



Actividad | 1 | Estructuras de Control

Nombre del curso

Ingeniería en Desarrollo de Software



TUTOR: Aarón Iván Salazar Macías.

ALUMNO: Uziel de Jesús López Ornelas.

FECHA: 20 de julio del 2025.

Tabla de Contenido

Introducción	1
Descripción	1
Justificación	2
Desarrollo	2
Interfaz.....	2
Codificación.....	6
Conclusión	12
Referencias	13

Introducción

Damos inicio a la materia de Lenguajes de programación IV y con ella nos adentraremos en los diferentes rubros y tópicos que esté nos proporciona, para ello es importante aclarar uno de los puntos más básicos en este ámbito, ¿Qué es un lenguaje de programación?, esta consiste en que nosotros le damos instrucciones a la computadora por medio de líneas de código, organizando la sintaxis, la semántica, las sentencias, entre otros. Estas instrucciones promueven la realización de diversas tareas para resolver problemas que puedan estar ejecutándose en el momento, para tener claro este concepto, supongamos lo siguiente; se tiene que realizar una aplicación que sume dos números, el problema se deriva en que se tiene que buscar una solución que permita crear dicho software en la computadora, para ello la solución es hacer mediante código (respetando la sintaxis y demás elementos importantes) los pasos de manera formal que permita a la computadora leer esas órdenes y comunicarlás para crear la tarea en específico, al crear este código el ordenador sabe lo que tiene que hacer y responde con la ejecución y la suma de dichos números dando como respuesta la solución del problema. Es la manera en la que nos podemos comunicar con la computadora para crear diversas soluciones a diversos problemas.

Descripción

Con conceptos básicos también nos referimos a aquello que nos indica la actividad y que nos da una idea de lo que se trabajara para completarla y estamos hablando de aquel software que se utilizará para ello, pues bien, comencemos para explicar lo que es NetBeans, este programa es conocido también por su nombre de Apache NetBeans se puede adquirir de manera gratuita desde su página oficial y está disponible para varias plataformas según sea el caso que mejor le convenga al usuario final, su launcher es fácil de instalar y no requiere mucha atención si solo se necesitan los pasos necesarios y básicos para acceder y utilizar esta herramienta, es un IDE (Entorno de Desarrollo Integrado) que por supuesto es de código abierto en donde su principal lenguaje de programación que tiene por defecto es el de “Java” pero al igual que otros IDE este soporta otros más haciendo que tenga una gran versatilidad y no solo se emplee para un campo en específico, los más conocidos son los siguientes:

- C
- C++
- PHP

- HTML5
- CSS

NetBeans ofrece muchas ventajas que demuestran lo útil que es este software, ya que presenta puntos y herramientas interesantes que lo hacen una opción viable a la hora de seleccionar este software en específico:

- Permite depurar, escribir y desplegar código.
- Permite crear estructuras de Interfaces gráfica en la que puedes trabajar para realizar aplicaciones de escritorio como de móviles.

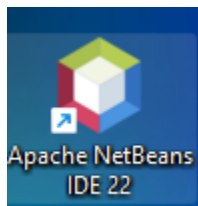
Justificación

¿Por qué es importante establecer estos métodos para solucionar los problemas establecidos en diferentes ámbitos?, esto sin duda nos indica la efectividad y necesidad de aprender de manera empírica aquellos procedimientos necesarios para su comprensión absoluta, por ello es importante mencionar que todo esto es necesario para que un ingeniero conozca las capacidades que tienen estos programas para aprender a manejar todas las funciones desde las más básicas y que no necesitan un conocimiento muy amplio hasta las más complejas que necesitan comprensión y practica seguida para dominar estos rubros y tomarlos como un hábito donde lo más importante es solucionar problemas con las herramientas que se tienen en la mano y que con ellas se ejecutan distintas opciones que pueden apoyar a la realización exitosa de dicha tarea o instrucción. Recordar que estamos trabajando en un IDE, esto nos ayuda en gran medida para desarrollar un software en un mismo sitio, en una misma aplicación con todas las herramientas necesarias para lograr el objetivo y resolver la problemática, todo parte desde el inicio, en base al no saber con exactitud los términos y familiarizarse con ellos para dominar típicos y funciones que faciliten el hábito.

