



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ

FACULTAD DE INGENIERÍA DE SISTEMAS
COMPUTACIONALES



LICENCIATURA EN CIBERSEGURIDAD

PROGRAMACION 1

PARCIAL 1

PREPARADO POR

ADRIAN JIMENEZ 4-839-2413

A CONSIDERACIÓN DE: NAPOLEON

IBARRA

GRUPO: 2S3111

FECHA: 1-10-2025

I PARTE.

PROBLEMA 1

```
src > Adria > OneDrive > Documentos > UTp - II Semestre > Programación I > 3. Problema1.java > [ ] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 99 args

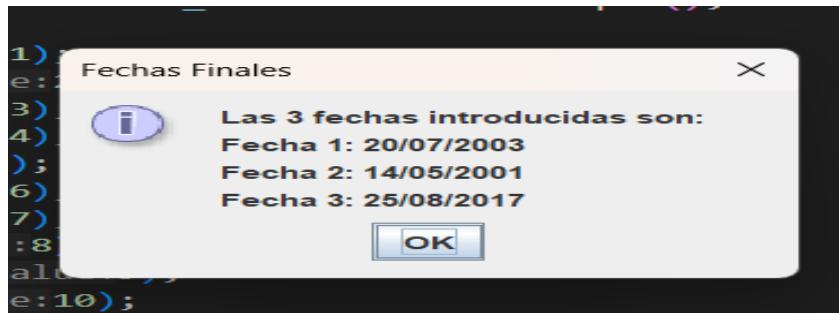
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import javax.swing.JOptionPane;

public class Problema1 {

    private static final Map<String, Integer> MAPA_MESES = new HashMap<>(5);
    static {
        MAPA_MESES.put(key:"enero", value:1);
        MAPA_MESES.put(key:"febrero", value:2);
        MAPA_MESES.put(key:"marzo", value:3);
        MAPA_MESES.put(key:"abril", value:4);
        MAPA_MESES.put(key:"mayo", value:5);
        MAPA_MESES.put(key:"junio", value:6);
        MAPA_MESES.put(key:"julio", value:7);
        MAPA_MESES.put(key:"agosto", value:8);
        MAPA_MESES.put(key:"septiembre", value:9);
        MAPA_MESES.put(key:"octubre", value:10);
        MAPA_MESES.put(key:"noviembre", value:11);
        MAPA_MESES.put(key:"diciembre", value:12);
    }

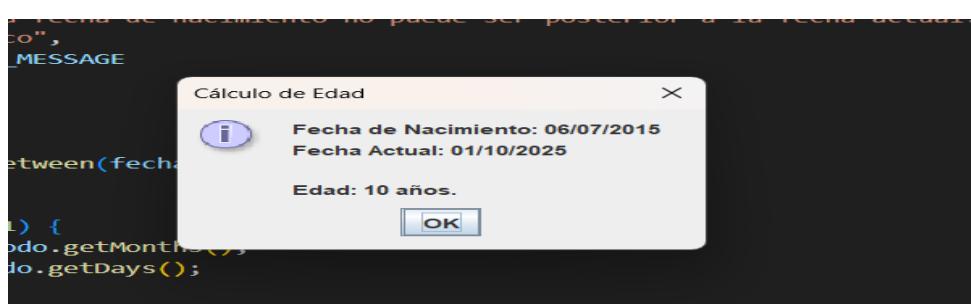
    Run main | Debug main
    public static void main(String[] args) {
        List<String> fechasFormateadas = new ArrayList<>();
        DateTimeFormatter formatoFechas = DateTimeFormatter.ofPattern(pattern:"dd/MM/yyyy");
        Final int NUM_FECHAS = 3;
        for (int i = 0; i < NUM_FECHAS; i++) {
            LocalDate fecha = obtenerFechaValida(" " + i);
            if (fecha== null) {

