



## Practica. Programación con el lenguaje Java

**Actividad del proyecto:** Instalación de herramientas y conceptos básicos.

### Estructura de contenido

- Introducción.
- Objetivos.
- Procedimientos.
- Bibliografía.



## Introducción

La presente practica se realizará procedimientos y ejercicios de programación con el lenguaje de programación Java usando herramientas como NetBeans para la codificación de los programas y ejecución de pruebas.

Realizar las practicas del lenguaje de programación Java es base requerida para ser desarrollador de programas o aplicaciones con uno de los lenguajes de programación más populares. Para el desarrollo del laboratorio es necesario que el estudiante tenga disponible un computador con acceso a internet y el entorno de desarrollo NetBeans.

## Objetivos

Al completar esta práctica, usted podrá:

- ✓ Conocer el concepto básico y características del lenguaje de programación Java.
- ✓ Realizar la instalación de los programas requeridos.
- ✓ Codificar algoritmos secuenciales con el lenguaje de programación Java.



## Procedimiento

Para el desarrollo de las practicas es necesario tener en cuenta los siguientes conceptos.

### Java

¿Qué es la tecnología Java para qué la necesito?

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes.

### Características

- ✓ Orientado a objetos.
- ✓ Robusto.
- ✓ Multiplataforma.
- ✓ Compilado.
- ✓ Tipado dinámico.

### ¿Qué es un IDE o entorno de desarrollo?

Un IDE o Integrated Development Environment, es una aplicación destinada a brindar servicios integrales al desarrollador en su trabajo, o dicho más simple, es un programa que nos permite construir código de una forma más sencilla o didáctica. Ejemplos de IDE importantes en el desarrollo de software actual son Synder, NetBeans, Eclipse, Visual Studio entre otros.

## JDK


JavaDevelopment Kit (JDK) es un software para los desarrolladores de Java. Incluye el intérprete Java, clases Javay herramientas de desarrollo Java(JDT): compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndice y generador de documentación.

El JDK le permite escribir aplicaciones que se desarrollan una sola vez y se ejecutan en cualquier lugar de cualquier máquina virtual Java. Las aplicaciones Javadesarrolladas con el JDK en un sistema se pueden usar en otro sistema sin tener que cambiar ni recompilar el código. Los archivos de clase Javason portables a cualquier máquina virtual Javaestándar.

### ¿Qué es un objeto?

¿qué es un objeto en Java? Un objeto en Javano es ni más ni menos que lo mismo que un objeto en el mundo real.

Los objetos corresponden a cosas que se encuentran en el mundo real. Por ejemplo, un sistema de compra en línea podría tener objetos como: carrito de compras, cliente y producto. Los objetos en Javano tienen existencia física, no puedes tocarlos, pero sí que existe como algo que contiene información sobre sí mismo y sobre su estado. Se puede interactuar con él y se puede modificarlo para que responda a nuestras necesidades de programación.

<p>Ejemplo Objeto perro. Atributos = características Métodos = funciones</p>	 <ul style="list-style-type: none"><li>• <b>Características:</b><ul style="list-style-type: none"><li>○ Raza</li><li>○ Color</li><li>○ Estatura</li><li>○ Edad</li><li>○ Etc.</li></ul></li><li>• <b>Acciones posibles:</b><ul style="list-style-type: none"><li>○ Comer</li><li>○ Dormir</li><li>○ Sentarse</li><li>○ Ladrar</li><li>○ Etc.</li></ul></li></ul>
--	---

## Operadores

### Aritméticos

OPERADOR	DESCRIPCIÓN	USO
+	Realiza Adición entre los operandos	$12 + 3 = 15$
-	Realiza Substracción entre los operandos	$12 - 3 = 9$
*	Realiza Multiplicación entre los operandos	$12 * 3 = 36$
/	Realiza División entre los operandos	$12 / 3 = 4$
%	Realiza un módulo entre los operandos	$16 \% 3 = 1$

### De comparación

OPERADOR	DESCRIPCIÓN	USO
>	Devuelve True si el operador de la izquierda es mayor que el operador de la derecha	$12 > 3$ devuelve True
<	Devuelve True si el operador de la derecha es mayor que el operador de la izquierda	$12 < 3$ devuelve False
==	Devuelve True si ambos operandos son iguales	$12 == 3$ devuelve False
>=	Devuelve True si el operador de la izquierda es mayor o igual que el operador de la derecha	$12 >= 3$ devuelve True
<=	Devuelve True si el operador de la derecha es mayor o igual que el operador de la izquierda	$12 <= 3$ devuelve False
!=	Devuelve True si ambos operandos no son iguales	$12 != 3$ devuelve True

### Lógicos

OPERADOR	DESCRIPCIÓN	USO
&&	Devuelve True si ambos operandos son True	$a \&\& b$
	Devuelve True si alguno de los operandos es True	$a \ \  b$
!	Devuelve True si alguno de los operandos False	$!a$

### De asignación

OPERADOR	DESCRIPCIÓN
=	b = 5. El valor 5 es asignado a la variable b
+=	b += 5 es equivalente a b = b + 5
-=	b -= 5 es equivalente a b = b - 5
*=	b *= 3 es equivalente a b = b * 3
/=	b /= 3 es equivalente a b = b / 3
%=	b %= 3 es equivalente a b = b % 3

### Tipos de datos

TIPO	DESCRIPCIÓN	TAMAÑO	EJEMPLOS
boolean	true o false	1 bit	true, false
byte	entero	8 bits	100, -50
char	carácter	16 bits	'a', 'u', '\', 'L'
int	entero	32 bits	100, -2, -1, 0, 1, 2, -200
double	coma flotante	64 bits	1.2345e300d, -1.2345e-300f, 1e1d