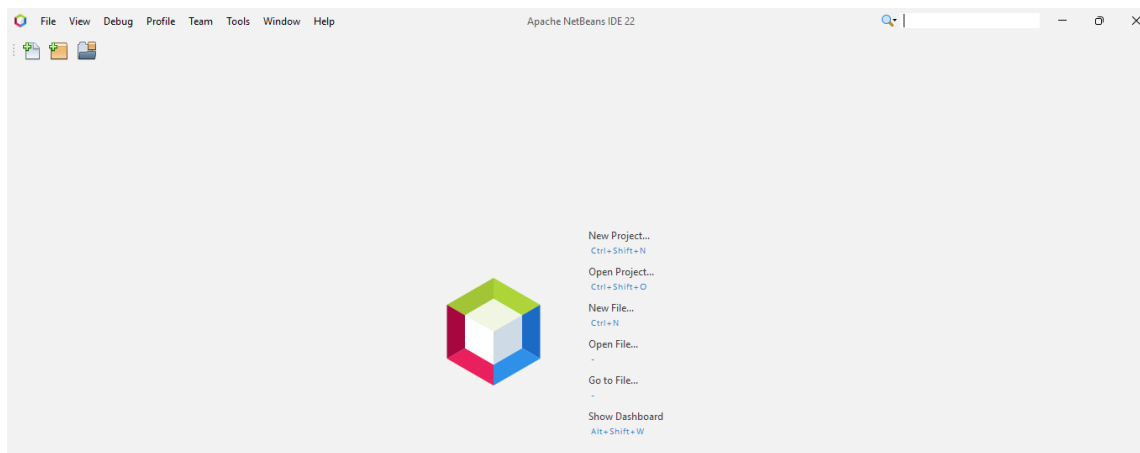
Desarrollo

Interfaz

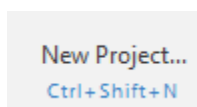
Para iniciar con esta etapa primero abriremos la aplicación de “NetBeans” en nuestro ordenador:



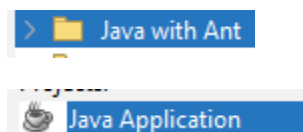
Ingresamos y esperamos a que cargue toda la configuración necesaria en donde se nos abrirá la ventana siguiente:



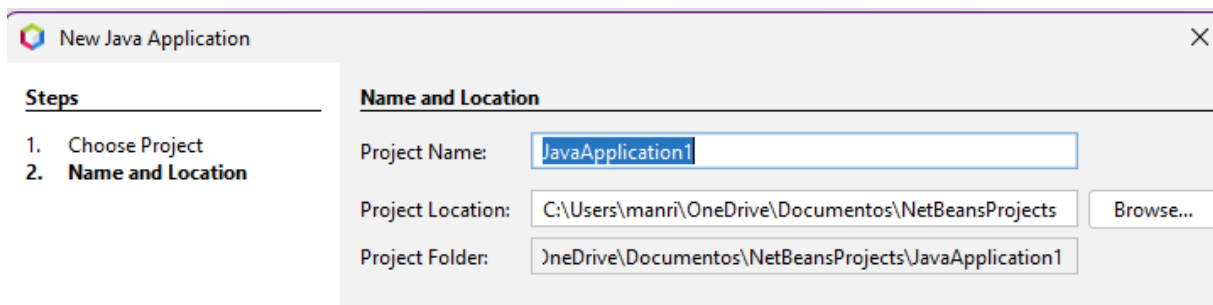
Para iniciar con nuestro proyecto seleccionaremos el siguiente apartado:



Con esto se nos abrirá una nueva ventana donde seleccionaremos apartados importantes para nuestro proyecto, como los siguientes:



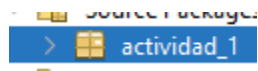
Después de ello daremos en siguiente para continuar, nos muestra una ventana en donde podemos cambiar el nombre, así como las rutas en las que se guardaría nuestros proyecto



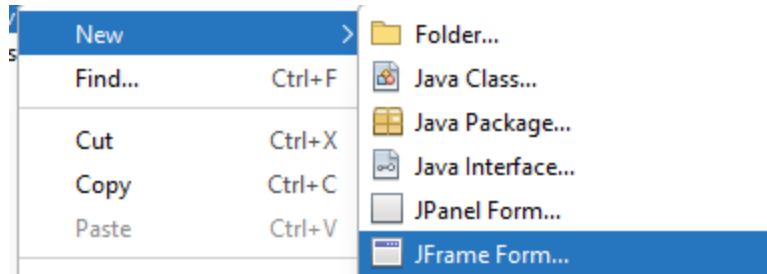
Con esto cambiaremos el nombre de nuestro proyecto:



En el siguiente segmento tenemos varios elementos importantes, por el momento trabajaremos en “actividad_1”:



Daremos clic derecho en ese apartado, seleccionamos “new” seguido de “JFrame Form”:

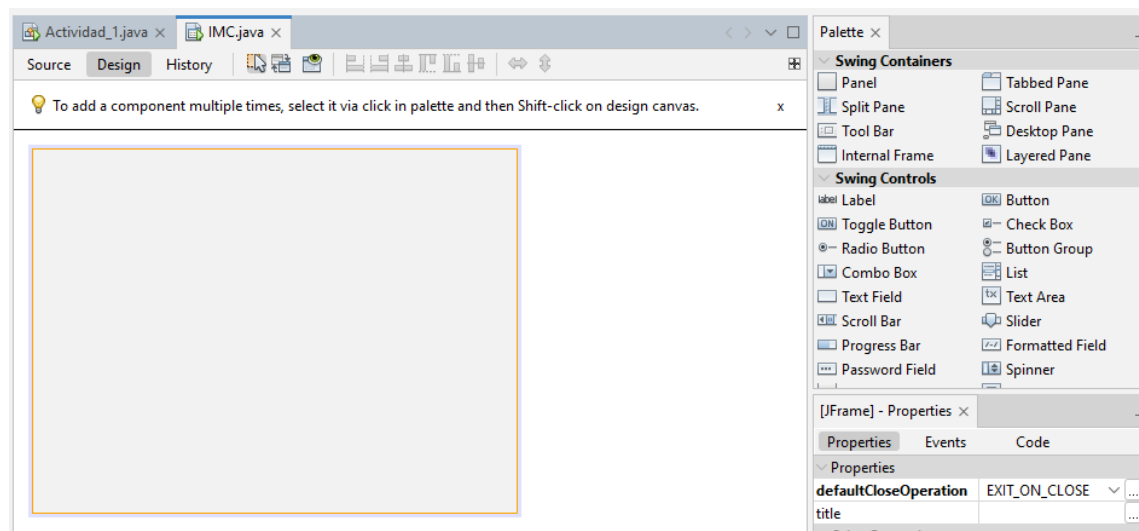


Se nos va a abrir nuevamente otra ventana en la que nombraremos el “JFrame Form” para ubicarlo, como la actividad se nos pide que realicemos una aplicación que determine o calcule el índice de masa corporal, lo llamaremos como “IMC”:

Name and Location

Class Name:

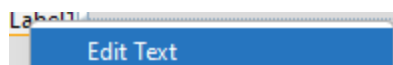
En automático tenemos ahora nuestro lienzo para crear la interfaz de nuestra aplicación con los elementos necesarios:



Para la creación de la interfaz se trabajará con “Label” o etiquetas que nos permitirá darle sentido y estructuración a nuestro contenido:



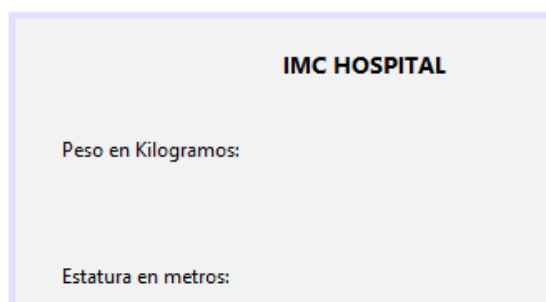
Daremos clic derecho para modificar el texto de la etiqueta y seleccionaremos “Edit Text”:



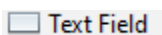
Existen varias maneras de modificar el texto para darle más personalización, por ejemplo, si damos clic derecho en la etiqueta seguido de “propiedades” y a los tres puntos que están en el apartado de “Font” podemos establecer el tamaño e intensidad con la que se quiere mostrar el texto:



Continuaremos agregando elementos para la interfaz:



Es hora de agregar los cuadros de texto, para esto buscamos el apartado de “Text Field”:



Agregamos los elementos a nuestro lienzo:

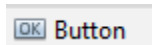


Cuando lo agreguemos tenemos que eliminar el texto que tiene por defecto y nombrar su nueva variable para lograr ubicarlo a la hora que estemos codificando:

New Name:

New Name:

Cuando agreguemos el botón se realizará con el mismo procedimiento, se busca el icono:



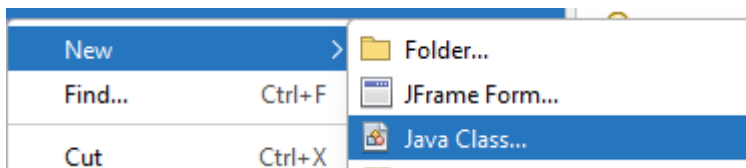
Se agrega y se cambia el nombre y la variable:

New Name:

La interfaz terminada se muestra de la siguiente manera:

Codificación

Ya que tenemos nuestra interfaz lista es momento de hacerla funcional, para ello es necesario crear una clase en el mismo lugar que creamos nuestra interfaz, para ello damos clic derecho y seleccionamos lo siguiente:



Al darle clic nos mandara una ventana en la que estaremos nombrando dicha clase para nuestra codificación:

Lo nombraremos como “Persona” ya que es a ello quien se estará calculando el IMC:

Name and Location

Class Name:

A continuación, creamos nuestra clase:

Finish

Tenemos nuestro lienzo listo para seguir codificando en nuestra clase:

```
package actividad_1;

/**
 *
 * @author manri
 */
public class Persona {

}
```

Lo siguiente que tenemos que hacer es colocar los atributos de dicha clase en base a lo que la actividad nos pide, en este caso será el de “peso” y “estatura”, donde los valores se darán como privados, “float” es utilizado cuando queremos utilizar números con decimales:

```
private float peso;
private float estatura;
```

Después de ello es necesario darle clic derecho y seleccionar nuestras variables en “Insert Code” seguido de “Getter and Setter” y se genera una nueva línea de valores que conforman y construyen nuestros atributos de la clase:

```
private float peso;
private float estatura;

public float getPeso() {
    return peso;
}

public void setPeso(float peso) {
    this.peso = peso;
}

public float getEstatura() {
    return estatura;
}

public void setEstatura(float estatura) {
    this.estatura = estatura;
}
```

Declararemos un método en el que nos genere el IMC de la persona:

```
public float calcularIMC(float p, float e) {
    float imc = p / (e * e);
    return imc;
}
```

Ahora nos dirigiremos al botón de “Calcular para anexar dicha acción de la clase al mismo:

CALCULAR

```
String p = jTextFieldPeso.getText();
String e = jTextFieldEstatura.getText();

float peso = Float.parseFloat(p);
float estatura = Float.parseFloat(e);

Persona persona = new Persona ();
persona.setPeso(peso);
persona.setEstatura(estatura);
persona.calcularIMC();
}
```

