

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 nombre y, además, de 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, aunque no será necesario que el programa compruebe que no hay repetidos. 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, si sumamos a la fecha de la primera dosis el número de semanas que deben transcurrir hasta la segunda, nos da una fecha anterior a la fecha actual (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, 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, 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,...

```
package ordinario2021;
```

```
import java.util.ArrayList;
```

```
import java.util.Iterator;
```

```
import java.util.Scanner;
```

```
/**
```

```
 *
```

```
 * @author Profesor
```

```
 */
```

```
public class Ordinario2021 {
```

```
    public static ArrayList<Paciente> pacientes = new ArrayList<Paciente>();
```

```
    public static int leerInt()
```

```
    {
```

```
        Scanner sc = new Scanner(System.in);
```

```
        while (true)
```

```
        {
```

```
            try
```

```
            {
```

```
                return sc.nextInt();
```

```
            }
```

```
            catch (Exception e)
```

```
            {
```

```
                sc.nextLine();
```

```
            }
```

```
        }
```

```
    }
```

```
    public static int menu()
```

```

{
    int op = 0;
    while (op<1 || op > 4)
    {
        System.out.println("1. Registrar paciente");
        System.out.println("2. Mostrar pacientes con dosis completa");
        System.out.println("3. Mostrar pacientes pendientes de segunda dosis");
        System.out.println("4. Salir");
        op = leerInt();
    }
    return op;
}

```

```

public static void registrarPaciente()
{
    Scanner sc = new Scanner(System.in);
    System.out.print("Nombre y apellidos: ");
    String nombre = sc.nextLine();
    System.out.print("Telefono: ");
    String tel = sc.nextLine();
    String fecha="";
    while (!Fecha.valida(fecha))
    {
        System.out.print("Fecha de la primera dosis: ");
        fecha = sc.nextLine();
    }
    System.out.print("Nombre de la vacuna: ");
    String vac = sc.nextLine();
    int tipoVac=0;

    while (tipoVac!=1 && tipoVac!=2)

```

```

{
    System.out.print("Monodosis(1) o Multidosis(2): ");
    tipoVac = leerInt();
}
Vacuna v = null;
if (tipoVac==2)
{
    System.out.print("Número de semanas para la segunda dosis: ");
    int sem = leerInt();
    v = new VMultidosis(vac, sem);
}
else
{
    v = new VMonodosis(vac);
}

pacientes.add(new Paciente(nombre, tel, fecha, v));

}

public static void mostrarDosisCompleta()
{
    Iterator<Paciente> it = pacientes.iterator();
    while (it.hasNext())
    {
        Paciente p = it.next();
        if (p.dosisCompleta())
            System.out.println(p);
    }
}

```

```

public static void mostrarDosisPendiente()
{
    Iterator<Paciente> it = pacientes.iterator();
    while (it.hasNext())
    {
        Paciente p = it.next();
        if (!p.dosisCompleta())
            System.out.println(p);
    }
}

public static void main(String[] args) {

    int op = menu();
    while (op!=4)
    {
        switch (op)
        {
            case 1: registrarPaciente(); break;
            case 2: mostrarDosisCompleta(); break;
            case 3: mostrarDosisPendiente(); break;
        }
        op = menu();
    }
}
}

```

VACUNA

```
package ordinario2021;

/**
 *
 * @author Profesor
 */
public abstract class Vacuna {

    protected String nombre;

    public Vacuna(String n)
    {
        nombre = n;
    }

    public String getNombre()
    {
        return nombre;
    }

    public String toString()
    {
        return nombre;
    }

}
```

VMONODOSIS

```
package ordinario2021;

/**
 *
 * @author Profesor
 */
public class VMonodosis extends Vacuna{

    public VMonodosis(String n)
    {
        super(n);
    }

}
```

VMULIDOSIS

```
package ordinario2021;

/**
 *
 * @author Profesor
 */
public class VMultidosis extends Vacuna{

    private int semanas;

    public VMultidosis(String n, int s)
    {
        super(n);
        semanas = s;
    }
    public int getSemanas()
    {
        return semanas;
    }
}
```

PACIENTE

```
package ordinario2021;

/**
 *
 * @author Profesor
 */
public class Paciente {

    private String nombre;
    private String telefono;
    private String pDosis;
    private Vacuna vacuna;

    public Paciente(String n, String t, String pDosis, Vacuna v)
    {
        nombre = n;
        telefono = t;
        this.pDosis = pDosis;
        vacuna = v;
    }

    public boolean dosisCompleta()
    {
        if (vacuna instanceof VMultidosis)
            return Fecha.fechaAlcanzada(Fecha.sumaSemanas(pDosis,
((VMultidosis)vacuna).getSemanas()));
        else
            return true;
    }

    public String toString()
    {
        String texto = nombre+"\t"+telefono+"\t"+vacuna.getNombre()+"\t(1) "+
            pDosis;
        if (vacuna instanceof VMultidosis)
            texto+="\t (2) " + Fecha.sumaSemanas(pDosis,
((VMultidosis)vacuna).getSemanas());
        return texto;
    }
}
```


FECHA

```
package ordinario2021;

import java.sql.Date;
import java.util.Calendar;
import java.util.GregorianCalendar;

/**
 *
 * @author Profesor
 */
public class Fecha {

    public static boolean esBisiesto(int a)
    {
        return (a%4==0 && !(a%100==0 && a%400!=0));
    }
    public static boolean valida(String f)
    {
        if (f.length()!=10) return false;
        if (f.charAt(2)!='/' || f.charAt(5)!='/') return false;
        try
        {
            int dia = Integer.valueOf(f.substring(0, 2));
            int mes = Integer.valueOf(f.substring(3, 5));
            int anio = Integer.valueOf(f.substring(6));
            if (mes < 1 || mes > 12) return false;
            int[] dias = {31,28,31,30,31,30,31,31,30,31,30,31};
            if (esBisiesto(anio))
                dias[1]++;
            if (dia < 1 || dia > dias[mes-1]) return false;
            return true;
        }
        catch (NumberFormatException e)
        {
            return false;
        }
    }

    public static int getDia(String f)
    {
        if (!valida(f)) return 0;
        else
            return Integer.valueOf(f.substring(0, 2));
    }

    public static int getMes(String f)
    {
        if (!valida(f)) return 0;
        else
            return Integer.valueOf(f.substring(3, 5));
    }

    public static int getAnio(String f)
```

```

    {
        if (!valida(f)) return 0;
        else
            return Integer.valueOf(f.substring(6));
    }

    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<10)?"0":"")+dia+"/"+ ((mes<10)?"0":"")+mes+"/"+anio;
    }

    public static String sumaSemanas(String pDosis,int sem)
    {
        if (!valida(pDosis)) return null;
        return sumaDias(getDia(pDosis),getMes(pDosis),getAnio(pDosis),sem*7);
    }

    public static boolean fechaAlcanzada(String f)
    {
        if (!valida(f)) return false;
        Calendar cal= Calendar.getInstance();
        int diaHoy = cal.get(Calendar.DAY_OF_MONTH);
        int mesHoy = cal.get(Calendar.MONTH)+1;
        int anioHoy= cal.get(Calendar.YEAR);
        int dia = getDia(f);
        int mes = getMes(f);
        int anio = getAnio(f);
        if (anioHoy > anio) return true;
        else if (anioHoy < anio) return false;
        else
        {
            if (mesHoy > mes) return true;
            else if (mesHoy < mes) return false;
            else return (diaHoy>=dia);
        }
    }
}

```