## Variables

Situación	Variable
Crear una variable llamada tipo entero <b>nota</b> con un valor de 3	<code>int nota =3;</code>
Crear una variable llamada String <b>información</b> que guarde el apellido de usted	<code>String Informacion= "Fernandez";</code>
Una variable llamada doble <b>resultado</b> que almacene la suma de 10.2 más 30.5	<code>double resultado= 10.2 + 30.5;</code>
Crear una variable carácter <b>género</b> que almacene el carácter <b>f</b>	<code>char genero = 'f';</code>
Crear una variable booleano <b>respuesta</b> que almacene el carácter <b>true</b>	<code>boolean respuesta = true;</code>

## NetBeans

NetBeans IDE ofrece herramientas de primer nivel para el desarrollo de aplicaciones móviles, de escritorio, empresariales y web Java. Es el primer IDE que admite las últimas versiones de JDK, JavaEE y JavaFX. Brinda resúmenes inteligentes que le ayudan a comprender y administrar sus aplicaciones, incluida la compatibilidad inmediata con tecnologías populares, como Maven.

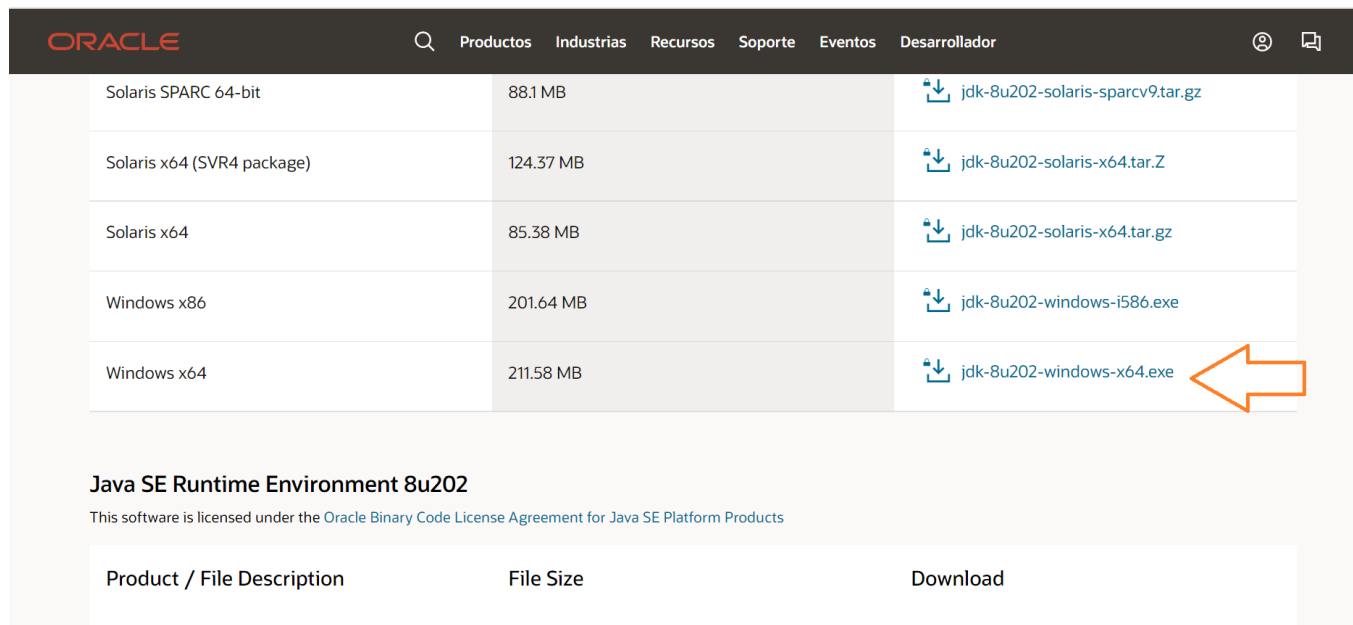
Gracias a sus características de desarrollo de aplicaciones integrales, la mejora constante de JavaEditor y las mejoras continuas de velocidad y rendimiento, NetBeans IDE marca el ritmo para el desarrollo de aplicaciones con tecnologías novedosas listas para usar.

## Instalación de NetBeans

### Paso 1

Se debe instalar como primer programa el JDK

<https://www.oracle.com/java/technologies/javase/jdk16-archive-downloads.html>



Product / File Description	File Size	Download
Solaris SPARC 64-bit	88.1 MB	<a href="#">jdk-8u202-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	124.37 MB	<a href="#">jdk-8u202-solaris-x64.tar.Z</a>
Solaris x64	85.38 MB	<a href="#">jdk-8u202-solaris-x64.tar.gz</a>
Windows x86	201.64 MB	<a href="#">jdk-8u202-windows-i586.exe</a>
Windows x64	211.58 MB	<a href="#">jdk-8u202-windows-x64.exe</a>

**Java SE Runtime Environment 8u202**  
This software is licensed under the Oracle Binary Code License Agreement for Java SE Platform Products

