Silvia Piñel Fañanás
```

```
*/
```

```
/*
```

Crea un objeto Profesor, un objeto Alumno y un objeto Administrativo.

Ejecuta el método toString() sobre cada uno de ellos.

Ejecuta el método trabajar() sobre cada uno de ellos.

Ejecuta el método llamar(Persona p) para que el alumno llame al profesor.

Ejecuta el método ponerNotas() del objeto Profesor.

Ejecuta el método hacerExamen() del objeto Alumno.

Ejecuta el método gestionarMatricula() del objeto Administrativo.

```
*/
```

```
public class Principal {
```

```
    public static void main(String[] args) {
```

```
        // Creamos un objeto Profesor, un objeto Alumno y un objeto Administrativo.
```

```
        Profesor profe = new Profesor("516632211R", "Carlos Robles", "C/ Pez Volador, 35",  
        "616669922", "Matemáticas y Lengua");
```

```
        Alumno alum = new Alumno("532224411A", "Alicia Torres", "C/ Luna lunera, 45",  
        "639889999", "1º Bachillerato");
```

```
        Administrativo admin = new Administrativo("123454411A", "Rosa Torres", "Avda de  
        prueba, 25", "695884456", "Hacer Matriculas, controlar asistencias");
```

```
        // Trabajamos con el objeto Profesor.
```

```
        System.out.println (profe.toString()); //Muestra la info del profesor
```

```
        System.out.println (profe.trabajar()); //Muestra la frase indicada para el objeto Profesor.
```

```
        System.out.println (profe.ponerNotas()); //Ejecutamos el método específico de Profesor.
```

```
        // Trabajamos con el objeto Alumno.
```

```
        System.out.println (alum.toString()); //Muestra la info del Alumno.
```

```
        System.out.println (alum.trabajar()); //Muestra la frase indicada para el objeto Alumno.
```

```
        System.out.println (alum.hacerExamen()); //Ejecutamos el método específico de Alumno.
```

AI-6. Centro educativo

// Trabajamos con el objeto Administrativo.

System.out.println (admin.toString()); //Muestra la info del Administrativo.

//Muestra la frase indicada para el objeto Administrativo.

System.out.println (admin.trabajar());

//Ejecutamos el método específico de Administrativo.

System.out.println (admin.gestionarMatricula());

// Ejecutamos el método que permite al alumno llamar a un profesor.

System.out.println (alum.llamar(profe));

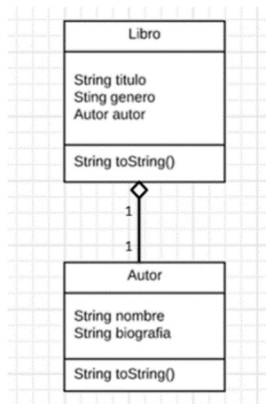
}

}

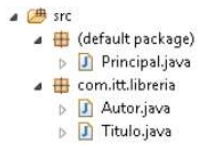
AI-6. Centro educativo

Especificaciones para el ejercicio 2:

Crea las clases *Libro* y *Autor* según el siguiente diagrama de clases:



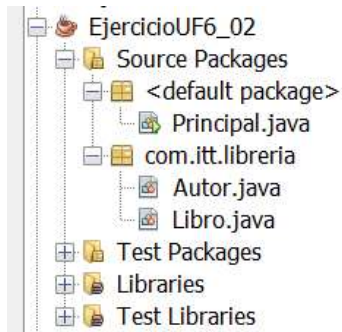
3. Las clases *Autor* y *Libro* estarán situadas en el paquete *com.itt.libreria*, mientras que en la clase *Principal*, donde incluirás el método *main*, estará la raíz del proyecto.



4. Cada clase debe tener su constructor con parámetros para recibir valores para todas las propiedades.
5. Todas las propiedades de ambas clases deben ser privadas y accesibles mediante métodos *get/set*.
6. El método *toString()* de ambas clases deberá devolver una cadena de texto con el estado del objeto (valores de todas las propiedades). Puedes crearlo de forma automática con ayuda de Eclipse.
7. También debes crear la clase *Principal*, con método *main*, donde crearás un objeto de tipo *Autor* y otro objeto de tipo *Libro*, al que se le pasará el objeto *Autor* como uno de los argumentos del constructor. Realizarás las siguientes operaciones con el objeto *Libro*:
 - Invoca al método *toString()* del nuevo objeto *Libro*.
 - Modifica la biografía del autor a través del objeto *Libro* que has creado.
 - Invoca al método *toString()* del objeto *Autor* que has creado de dos formas distintas: a partir de la referencia al objeto *Libro* y a partir de la referencia al objeto *Autor* que creaste.
 - Modifica el valor de la propiedad *género* y vuelve a invocar al método *toString()* para ver los cambios.

AI-6. Centro educativo

Se han guardado las clases con esta estructura:



Las clases *Autor* y *Libro* estarán situadas en el paquete *com.itt.libreria*. La clase principal en la raíz del proyecto.

/*Creamos las clases Libro y autor. Se trata de una Agregación de composición débil.*/

