



Universidad Nacional del Litoral
Facultad de Ingeniería y Ciencias Hídricas
Departamento de Informática

Tecnicatura en Informática Aplicada al Diseño Multimedia y Sitios Web

Elementos de Programación

Unidad 2: JDK y NetBeans

Docente: Ing. Risso, Oscar Luis
Año: 2024

Índice

1. ¿Qué es JDK y NetBeans?	3
1.1. Instalación	4
1.2. NetBeans	4
1.2.1. Entorno de Desarrollo de NetBeans I.D.E.	5
1.3. Pasos para la creación y ejecución de programas Java	5
2. Comencemos a Programar	6
2.1. Ejemplo 2.1	6
2.2. Ejemplo 2.2	10
3. Resumen	12

Unidad 2: JDK y NetBeans

En este Tema

En esta unidad temática aprenderemos a instalar y utilizar NetBeans IDE con Java, y las herramientas básicas para el diseño de interfaces de usuario.

Para comenzar estudiaremos las partes más importantes que el entorno de NetBeans IDE provee para el desarrollo de aplicaciones del tipo de interfaz orientada a objetos.

Sin embargo al ser Java un lenguaje fuertemente orientado a objetos tendremos que pasar por alto algunos temas, conceptos y palabras propios de este paradigma ya que por ahora explicarlos solo contribuiría a confundir. En la unidad 4 estudiaremos en detalle el Paradigma de Programación Orientada a Objetos.-

En esta unidad temática haremos uso de las APIs AWT (Abstract Windows Toolkits) y la Biblioteca de Clases Swing, cabe destacar que Java puede operar también en modo consola de caracteres.

Tendremos una aproximación al lenguaje de programación JAVA. Desarrollaremos una serie de ejemplos a través de los cuales conoceremos su sintaxis y semántica.

Objetivos

Al finalizar este tema Ud. debe lograr:

- Conocer los principales elementos de NetBeans IDE y la filosofía de trabajo de esta herramienta de programación en Java.
- Crear aplicaciones o proyectos Java sencillos.
- Comprender la organización de un proyecto Java y los archivos empleados.

1. ¿Qué es JDK y NetBeans?

■ J.D.K. y NetBeans

JDK (Java Development Kit) es el kit de desarrollo de Java una poderosa herramienta de utilidades, documentación y ejemplos para desarrollar aplicaciones Java. NetBeans IDE es la interfase de desarrollo gratuita y de código abierto para desarrollo de software, cuenta con todas las herramientas para crear aplicaciones ya sea en Java, C++, incluso Ruby. Esta ha sido desarrollada para varias plataformas como Linux, MacOS, X, Solaris y también Windows.

■ Objetos

Como estudiaremos más adelante, Java es un lenguaje fuertemente orientado a objetos y consecuentemente permite crear aplicaciones que responden al modelo o paradigma de la programación orientada a objetos (POO).

■ Programación Visual

NetBeans IDE puede utilizar también el modelo de la programación visual, es decir, veremos que podemos diseñar partes muy importantes de nuestra aplicación operando con el mouse. Así colocaremos objetos gráficos en la ventana o formulario del programa sin escribir nada pues ¡el código lo escribe NetBeans IDE automáticamente!.

■ Eventos

Como en cualquier aplicación de ventanas, estos componentes visuales (botones, iconos, menús, listboxes, etc) responden a eventos o acciones que debemos programar.

■ Lenguaje Java

NetBeans IDE utiliza como lenguaje de programación Java (SE, en nuestro caso), un potente y elegante lenguaje de alto nivel y orientado a objetos que nos permitirá codificar algoritmos fácilmente, desde los más simples hasta casos muy complejos.

1.1. Instalación

Debido a la cantidad y variedad de sistemas operativos utilizados no podemos realizar un paso a paso de la instalación para cada uno. A continuación se dejan los links oficiales para descargar las herramientas y buscar la documentación necesaria para la instalación.