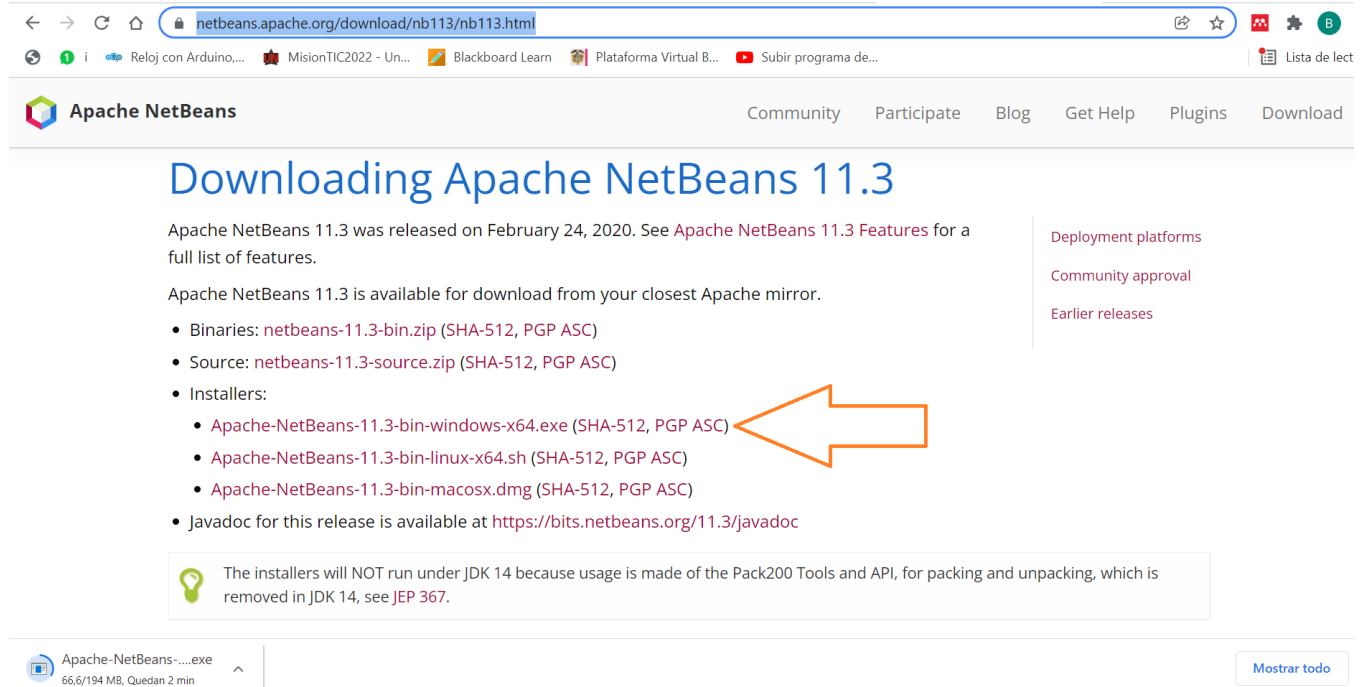
Una vez descargado, se procede con la instalación ejecutando doble clic y siguiendo las instrucciones



## Paso 2

Para la instalación del IDE NetBeans

<https://netbeans.apache.org/download/nb113/nb113.html>



Apache NetBeans 11.3 was released on February 24, 2020. See [Apache NetBeans 11.3 Features](#) for a full list of features.

Apache NetBeans 11.3 is available for download from your closest Apache mirror.

- Binaries: [netbeans-11.3-bin.zip](#) (SHA-512, PGP ASC)
- Source: [netbeans-11.3-source.zip](#) (SHA-512, PGP ASC)
- Installers:
  - [Apache-NetBeans-11.3-bin-windows-x64.exe](#) (SHA-512, PGP ASC)
  - [Apache-NetBeans-11.3-bin-linux-x64.sh](#) (SHA-512, PGP ASC)
  - [Apache-NetBeans-11.3-bin-macosx.dmg](#) (SHA-512, PGP ASC)
- Javadoc for this release is available at <https://bits.netbeans.org/11.3/javadoc>

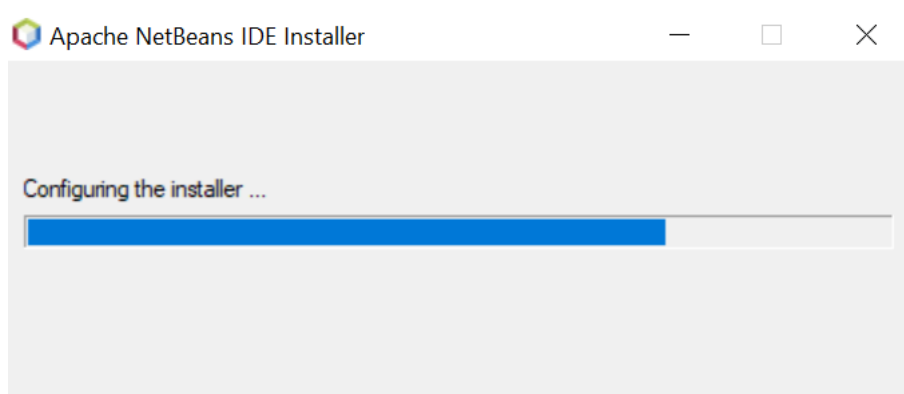
The installers will NOT run under JDK 14 because usage is made of the Pack200 Tools and API, for packing and unpacking, which is removed in JDK 14, see [JEP 367](#).