```
package com.itt.libreria;
```

```
/**
```

```
 * @author Silvia Piñel Fañanás
```

```
 */
```

```
public class Autor {
```

```
    private String nombre;
```

```
    private String biografia;
```

// Creamos el constructor sin parámetros

```
    public Autor () {
```

```
        nombre = "";
```

```
        biografia = "";
```

```
    }
```

// Definimos el constructor con parámetros

```
    public Autor(String nombre, String biografia) {
```

```
        this.nombre = nombre;
```

```
        this.biografia = biografia;
```

```
    }
```

AI-6. Centro educativo

// Definimos los métodos getters y los setters.

```
public String getNombre() {  
    return nombre;  
}  
  
public void setNombre(String nombre) {  
    this.nombre = nombre;  
}  
  
public String getBiografia() {  
    return biografia; }  
  
public void setBiografia(String biografia) {  
    this.biografia = biografia; }
```

//Sobrescribimos el método toString() para que nos muestre la información de Autor.

```
@Override  
public String toString() {  
    return "Autor{" + "nombre=" + nombre + ", biografia=" + biografia + '}';  
} }
```

// Creamos la clase Libro.

```
package com.itt.libreria;  
  
/**  
 * @author Silvia Piñel Fañanás  
 */  
  
public class Libro {  
    private String titulo;  
    private String genero;  
    private Autor escritor;
```

//Definimos el constructor sin parámetros.

```
public Libro () {  
    titulo="";  
    genero="";  
    escritor = null;  
}
```

AI-6. Centro educativo

//Definimos el constructor con parámetros.

```
public Libro(String titulo, String genero, Autor escritor) {  
    this.titulo = titulo;  
    this.genero = genero;  
    this.escriptor = escritor;  
}
```

//Declaramos los setters y los getters.

```
public String getTitulo() {  
    return titulo;  
}  
  
public void setTitulo(String titulo) {  
    this.titulo = titulo;  
}  
  
public String getGenero() {  
    return genero;  
}  
  
public void setGenero(String genero) {  
    this.genero = genero;  
}  
  
public Autor getEscriotor() {  
    return escritor;  
}  
  
public void setEscriotor(Autor escritor) {  
    this.escriptor = escritor;  
}
```

// Sobrescribimos el método toString() que nos muestra la info de libro.

```
@Override  
public String toString() {  
    return "Libro{" + "titulo=" + titulo + ", genero=" + genero + ", autor=" + escritor + '}';  
}  
}
```

AI-6. Centro educativo

/*Creamos la clase principal donde vamos a crear los objetos y llamar a los métodos de cada uno de ellos.

Los primero que hacemos es importar las librerías para que tanto autor como libro sean visibles en el mail. */

```
import com.itt.libreria.Autor;
```

```
import com.itt.libreria.Libro;
```

```
/**
```

```
*
```

```
* @author Silvia Piñel Fañanás
```

```
*/
```

```
public class Principal {
```

```
public static void main(String[] args) {
```

```
//Creamos el objeto escritor con sus distintos parámetros.
```

```
Autor escritor1 = new Autor("Stephen King", "Escritor estadounidense de novelas de terror ");
```

/*Creamos el objeto tomo1 con sus distintos parámetros. Nótese que le estamos pasando el objeto escritor como parámetro a objeto tomo1.

```
Libro tomo1 = new Libro ("IT","Terror",escritor1);
```

//Ejecutamos el método toString() para ver toda la información disponible que tenemos de escritor1.

```
System.out.println (escritor1.toString());
```

//Modificamos la biografía del autor a través del objeto tomo1 (Libro) y mostramos de nuevo la información para ver la variación que hemos hecho.

```
tomo1.getEscritor().setBiografia("Escritor estadounidense de novelas de varios géneros");
```

```
System.out.println (escritor1.toString());
```

//Modificamos el género del libro que hemos creado y mostramos como queda el objeto tomo1.

```
tomo1.setGenero("Suspense");
```

```
System.out.println (tomo1.toString());
```

```
}
```

```
}
```