Vamos a diseñar un programa en Java para gestionar los datos de los empleados de la empresa ACME S.A. En ella trabajan empleados de distintas categorías. De todos ellos interesa conocer el nombre, apellidos, dni, horas que trabajan a la semana y salario.

La clase **Empleado** reflejará todas las características comunes a todas las categorías y servirá de clase base para el resto de las clases. Todos sus atributos son privados.

La clase **Empleado** deberá tener un constructor con parámetros que recibirá el **nombre**, el **apellido** y el **dni**. El constructor asignará un **salario** de 20000 € anuales, 40 **horas a la semana** de trabajo y 15 **días de vacaciones**.

De los **Empleados** se podrá modificar todo menos las horas de trabajo a la semana.

Todas las clases que hereden de Empleado tienen que **estar obligadas** a desarrollar un método que se llama **calculaIngresos()** que devuelve la cantidad anual a cobrar.

La clase **socioCapitalista** hereda de Empleado y tiene el atributo privado **capitalAportado**. La cantidad anual que cobrará será el salario del resto de los empleados + 5000 €. Los socios capitalistas no tienen ningún día de vacaciones.

La clase **Vendedor** hereda de Empleado y tiene los atributos privados:

- **ventasRealizadas** (importe total vendido por el vendedor en un año. Al crear el objeto se pondrá a cero). Este atributo sólo se podrá modificar con los métodos:
 - o **vender**, que recibirá como parámetro el importe de una venta y se lo acumulará al atributo ventasRealizadas.
 - o **anularVenta**, que recibirá como parámetro el importe de la venta que hay que anular y se le restará a ventasRealizadas.
- **comisionACobrar**, que es un 1% de las ventas realizadas, y que en el momento de instanciar un objeto de clase Vendedor será cero. Este atributo sólo se podrá modificar con el método **calcularComision()**.
- Los vendedores tienen 5 días más de vacaciones que el resto de los empleados.

La clase **Vigilante** hereda de Empleado y tiene el atributo privado **plusPeligrosidad**, que supone 3000 € anuales y al que se le da valor en el constructor de la clase.

Todas las clases deben sobrescribir el método tostring() para que devuelva un String con todos los datos del objeto desde el que se invoca.

Escribe un programa en Java para realizar la gestión de la plantilla de la empresa, que tiene 2 socios capitalistas, 3 vendedores y 2 vigilantes, todos en un array de referencias a Empleado.

El programa debe mostrar un menú con las opciones:

- 1. Crear empleados. Creará los empleados generando los datos al azar y comprobando que no hay dos empleados distintos con el mismo dni.
- 2. Listar datos de los empleados (hay que usar el método toString().
- 3. Modificar datos de un empleado. Sólo se podrán modificar los datos que son comunes a todos los empleados.
- 4. Asignar ventas a los vendedores.
- 5. Mostrar la cantidad total que debe pagar la empresa en concepto de sueldos.
- 0. Salir

En el método main hay que invocar, al menos, a los métodos cuya firma se indica a continuación:

```
private static void crearEmpl eados(List<Empl eado> col eccion)
private static String generarDNI()
```

```
y al método cuyo código es:

private static boolean dni Repetido(String dni, List<Empleado> coleccion){
    boolean repetido=false;
    int k;
    if(k coleccion.contains(dni)){
        repetido=true;
    }
    return repetido;
```

Se recomienda utilizar la clase Leer que se ha usado en ejercicios anteriores.