

The screenshot shows the Eclipse IDE interface. The top editor window displays the file 'PracticaDepuracion2.java' with the following Java code:

```
1 package practicaDepuracion2;
2 import java.util.Scanner;
3
4 public class PracticaDepuracion2 {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args)
10    {
11        int dd, mm, aa;
12        int dds, mms, aas;
13        int dias = 0;
14        Scanner sc = new Scanner(System.in);
15        System.out.print("Introduzca Día: ");
16        dd = sc.nextInt();
17        System.out.print("Introduzca Mes: ");
18        mm = sc.nextInt();
19        System.out.print("Introduzca Año: ");
20        aa = sc.nextInt();
21        dds = dd + 1;
22        mms = mm;
```

Below the editor, the 'Console' tab is active, showing the output of the program:

```
PracticaDepuracion2 [Java Application] C:\Program Files\Java\jre1.8.0_111\
Introduzca Día: 2
Introduzca Mes: 3
Introduzca Año: 2010

Fecha: 3//3//2010
```

Practica unidad 3

DEPURACION EN ECLIPSE IDE

Emiliano Montesdeoca | 1ºDAW-B |
17/10/2016

Índice

- 1) Introducción.
- 2) Enunciado.
- 3) Código.
- 4) Depuración.
 - a) Línea a línea.
 - b) Cambio de valores en depuración
 - c) Punto de interrupción

Introducción

A continuación, se desarrollará la práctica de depuración de la práctica número 3 que se basa en calcular el día siguiente introduciendo una fecha en modo comando. El estudio de la depuración se realiza en el entorno de desarrollo Eclipse y el lenguaje utilizado para la práctica es Java.

Cabe destacar que la versión utilizada para esta práctica del entorno de desarrollo es una superior y puede parecer diferente estéticamente.

Enunciado.

Enunciado citado desde el documento entregado por el profesor:

1. Con el programa que teníamos en C# Diamañana, hacerlo en Java para hacer una depuración.
2. Primero probar una ejecución normal-
3. Después mediante F11 probamos la depuración línea a línea.
4. En la Ventana de Locales vemos las variables y los contenidos que van teniendo.
5. Después de la anterior opción, cambiamos el valor de alguna de las variables en la ventana de Locales o la de Inspección.
6. Comprobar que el resultado se ajusta al nuevo valor de la variable.
7. Poner un punto de interrupción y continuar la ejecución desde ahí.

A continuación se realizara un estudio del código y de cómo funciona parte por parte una vez introducido los valores que se piden.

Código.

```
package practicadepuracion2;
import java.util.Scanner;

public class PracticaDepuracion2 {
    public static void main(String[] args)
    {
        int dd, mm, aa;
        int dds, mms, aas;
        int dias = 0;
        Scanner sc = new Scanner(System.in);
        System.out.print("Introduzca Día: ");
        dd = sc.nextInt();
        System.out.print("Introduzca Mes: ");
        mm = sc.nextInt();
        System.out.print("Introduzca Año: ");
        aa = sc.nextInt();
        dds = dd + 1;
        mms = mm;
        aas = aa;
        switch (mm)
        {
            case 1:
            case 3:
            case 5:
            case 7:
            case 8:
            case 10:
            case 12:
                dias = 31;
                break;
            case 2:
                if ((aa % 4 == 0) && (aa % 100 != 0) || (aa % 400 == 0))
                    dias = 29;
                else
                    dias = 28;
                break;
            case 4:
            case 6:
            case 9:
            case 11:
                dias = 30;
                break;
        }
        if (dds > dias)
        {
            dds = 1;
            if (mms == 12)
            {
                mms = 1;
                aas++;
            }
        }
    }
}
```

```
        else
            mms++;
    }
    System.out.print("\n\n\n");
    System.out.print("Fecha: ");
    System.out.print(dds + "/" + mms + "/" + aas);
}
}
```

Código en funcionamiento.

