



Un centro de salud al que se le ha encargado la administración de vacunas a los habitantes de un pequeño pueblo decide encargar una aplicación que le permita llevar el control de qué habitantes ha vacunado, con qué vacuna y cuando administrar la segunda dosis en caso de ser necesaria.

Para implementar dicha aplicación se sabe que hay dos tipos de vacunas, vacunas monodosis que sólo deben administrarse una vez o las vacunas multidosis que deben administrarse en dos dosis.

De todas las vacunas se debe conocer su id (no será necesario comprobar que no haya id repetidos) y su nombre y, además, e las vacunas multidosis se debe saber el tiempo en semanas que debe transcurrir entre la administración de la primera dosis y de la segunda.

Por otra parte debemos registrar en nuestro sistema a los pacientes, de los cuales guardaremos su nombre y apellidos y su teléfono que nos servirá como identificador ya que no podrá haber dos pacientes con el mismo teléfono. Además de cada paciente guardaremos la fecha en la que se puso la primera dosis y la vacuna que se le puso, que será un objeto de la clase vacuna.

En el programa principal se mostrará un menú con las siguientes opciones:

1. Registrar paciente. Se pedirán los datos del paciente, la fecha en la que se suministra la primera dosis de la vacuna (será un String con el formato dd/mm/aaaa) y los datos de la vacuna a suministrar que puede ser monodosis o multidosis.
2. Mostrar pacientes con dosis completa. Mostrará un listado con los pacientes que han recibido una dosis de una vacuna monodosis o aquellos que, habiendo recibido la primera dosis de una vacuna multidosis, ya deban haber recibido la segunda por haberse superado las semanas de su administración desde que se puso la primera. (Se asume que el paciente siempre se pone la segunda dosis en la fecha indicada según la vacuna). Se debe mostrar el nombre del paciente, el teléfono, id de vacuna, nombre de vacuna, fecha de administración de la primera dosis y, en el caso de las multidosis, fecha de administración de la segunda.
3. Mostrar pacientes pendientes de segunda dosis. Muestra un listado con los pacientes que han recibido la primera dosis de una vacuna multidosis pero aún no han transcurrido las semanas necesarias para recibir la segunda. Se debe mostrar el nombre del paciente, el teléfono, id de vacuna, nombre de vacuna, fecha de administración de la primera dosis y fecha en la que se deberá administrar la segunda.
4. Salir del programa.

Antes de cerrar el programa se deberá guardar toda la información en un fichero binario que deberá recuperarse al arrancar nuevamente el programa.

Se deberá controlar todas las posibles excepciones.

Se recomienda usar una clase Fecha que implemente todos los métodos necesarios para realizar las operaciones que hagan falta con las fechas: comprobar si un String tiene una fecha válida, convertir un String a día, mes y año, calcular la fecha de administración de la segunda dosis teniendo en cuenta la fecha de la primera dosis y las semanas que deben transcurrir hasta la segunda, calcular si un año es bisiesto o no para saber los días que tiene febrero,...

Cualquier año divisible por 4 es un año bisiesto a excepción de aquellos que, siendo divisibles por 100, no lo son por 400.

Para obtener la fecha actual podemos usar los siguientes métodos de la clase Calendar

```
Calendar cal= Calendar.getInstance();  
int dia = cal.get(Calendar.DAY_OF_MONTH);  
int mes = cal.get(Calendar.MONTH);  
int year= cal.get(Calendar.YEAR);
```

Para sumar un número determinado de días a una fecha se podrá usar el siguiente código, que podrá adaptarse a las necesidades del alumno:

```
public static String sumaDias(int d,int m,int a, int dias) {  
    Date fecha = Date.valueOf(a+"-"+(m+1)+"-"+d);  
    GregorianCalendar cal = new GregorianCalendar();  
    cal.setTime(fecha);  
    cal.add(Calendar.DATE, dias);  
    int dia = cal.get(Calendar.DAY_OF_MONTH);  
    int mes = cal.get(Calendar.MONTH);  
    int anio = cal.get(Calendar.YEAR);  
  
    return dia+"/"+mes+"/"+anio;  
}
```

Para aprobar el examen será imprescindible que éste no tenga errores de compilación.