Links para descargar:

- <https://www.java.com/es/>
- <https://www.oracle.com/ar/java/technologies/downloads/>
- <https://netbeans.apache.org/download/index.html>

1.2. NetBeans

Aclaración

Las imágenes y algunos conceptos de esta sección pueden haberse modificado en sucesivas versiones de la herramienta. Son utilizadas de manera ilustrativa.

NetBeans es un proyecto exitoso de código abierto con una gran base de usuarios, una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio 2000 y continúa siendo el patrocinador principal de los proyectos.

Hoy hay disponibles dos productos: el NetBeans IDE y NetBeans Platform. NetBeans IDE es un entorno de desarrollo - una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

También está disponible NetBeans Platform; una base modular y extensible usada como estructura de integración para crear grandes aplicaciones de escritorio. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones.

Ambos productos son de código abierto y gratuitos para uso tanto comercial como no comercial. El código fuente está disponible para su reutilización de acuerdo con la Common Development and Distribution License (CDDL) v1.0 and the GNU General Public License (GPL) v2.

1.2.1. Entorno de Desarrollo de NetBeans I.D.E.

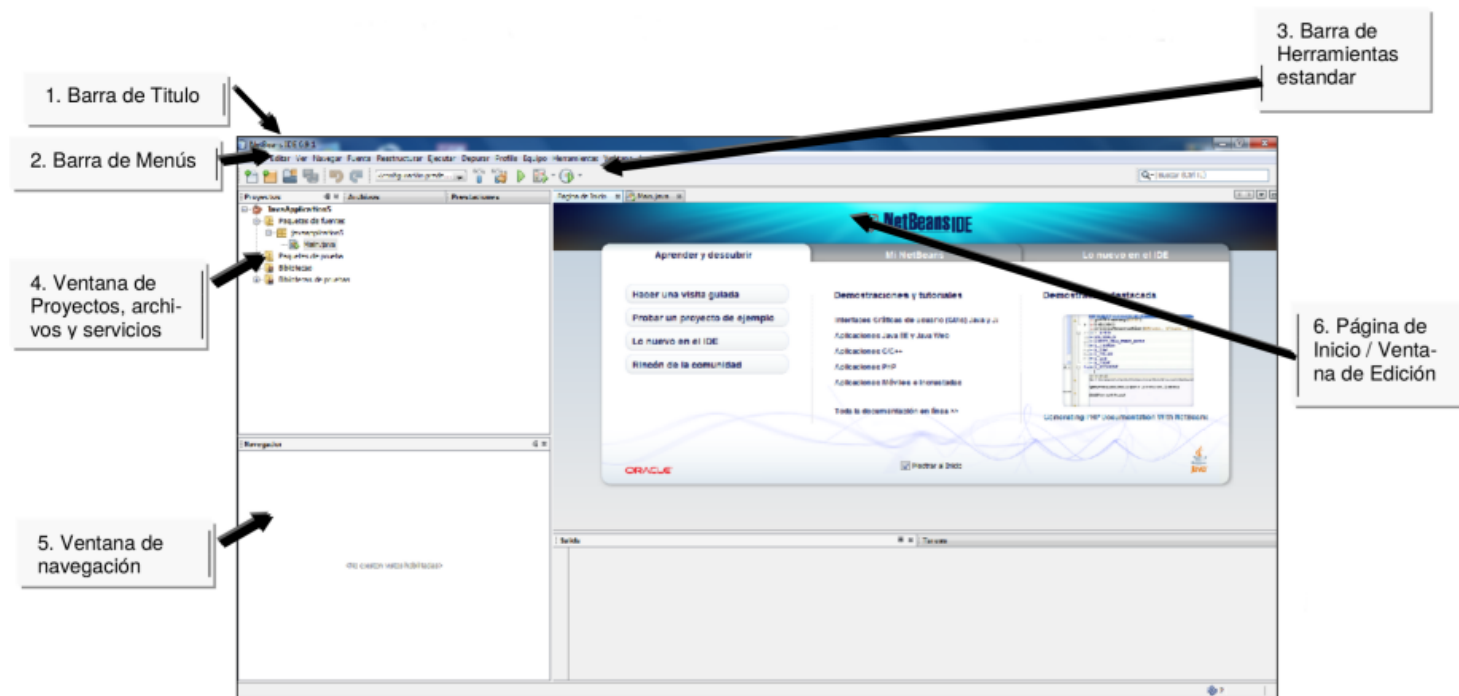


Figura 1: NetBeans IDE

1. **Barra de Título:** Esta situada en la parte superior de la ventana y contiene el título de la aplicación o programa y a la derecha contiene los botones de control de ventana minimizar, maximizar/restaurar y cerrar.
2. **Barra de Menús:** En esta área aparecen los menús disponibles, cada menú contiene una lista de opciones específicas dependientes del nombre del menú.
3. **Barra de Herramientas estándar:** esta contiene botones asociados a las operaciones mas comunes que se encuentran en los menús.
4. **Ventana de Proyectos, Archivos y Servicios:** Es la ventana que contiene el objeto activo del proyecto, los paquetes y las librerías, además están los servicios que se desean utilizar.
5. **Ventana de Navegación:** nos permite visualizar los objetos que contiene el proyecto actual, agrupados por categorías.
6. **Ventana de edición:** aquí se va a realizar la codificación de los proyectos, el diseño de formularios, etc.