Es importante mencionar que tenemos que crear una acción en donde el resultado que coloquemos nos indique si es peso normal, sobrepeso, etc...:

```
public float calcularIMC() {
    float imc = getPeso() / (getEstatura() * getEstatura());

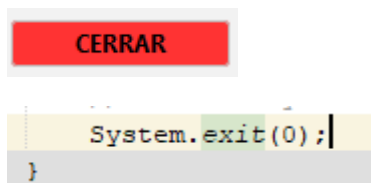
    if (imc < 18.5) {
        JOptionPane.showMessageDialog(null, "Bajo Peso");
    } else if (imc >= 18.5 && imc <= 24.9) {
        JOptionPane.showMessageDialog(null, "Peso Normal");
    } else if (imc >= 25 && imc <= 29.9) {
        JOptionPane.showMessageDialog(null, "Sobrepeso");
    } else if (imc >= 30 && imc <= 34.9) {
        JOptionPane.showMessageDialog(null, "Obesidad I");
    } else if (imc >= 35 && imc <= 39) {
        JOptionPane.showMessageDialog(null, "Obesidad II");
    } else if (imc > 40) {
        JOptionPane.showMessageDialog(null, "Obesidad III");
    }

    return imc;
}
```

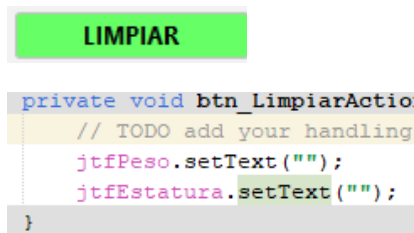
Y agregamos en el botón de “Calcular” un mensaje en el que cuando detecte si es “Peso normal” o “Bajo peso” lo muestre de inmediato:

```
JOptionPane.showMessageDialog(null, "IMC: " + persona.calcularIMC());
```

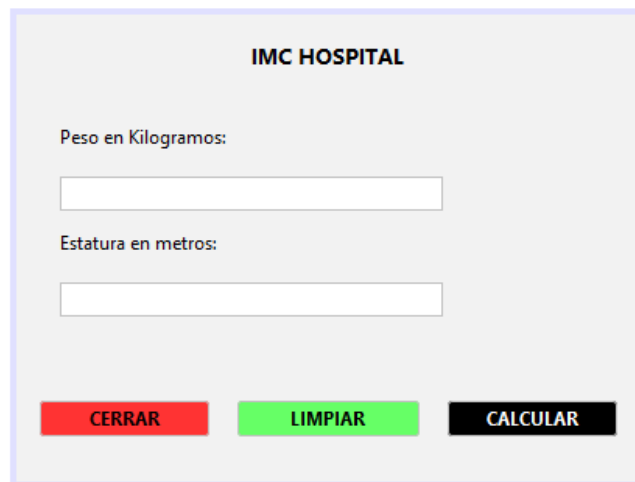
Agregaremos un botón de cerrar para que este termine la aplicación sin necesidad de seleccionar la “x” que normalmente se utiliza:



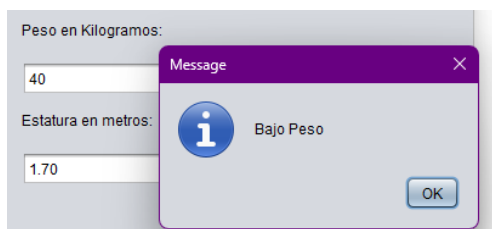
Agregaremos un botón para limpiar el contenido de las celdas:



Con esto nuestra interfaz quedaría de la siguiente manera:

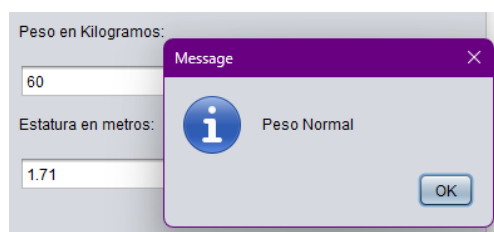


Ahora repasaremos lo que tenemos, para el “Bajo peso” el IMC es menor de 18.5:

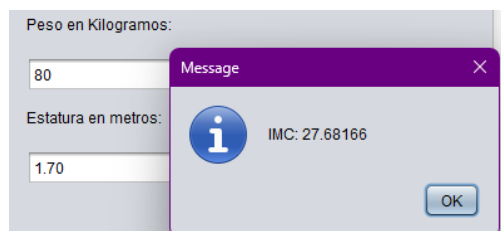
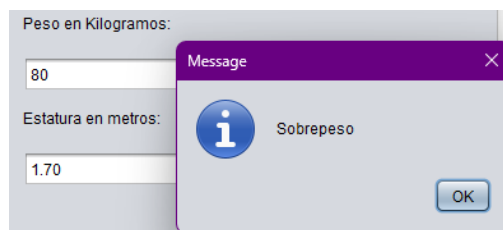




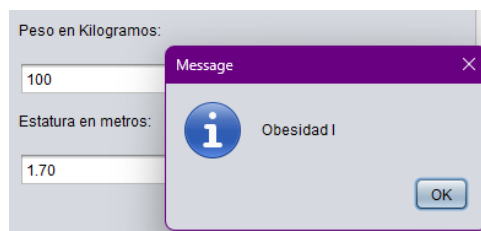
Para el “Peso Normal” el IMC es 18.5 a 24.9:



El siguiente es de “Sobrepeso” que es de 25 a 29.9:



Continuamos con “Obesidad I” que es de 30 a 34.9:




Peso en Kilogramos:

100

Estatura en metros:

1.70

Message

 IMC: 34.602074

OK

“Obesidad II” indica que es 35 a 39.9:


Peso en Kilogramos:

110

Estatura en metros:

1.70

Message

 Obesidad II

OK


Peso en Kilogramos:

110

Estatura en metros:

1.70

Message

 IMC: 38.062283

OK

“Obesidad III” indica que es mayor de 40:


Peso en Kilogramos:

120

Estatura en metros:

1.70

Message

 Obesidad III

OK


Peso en Kilogramos:

120

Estatura en metros:

1.70

Message

 IMC: 41.52249

OK

Conclusión

En base a lo establecido, los métodos y las maneras en las que se resolvió la actividad me doy cuenta que cada lenguaje de programación se relaciona y se diferencia entre unos y otros. Esta actividad nos indicaba crear un software con interfaz que permitiera calcular el IMC de un ser humano de acuerdo con los estándares, específicamente en el lenguaje de programación “Java” donde, utilizando la semántica y sentencias correctas se lograría crear con éxito la actividad y cumplir con los rubros que esta nos pedía. Me recuerda bastante a “Windows Forms” en el lenguaje de programación de “C#”, de la misma manera a través de colocar el diseño (botones, textbox, listbox, etc...), la codificación y concatenar todo de manera correcta podías tener una aplicación con interfaz y no solo una aplicación de consola donde solo se ven las líneas de código. Por ello es importante aprender distintos tipos de lenguaje ya que ayudan a que tengamos un panorama más amplio y total de lo que podemos crear con las herramientas a nuestra disposición.

Link de GitHub

<https://github.com/UZLOP984/Lenguajes-de-Programaci-n-IV.git>

Referencias

¿Qué es un IDE? - Explicación de los entornos de desarrollo integrado - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/ide/>

colaboradores de Wikipedia. (2025, 16 julio). *Lenguaje de programación*. Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n

Monterde, U. M. (s. f.). *Lenguajes de programación*. https://repositorio-uapa.cuaed.unam.mx/repositorio/moodle/pluginfile.php/2655/mod_resource/content/1/UAPA-Lenguajes-Programacion/index.html

López, M. (2022, 13 abril). ¿Qué es NetBeans? Ventajas y usos - IMMUNE Institute. *IMMUNE Technology Institute*. <https://immune.institute/blog/que-es-netbeans/>