```



PROBLEMA 2

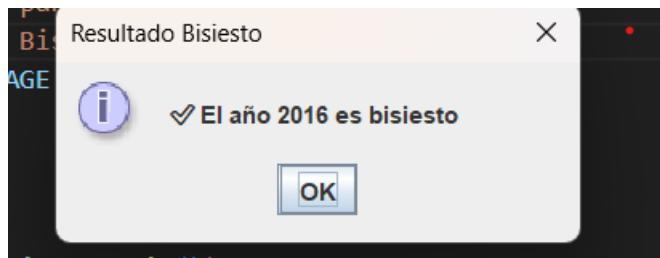
```
1  import java.time.LocalDate;
2  import java.time.Period;
3  import java.time.format.DateTimeFormatter;
4  import java.time.format.DateTimeParseException;
5  import javax.swing.JOptionPane;
6
7  public class Problema2 {
8
9      private static final DateTimeFormatter FORMATO_ENTRADA = DateTimeFormatter.ofPattern(pattern:"dd/MM/yyyy");
10
11     Run main | Debug main
12     public static void main(String[] args) {
13
14         LocalDate fechaNacimiento = solicitarFecha(mensaje:"Introduce fecha de nacimiento:");
15         if (fechaNacimiento == null) return;
16
17         LocalDate fechaActual = solicitarFecha(mensaje:"Introduce fecha actual:");
18         if (fechaActual == null) return;
19
20
21         if (fechaNacimiento.isAfter(fechaActual)) {
22             JOptionPane.showMessageDialog(null,
23                 parentComponent != null,
24                 message:"Error: la fecha de nacimiento no puede ser posterior a la fecha actual.",
25                 title:"Error Lógico",
26                 JOptionPane.ERROR_MESSAGE
27             );
28             return;
29         }
30         Period periodo = Period.between(fechaNacimiento, FechaActual);
31         string resultado;
32         if (periodo.getYears() < 1) {
33             int totalMeses = periodo.getMonths();
34             resultado = "La diferencia es de " + totalMeses + " meses";
35         } else {
36             int totalAños = periodo.getYears();
37             resultado = "La diferencia es de " + totalAños + " años";
38         }
39         JOptionPane.showMessageDialog(null, resultado);
40     }
41
42     private LocalDate solicitarFecha(String mensaje) {
43
44         String fechaString = JOptionPane.showInputDialog(mensaje);
45         LocalDate fecha;
46         try {
47             fecha = FORMATO_ENTRADA.parse(fechaString);
48         } catch (DateTimeParseException e) {
49             JOptionPane.showMessageDialog(null, "Formato incorrecto de fecha.", "Error", JOptionPane.ERROR_MESSAGE);
50             return null;
51         }
52         return fecha;
53     }
54 }
```



PROBLEMA 3

```
C:\> Users > adria > OneDrive > Documents > UTP > II Semestre > Programacion 1 > Problema3.java > La... || ⌂ ⌃ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉ ⌊ ⌋ ⌍ ed Hat > Problema3 > main(String[])
```

```
1 import javax.swing.JOptionPane;
2
3 public class Problema3 {
4
5     Run main|Debug main
6     public static void main(String[] args) {
7         String aniostr;
8         int anio = 0;
9         boolean entradaValida = false;
10
11     while (!entradaValida) {
12         aniostr = JOptionPane.showInputDialog(
13             parentComponent:null,
14             message:"Introduce un año para saber si es bisiesto :",
15             title:"Verificador de Año Bisiesto",
16             JOptionPane.QUESTION_MESSAGE
17         );
18         if (aniostr == null) return;
19         try {
20
21             anio = Integer.parseInt(aniostr.trim());
22             entradaValida = true;
23
24             boolean esBisiesto = esAnioBisiesto(anio);
25
26             String resultado;
27             if (esBisiesto) {
28                 resultado = "✓ El año " + anio + " es bisiesto";
29             } else {
30                 resultado = "✗ El año " + anio + " no es bisiesto.";
31             }
32         }
33     }
34 }
```



PROBLEMA 4

```
C:\> Users > adria > OneDrive > Documents > UTP > II Semestre > Programacion 1 > Problema4.java > La... || ⌂ ⌃ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉ ⌊ ⌋ ⌍ ed Hat > Problema4 > main(String[])
```

```
1 import javax.swing.JOptionPane;
2
3 public class Problema4 {
4
5     Run main|Debug main
6     public static void main(String[] args) {
7         String horasStr, tasaStr;
8         double horasTrabajadas = 0;
9         double tasaOrdinaria = 0;
10        boolean entradaValida = false;
11        while (!entradaValida) {
12            horasStr = JOptionPane.showInputDialog(parentComponent:null, message:"Introduce las HORAS semanales trabajadas:", title:"Entrada de Datos");
13            if (horasStr == null) return;
14
15            tasaStr = JOptionPane.showInputDialog(parentComponent:null, message:"Introduce la TASA horaria ordinaria (ej: 10.50 Balboas):", title:"Entrada de Datos");
16            if (tasaStr == null) return;
17
18            try {
19
20                horasTrabajadas = Double.parseDouble(horasStr);
21                tasaOrdinaria = Double.parseDouble(tasaStr);
22
23                if (horasTrabajadas < 0 || tasaOrdinaria < 0) {
24                    JOptionPane.showMessageDialog(parentComponent:null, message:"Las horas y la tasa deben ser valores positivos válidos.", title:"Cálculo de Salario Semanal");
25                    continue;
26                }
27                entradaValida = true;
28            } catch (NumberFormatException e) {
29                JOptionPane.showMessageDialog(parentComponent:null, message:"Error: Debes introducir valores numéricos válidos para horas y tasa.", title:"Cálculo de Salario Semanal");
30            }
31        }
32    }
33 }
```