1.3. Pasos para la creación y ejecución de programas Java

1. **Primero Edición:** Se crea un programa con ayuda de un editor de textos o una IDE. Se almacena el programa con extensión java. En nuestro caso usaremos NetBeans IDE.
2. **Segundo Compilación:** El compilador lee el código Java del fichero .java, si se detectan errores sintácticos el compilador avisa donde y de que tipo son. Se generan los bytecodes que se almacenan en ficheros tipo .class

3. **Tercero Cargador de Clases:** El cargador de clases lee los ficheros los bytecodes (ficheros .class) que están en el disco y los almacena en la memoria de la computadora.
4. **Cuarto Verificador de bytecodes:** El verificador de bytecodes comprueba que sean validos y no violan la seguridad de la máquina virtual java (JVM).
5. **Quinto Interpretes de bytecodes o compilador JIT:** La máquina virtual Java (JVM) lee los bytecodes y los traduce al idioma que el ordenador comprende, código máquina.

NOTA

A medida que se ejecuta el programa se hace uso de la memoria principal para almacenar los datos con los que trabaja el programa.

2. Comencemos a Programar

2.1. Ejemplo 2.1

Problema: Crear un proyecto Java cuya interfaz de usuario muestre el mensaje “Hola Mundo!”

Solución

Video en YouTube, click en el nombre -> JAVA U01 Ejemplo01
URL del video -> https://youtu.be/5-syjV43v_A

1. Primero debemos crear nuestro nuevo proyecto desde el menú Archivo / Proyecto Nuevo (Ctrl+Mayusculas+N), como lo muestra la Figura 2

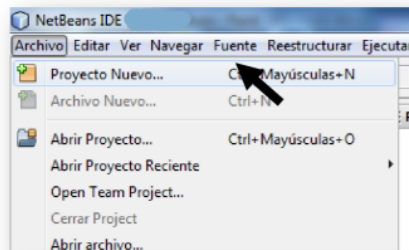


Figura 2: Menú Archivo

O también podemos utilizar la Barra de Herramientas Estándar haciendo click en el botón que se muestra en el círculo rojo (Figura 3)

Obtendremos la siguiente pantalla, que nos permite elegir entre distintos tipos de Proyectos: Aplicación Java – Aplicación de Escritorio Java – Biblioteca de clases Java – Proyectos Java con Fuentes Existentes – Proyecto Java Free - Form.

Seleccionamos la primer opción “**Aplicación Java**” **Crear una nueva aplicación Java SE** en un proyecto estándar del IDE. También puedes generar una clase principal en el proyecto. Proyecto estándar usa un script generado por el IDE para construir, ejecutar, y depurar el proyecto.