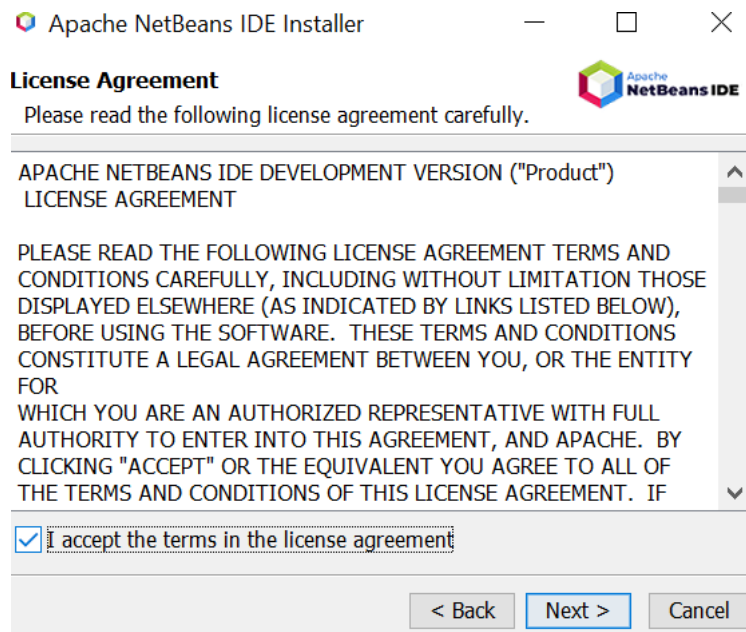
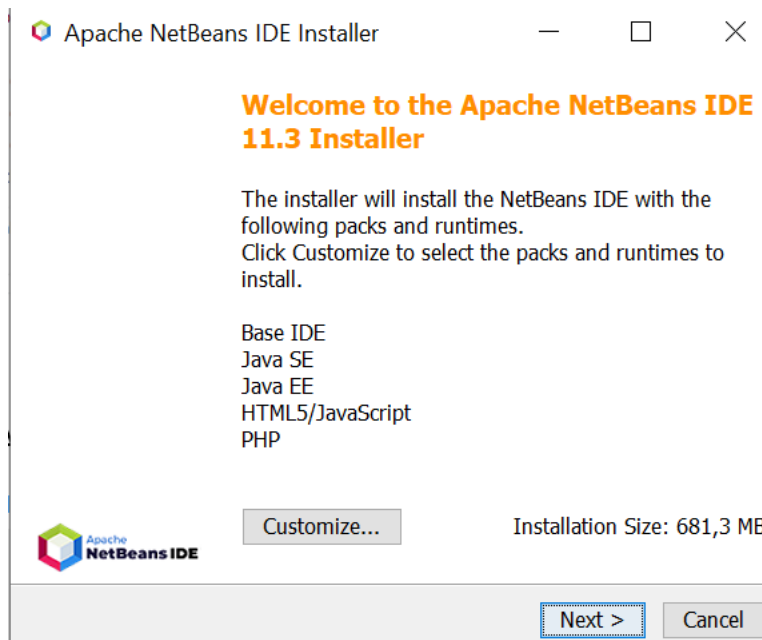
Apache-NetBeans-....exe  
66.6/194 MB, Quedan 2 min