Una vez iniciada el programa nos pide los valores del día, mes y año que queramos.

```
Introduzca Día: 10
Introduzca Mes: 12
Introduzca Año: 1995
```

Una vez introducido esos valores, procede a calcular el siguiente día al que introducimos.

```
Introduzca Día: 10
Introduzca Mes: 12
Introduzca Año: 1995
|
Fecha: 11//12//1995
```

El programa también está encargado de calcular si el día siguiente significaría un cambio de mes:

```
Introduzca Día: 31
Introduzca Mes: 1
Introduzca Año: 2010
|
Fecha: 1//2//2010
```

También realiza el cálculo con los años:

```
Introduzca Día: 31
Introduzca Mes: 12
Introduzca Año: 2010
|
Fecha: 1//1//2011
```

Y por último se encarga de calcular si el año es bisiesto o no:

Introduzca Día: 28	Introduzca Día: 28
Introduzca Mes: 2	Introduzca Mes: 2
Introduzca Año: 2010	Introduzca Año: 1988
Fecha: 1//3//2010	Fecha: 29//2//1988

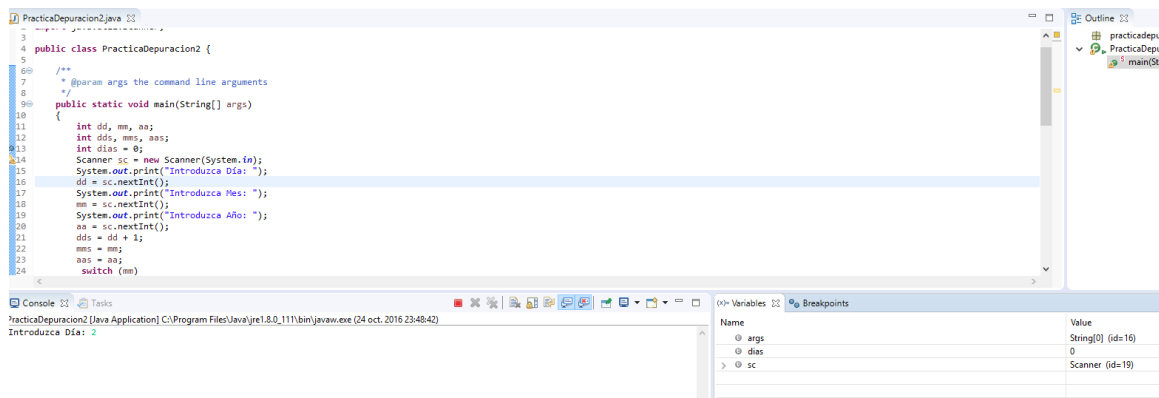
Depuración

Depuración línea a línea

La depuración línea a línea consiste en poder ver cómo funciona el programa paso a paso, así podemos detectar fácilmente qué parte no funciona de nuestro código.

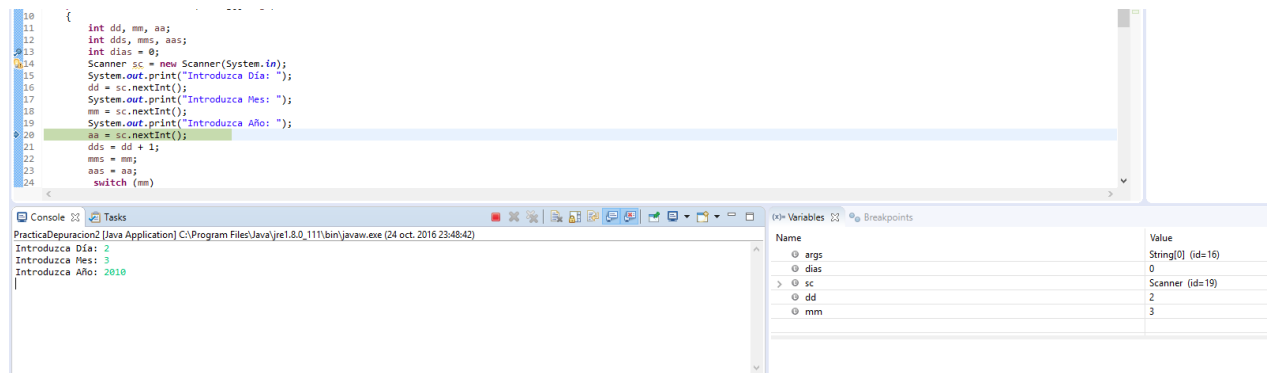
Para activar la depuración línea a línea obligatoriamente hay que agregar un punto de interrupción, y se hace dando clic derecho en el lado azul, en la línea que queremos y agregar un punto de interrupción.