2. Ahora hacemos click en el botón Siguiente y pasamos a esta pantalla (Figura 5)

- **Nombre proyecto:** en este campo escribimos el nombre que le queremos dar al proyecto. “U2.Ejemplo1”

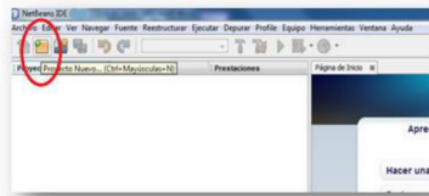


Figura 3: Botón Crear Proyecto Nuevo

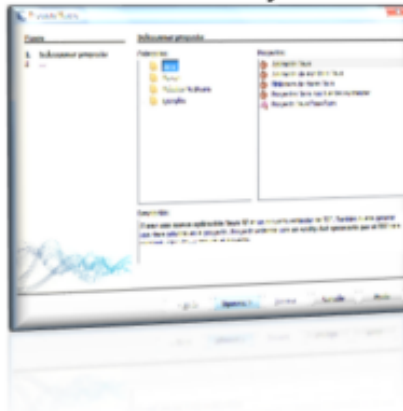


Figura 4: Tipos de Proyectos

- **Ubicación del proyecto:** Aquí seleccionaremos la carpeta en donde guardaremos nuestros proyectos. El botón Examinar nos permite cambiar de carpeta.-
- **Carpeta de Proyecto:** Aquí guardaremos todos los archivos de nuestro proyecto Java.-
- Verifique que estén tildadas las casillas Crear Clase principal y Configurar como proyecto principal

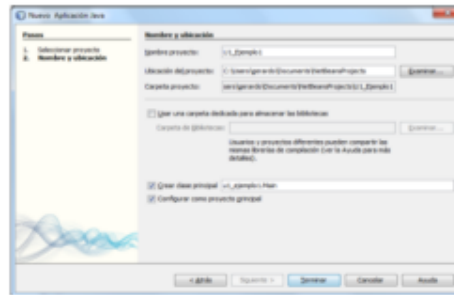


Figura 5: Nombre y ubicación del proyecto

3. Por ultimo hacemos click en el botón Terminar y aparece la siguiente ventana (Figura 6)
Esta es la estructura básica de un programa en Java.