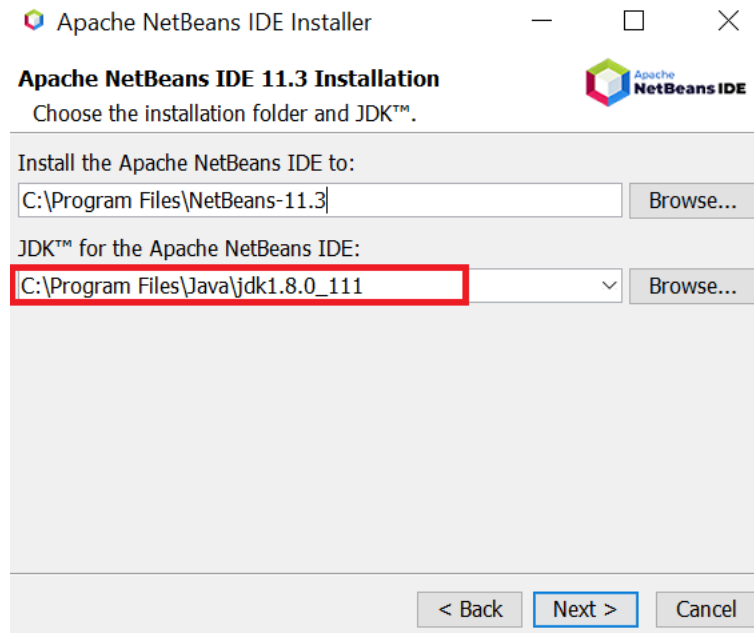
Mostrar todo

## Paso 3

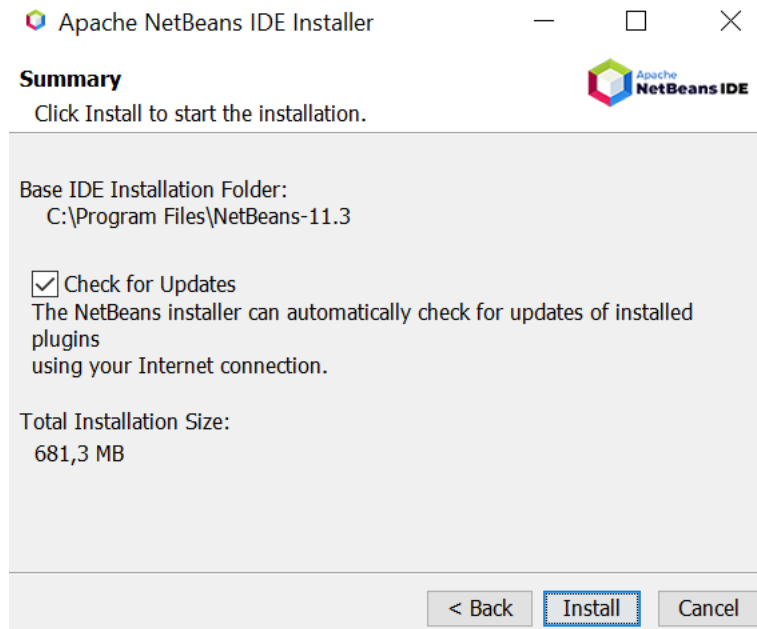
Abrir el archivo de Apeche NetBeans



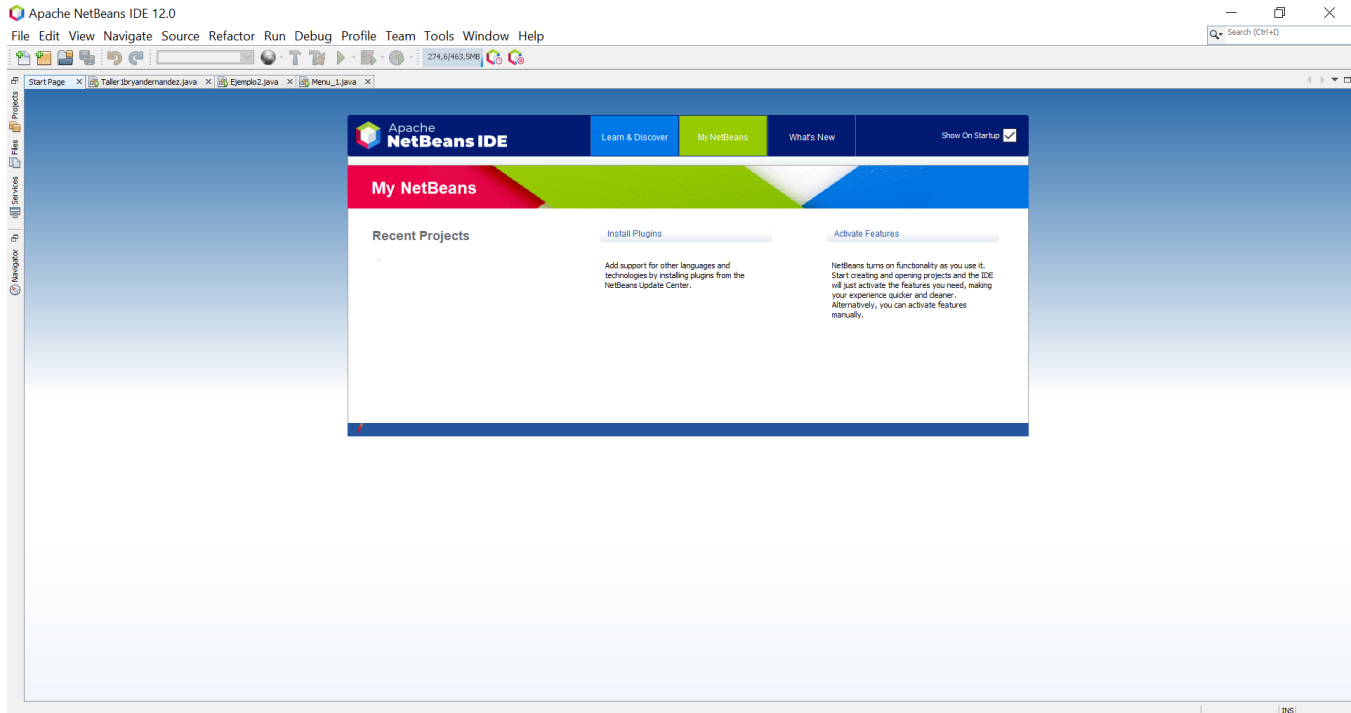




Verificar que el jdk esté instalado



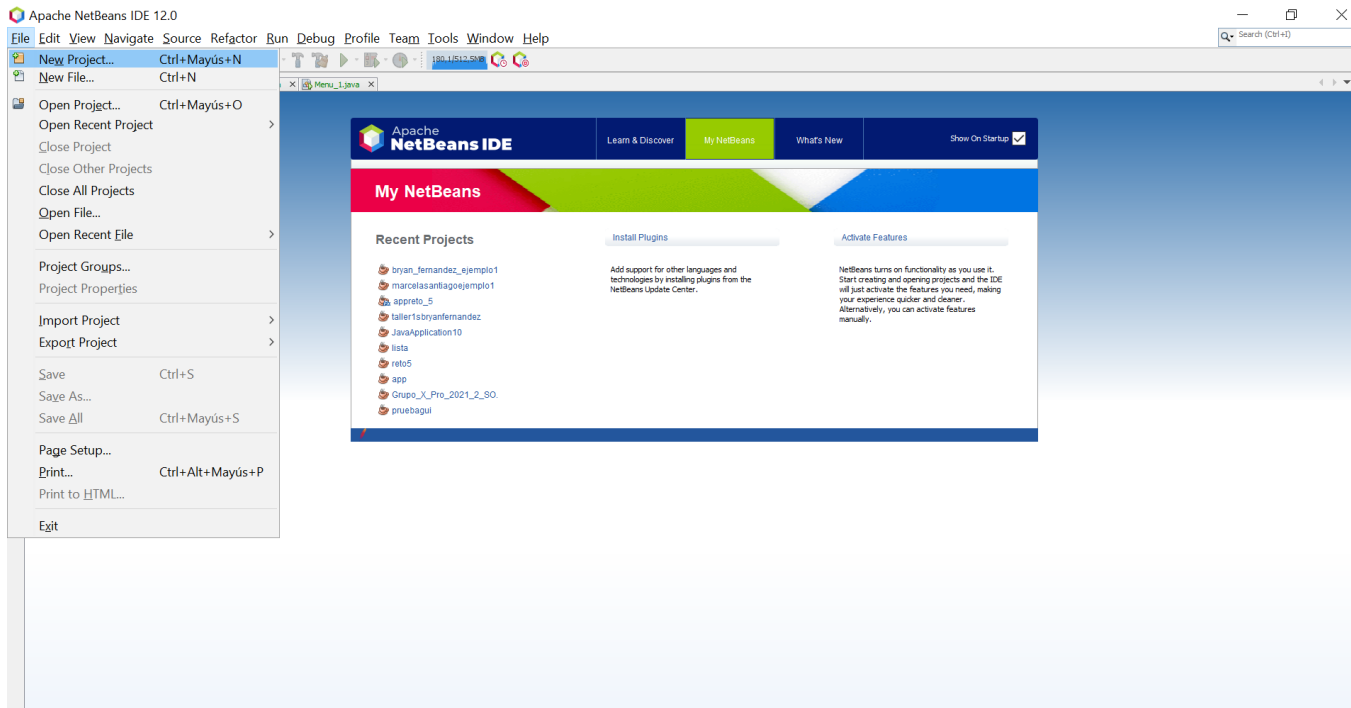
Se procede en hacer clic en instalar y se inicia el proceso de instalación del IDE.  
Una vez finalizado se procede en ingresar al IDE