Desde ahí pulsamos F5 e iremos línea a línea.

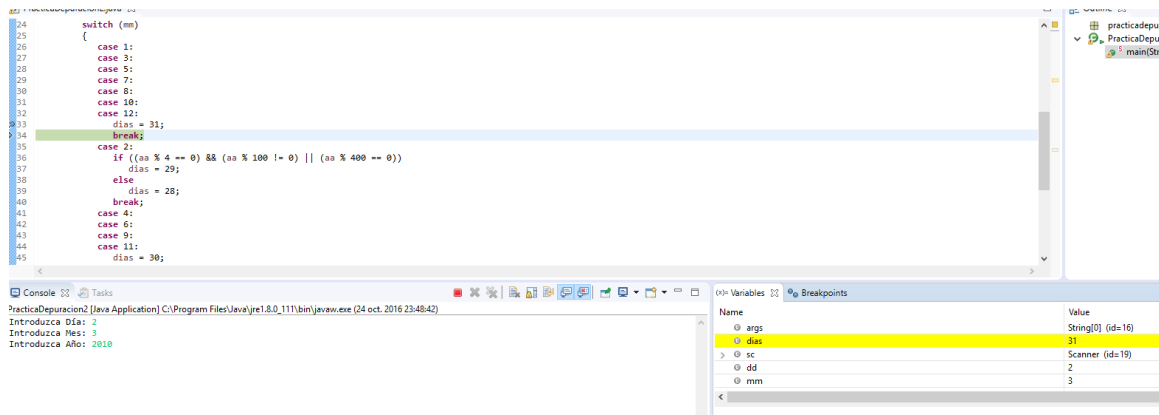


Aquí podemos ver como se encuentra a la espera de recibir un numero por el usuario utilizando el teclado.

Una vez introducido los valores podemos ver como el gestor de depuración, en las que se encuentras las variables y su valor, el cual introducimos anteriormente.



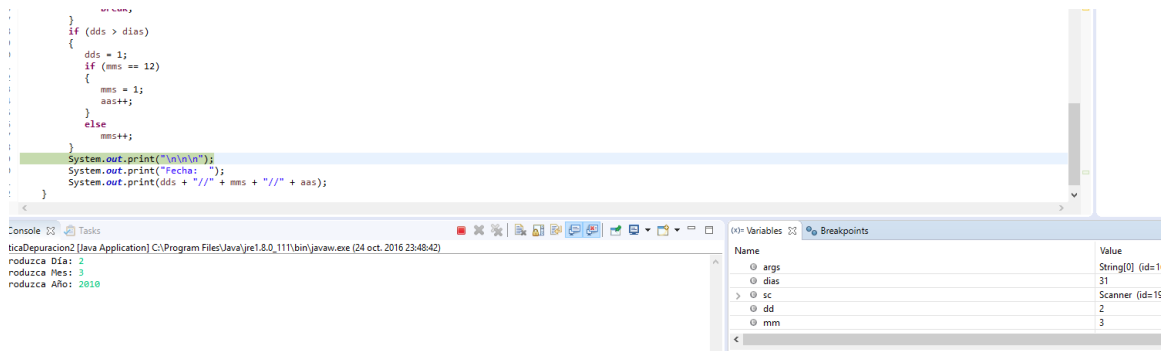
Más adelante en el programa empieza a cargar la funcion donde decide cuantos días tiene el mes introducido:



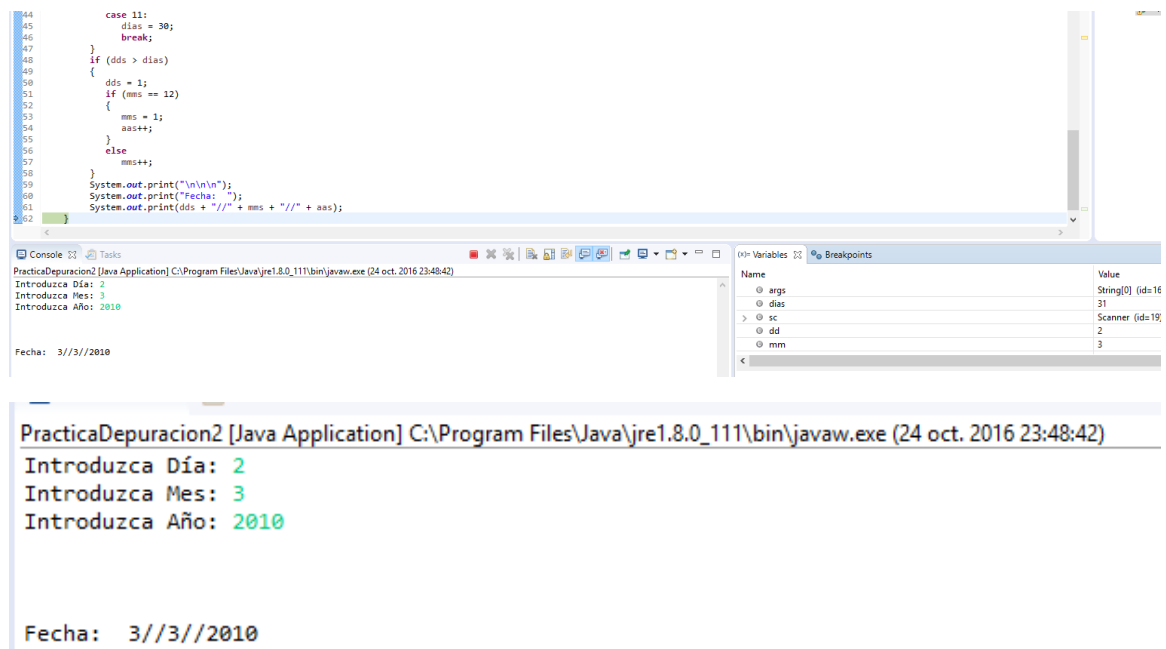
El valor que introducimos fue 3, es decir, marzo, por lo que ahora tiene que ir a cargar la cantidad de días que tiene ese mes. Como se puede ver mientras hacemos la depuración paso a paso, entra en el Switch y carga el mes 3 con 31 días.

(x)= Variables Breakpoints	
Name	Value
args	String[] (id=16)
dias	31
sc	Scanner (id=19)
dd	2
mm	3

Después en el programa se encarga de calcular el siguiente año.