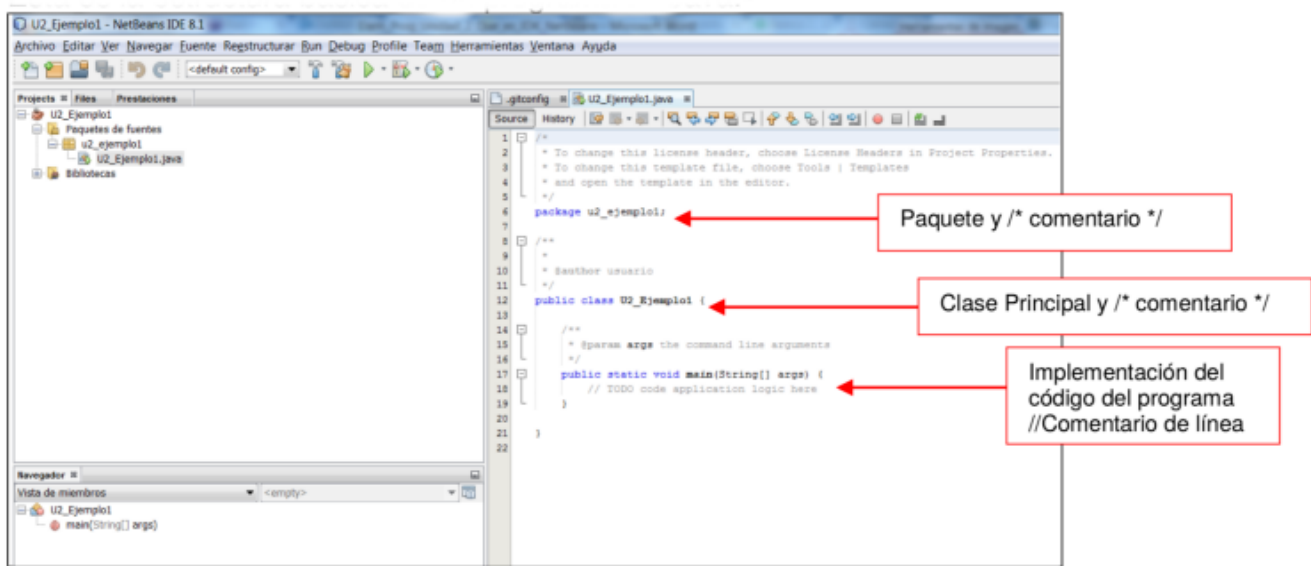


Figura 6: Estructura básica

4. En la ventana de edición de código escribe las siguientes líneas:

```

1  package u2_ejemplo1;
2
3  import javax.swing.JOptionPane;
4
5  /**
6   *
7   * @author usuario
8   */
9
10 public class U2_Ejemplo1 {
11     /**
12      * @param args the command line arguments
13      */
14

```



```

15 public static void main(String[] args) {
16     // TODO code application logic here
17     JOptionPane.showMessageDialog(null,"Hola Mundo!!!");
18 }
19
20 }

```

Código 1: Ejemplo 2.1

Las líneas **import** y **JOptionPane** son las que hay que agregar.

NOTA

Toda línea de código debe acabar con “;” (Punto y Coma)

- Primero pulsamos F9 para compilar nuestro programa o compilamos desde el Menú Ejecutar (Figura 7) Si existen errores en la compilación de nuestro programa podremos ver los errores, ya que la línea errónea se subraya en rojo y el programa jamás se ejecutara hasta que los corrijamos. De lo contrario seguimos con el próximo paso

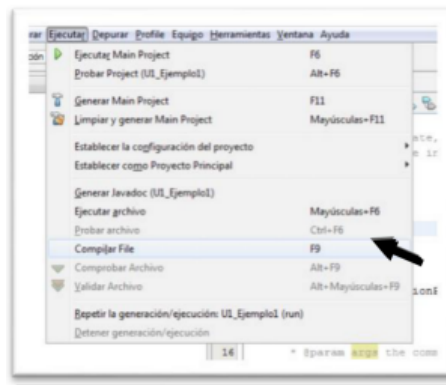


Figura 7: Menú Ejecutar / Compilar F9

Cuando el código está libre de errores pulsamos la tecla F6 Run, también se puede hacer desde el Menú Run (Figura 8)

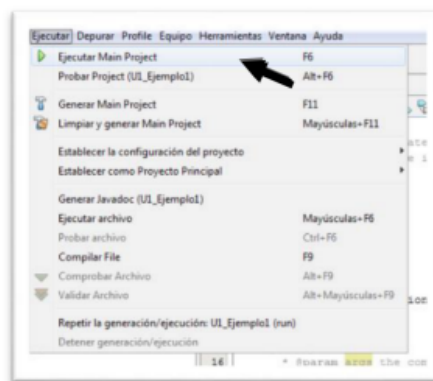


Figura 8: Menú Ejecutar / Run “F6”

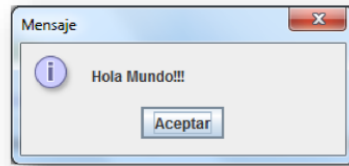


Figura 9: Hola Mundo

- Finalmente aparecerá en pantalla la siguiente ventana (Figura 9), haga click en “Aceptar”

NOTA

import javax.swing.JOptionPane

import es una instrucción de código que permite importar clases y estas puedan ser utilizadas dentro del programa.

javax.swing.JOptionPane

Llamada al objeto JOptionPane que sirve para representar un cuadro de diálogo (dialog box) con varios propósitos:

- Mostrar mensajes a través de ShowMessageDialog.
- Preguntar por la confirmación del usuario ShowConfirmDialog.
- Obtener datos de entrada ingresados por el usuario ShowInputDialog.
- La combinación de los tres anteriores en ShowOptionDialog.