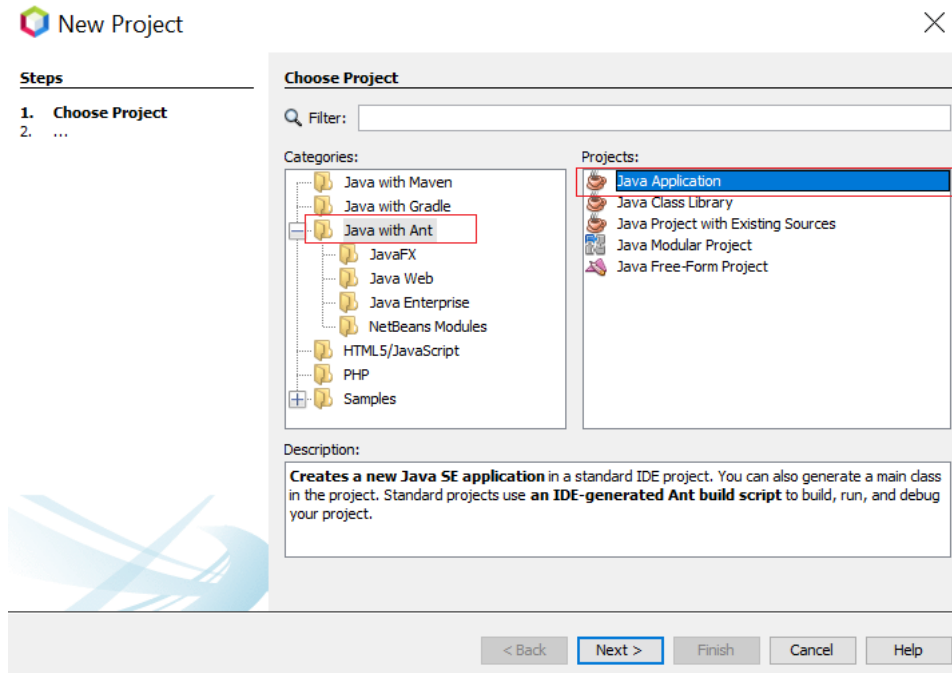


## Paso 4

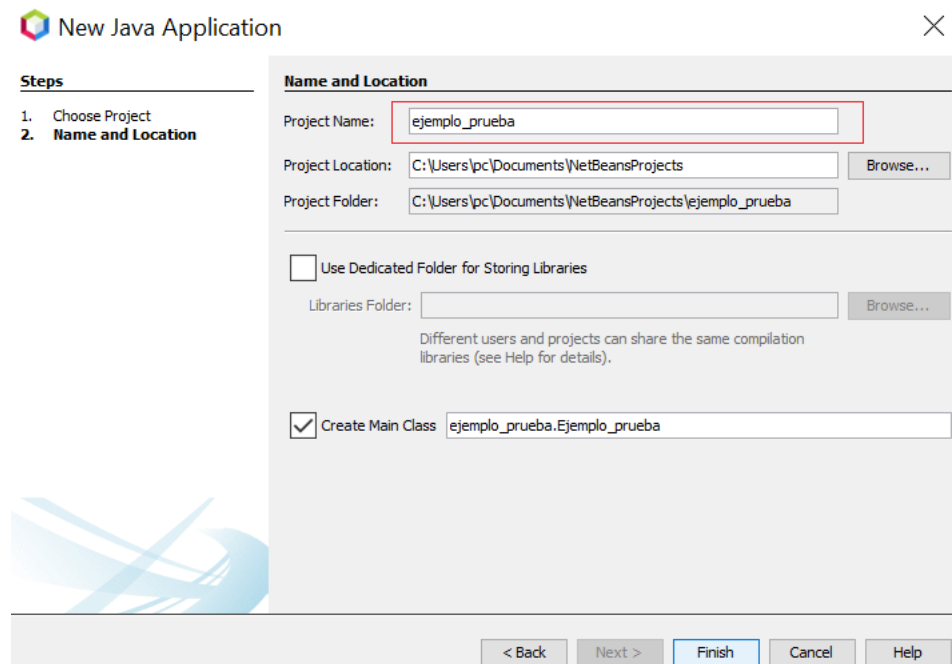
Crear un proyecto de Java en NetBeans.