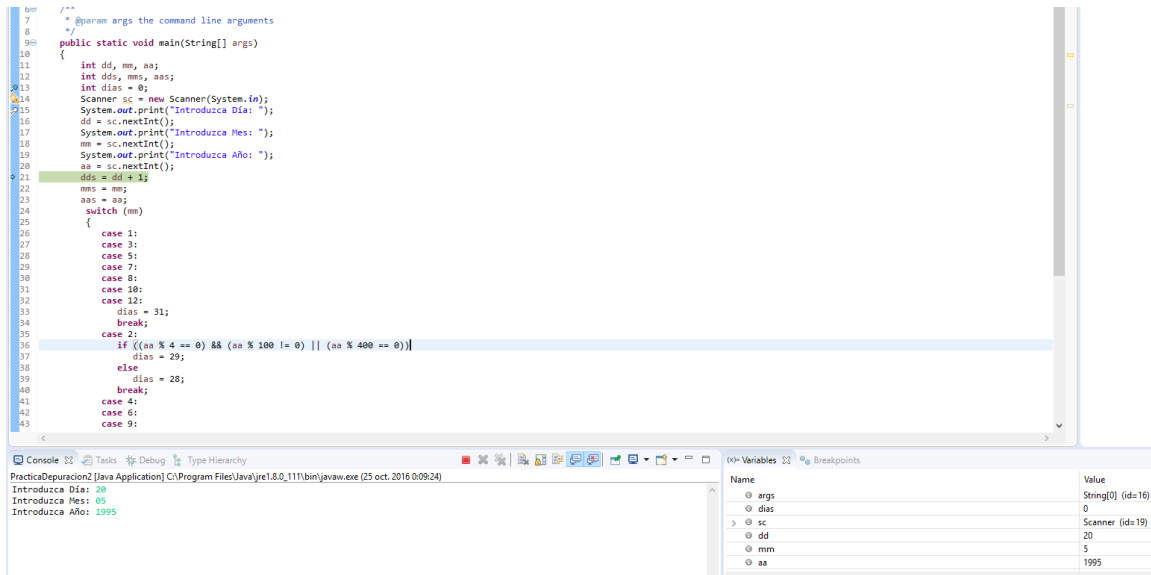


Y por último enseña los valores:

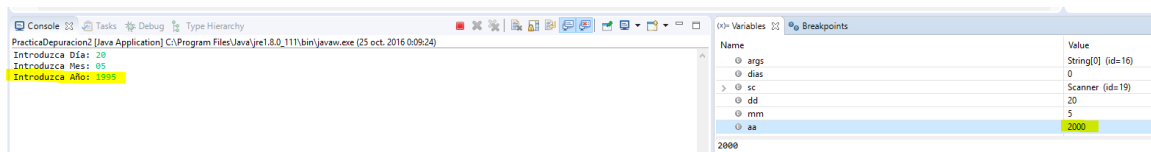


Cambio de valores en depuración

Una de las ventajas de tener la depuración línea a línea, es que podemos cambiar los valores mientras se ejecuta el programa, para poder hacer pruebas, en la siguiente prueba introduciremos una fecha 21/05/1995, y antes de acabar el programa cambiaremos lo que va a enseñar.



Aquí ya hemos introducido nuestra fecha, por lo que vamos a cambiar le valor de 1995 por 2000.



Cuando la variable esta en rojo significa que se ha modificado su valor. Una vez finalizado el programa, debería de mostrar el año 2000 aunque hayamos introducido 1995.

```
Introduzca Día: 20
Introduzca Mes: 05
Introduzca Año: 1995
```

```
Fecha: 21//5//2000
```

Punto de interrupción

A parte de poder ejecutar el programa línea a línea para poder hacer pruebas y buscar errores, también podemos parar el programa en un determinado punto, por ejemplo, si queremos para el programa justo cuando vaya a ejecutar la parte donde decide cuando días tiene el mes introducido, quedaría así:

```
12      int dds, mms, aas;
13      int dias = 0;
14      Scanner sc = new Scanner(System.in);
15      System.out.print("Introduzca Día: ");
16      dd = sc.nextInt();
17      System.out.print("Introduzca Mes: ");
18      mm = sc.nextInt();
19      System.out.print("Introduzca Año: ");
20      aa = sc.nextInt();
21      dds = dd + 1;
22      mms = mm;
23      aas = aa;
24      switch (mm)
25      {
26          case 1:
27          case 3:
28          case 5:
29          case 7:
30          case 8:
31          case 10:
32          case 12:
33          dias = 31;
```

Para activar el punto de interrupción, lo que hay que hacer es darle clic derecho en la línea que queramos y pulsar en agregar un punto de interrupción, y desde ahí apretar F5, para poder seguir en adelante línea a línea.