JOptionPane.showMessageDialog(null, "Hola Mundo!!!");

Se informa al usuario el mensaje “Hola Mundo!!!” por intermedio de JOptionPane para ello se utiliza la propiedad

ShowMessageDialog(1er parámetro, 2do parámetro)

1ro. null: Significa que el cuadro de diálogo se mostrará en el centro de la pantalla, 2do se escribe el mensaje entre comillas.

2.2. Ejemplo 2.2

Problema: Crear un proyecto Java que permita ingresar su nombre y muestre un mensaje de bienvenida con este nombre.

Solución

Video en YouTube, click en el nombre -> JAVA U02 Ejemplo02

URL del video -> <https://youtu.be/ds26M8u2x8g>

- Repetimos los tres primeros pasos del ejemplo anterior, pero esta vez el nombre del Proyecto será U2.Ejemplo2.
- Copiamos el siguiente código:

```

1  /*
2  * To change this template, choose Tools | Templates
3  * and open the template in the editor.
4  */

```

```

5
6 package u2_ejemplo2;
7 /**
8  *
9  * @author FICH
10 */
11
12 import javax.swing.JOptionPane;
13
14 public class Main {
15     /**
16     * @param args the command line arguments
17     */
18
19     public static void main(String[] args) {
20         // TODO code application logic here
21         String nom;
22         nom = JOptionPane.showInputDialog("Ingrese su Nombre");
23         JOptionPane.showMessageDialog(null, "Bienvenido "+nom);
24     }
25 }
26

```

Código 2: Ejemplo 2.2

Obtendremos el siguiente cuadro de diálogo:

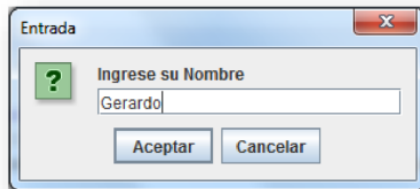


Figura 10: Ingrese Nombre

Hacemos click en el botón Aceptar aparecerá esta ventana.

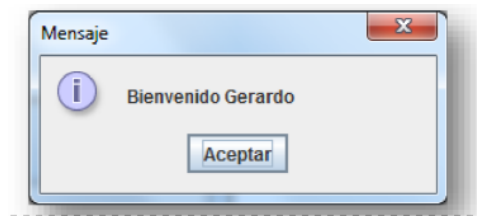


Figura 11: Bienvenido

NOTA

String nom;

String es una clase que guarda un tipo de dato que permite cualquier combinación de caracteres. Y nom es el identificador de una variable, que reserva un lugar de memoria donde guardar los datos.

nom = JOptionPane.showInputDialog("Ingrese su Nombre");

Se asigna por medio del = a la variable **nom** lo que ingresamos en la caja de diálogo.

JOptionPane.showMessageDialog(null,"Bienvenido -nom);

Se muestra como en el ejemplo anterior un mensaje y se agrega (+) el contenido de la variable **nom**.

3. Resumen

1. Java es una herramienta de programación muy poderosa que permite desarrollar aplicaciones rápidamente.
2. Java permite crear programas (llamados proyectos o aplicaciones) basados en el modelo de objetos o diseñados según el paradigma de la programación orientada a objetos.
3. Java el lenguaje de alto nivel empleado por NetBeans IDE.
4. La barra de Herramientas estándar contiene los comandos del menú de NetBeans IDE más usuales.
5. NetBeans IDE posee un editor de textos el cual se usa para crear los archivos con código Java que controlarán nuestra aplicación o proyecto.
6. De acuerdo al proyecto a diseñar, se deben programar algunos elementos, eventos de componentes, para que la aplicación responda cuando se la utilice.