Ingresar al IDE NetBeans y a continuación clic en file, clic en new Project

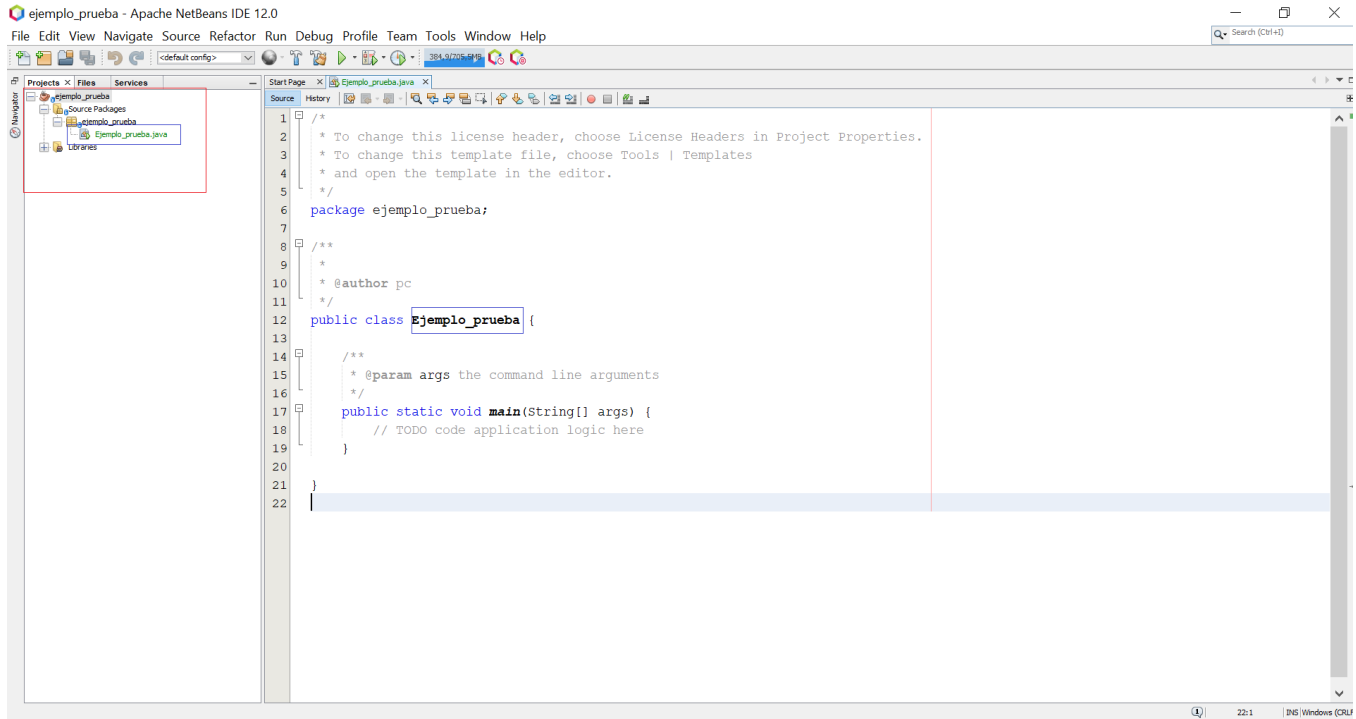




Se ingresa el nombre del proyecto



Clic en finalizar



Tenga en cuenta que el programa inicia en el método principal(main). Para ejecutar el programa en NetBeans se presiona el botón verde run o presionar la tecla F6.

## Paso 6

### Escribir en consola de Java

Para la escritura se utiliza la sintaxis `System.out.println()`. Tenga en cuenta que dentro del paréntesis puede enviar una cadena o una variable. Se puede usar la palabra `sout` seguido de la tecla `tab` para generar la sintaxis de escribir en consola.



## Algoritmo en pseudocodigo

```
Algoritmo sin_titulo
    escribir "hola mundo"
FinAlgoritmo
```

## Programa en Java

```
ejemplo_prueba - Apache NetBeans IDE 12.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
c:\default config> 206.01555-0181
Start Page x | Ejemplo_prueba.java x
Source History
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ejemplo_prueba;
7
8  /**
9  *
10 * @author pc
11 */
12 public class Ejemplo_prueba {
13
14     /**
15     * @param args the command line arguments
16     */
17     public static void main(String[] args) {
18         System.out.println("Hola mundo");
19     }
20
21 }
22
23
24 }
25
```

## Algoritmo en pseudocodigo

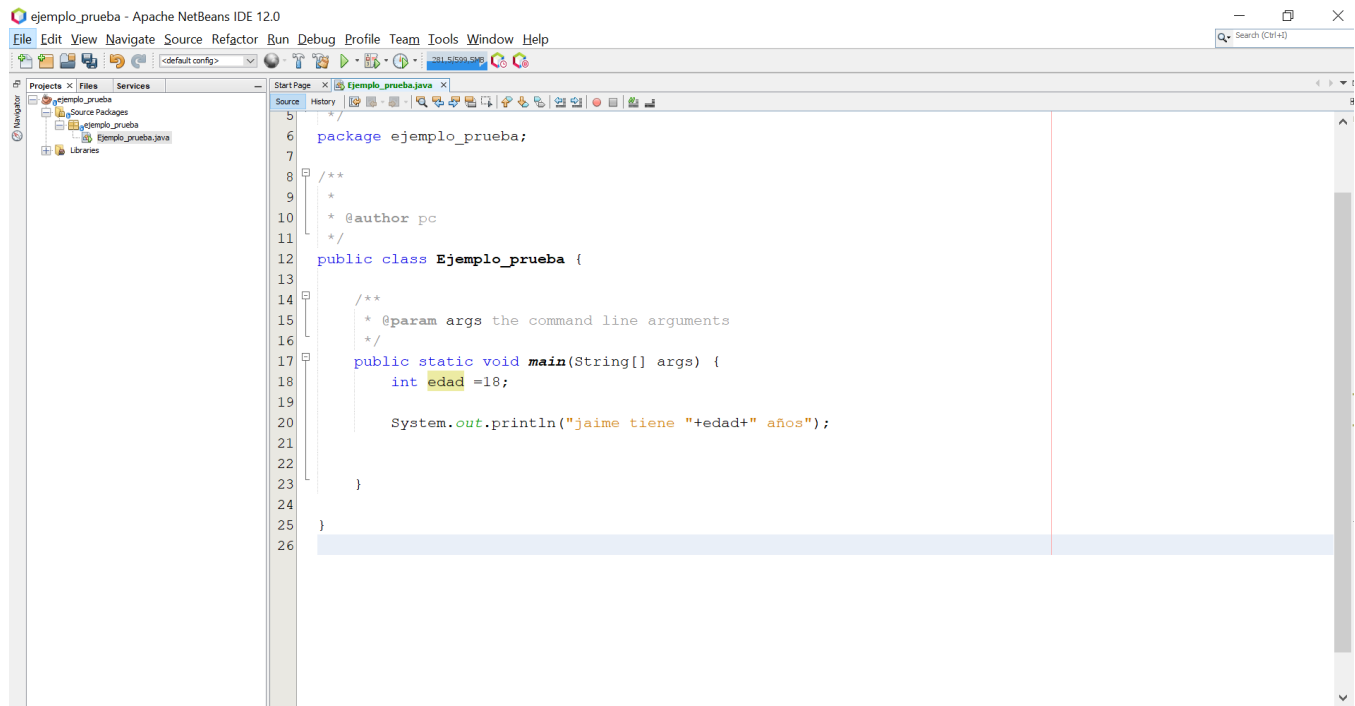
Algoritmo `sin_titulo`

edad ← 18

escribir "jaime tiene ", edad, " años"

FinAlgoritmo

## Programa en java



```
5 //
6 package ejemplo_prueba;
7
8 /**
9  *
10  * @author pc
11  */
12 public class Ejemplo_prueba {
13
14     /**
15     * @param args the command line arguments
16     */
17     public static void main(String[] args) {
18         int edad = 18;
19
20         System.out.println("jaime tiene "+edad+" años");
21
22     }
23
24 }
25
26 }
```

Tenga en cuenta que el signo más (+) que se encuentra fuera de la cadena hace referencia a concatenación. La anterior imagen tiene como codificación la creación de una variable de tipo entero llamada edad cuyo valor predeterminado es 18. A continuación se envía un mensaje a la consola formado por una cadena unido con la variable y otra cadena. Al momento de ejecutar el programa se puede observar el siguiente resultado.





Docente: Mg. Bryan Fernández Ebrath.

The screenshot shows the Apache NetBeans IDE 12.0 interface. The main editor window displays the following Java code:

```
8  /**
9  *
10 * @author pc
11 */
12 public class Ejemplo_prueba {
13
14     /**
15     * @param args the command line arguments
16     */
17     public static void main(String[] args) {
18         int edad =18;
```

The Output window at the bottom shows the execution results:

```
run:
jaime tiene 18 años
BUILD SUCCESSFUL (total time: 0 seconds)
```

The text "jaime tiene 18 años" is highlighted with a red box in the original image.

## Paso 7

### Lectura de datos en java

Existe varias maneras per obtener datos de entrada

- ✓ Clase Scanner.
- ✓ Archivos de texto.
- ✓ Lectura con buffer reader.
- ✓ JOptionpane.

### Clase Scanner

Scanner es una clase en el paquete java.util utilizada para obtener la entrada de los tipos primitivos como int, double etc. y también String. Es la forma más fácil de leer datos en un programa Java, aunque no es muy eficiente si se quiere un método de entrada para escenarios donde el tiempo es una restricción, como en la programación competitiva.

## Paso 8

### Lectura de datos en Java con la clase Scanner

Importar la librería java.util.Scanner

Las librerías se importan entre el paquete y la clase

```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ejemplo_prueba;
7  import java.util.Scanner;
8  /**
9  *
10 * @author pc
11 */
12 public class Ejemplo_prueba {
13
14     /**
15     * @param args the command line arguments
16     */
17     public static void main(String[] args) {
```

Se crea el objeto de la clase Scanner. Este objeto se coloca el nombre cualquiera pero comúnmente lo llaman lectura, leer, read o teclado. Para esta práctica se llamará **lectura**. El objeto se crea dentro del método principal

```
package ejemplo_prueba;
import java.util.Scanner;
/**
 *
 * @author pc
 */
public class Ejemplo_prueba {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        Scanner lectura = new Scanner(System.in);

    }

}
```

Para aplicar lectura de datos se debe tener en cuenta el tipo de datos a leer.

Si el dato a recibir es de tipo int, el método a utilizar es nextInt().

Si el dato a recibir es de tipo double, el método a utilizar es nextDouble().

Si el dato a recibir es de tipo String sin espacios, el método a utilizar es next().

Si el dato a recibir es de tipo String con espacios, el método a utilizar es nextLine().

Si el dato a recibir es de tipo boolean sin, el método a utilizar es nextBoolean().



## Paso 9

### Lectura y escritura en Java

#### Algoritmo en pseudocódigo

Algoritmo `sin_titulo`

    Escribir "Digite numero 1"

    leer n1

    Escribir "Digite numero 2"

    leer n2

$s \leftarrow n1 + n2$

    Escribir "la suma es ",s

FinAlgoritmo

#### Codificación en java

- ✓ Antes de realizar la codificación en el lenguaje Java tenga en cuenta los siguientes detalles:
- ✓ Se debe importar la librería `java.util.Scanner`.
- ✓ Se debe crear el objeto de la clase `Scanner`. Para este ejemplo se llamará **lectura**.
- ✓ El algoritmo a codificar tiene tres variables que se deben crear. Para este ejemplo serán variables `double` (reales).
- ✓ Como los datos de entrada o que se recibe son reales, el método debe ser `nextDouble`.
- ✓ Si desea programar sobre un documento más claro, puede borrar las palabras de color gris que son llamadas **comentarios**. Tenga en cuenta borrar el comentario desde el inicio hasta el final al mismo tiempo.



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

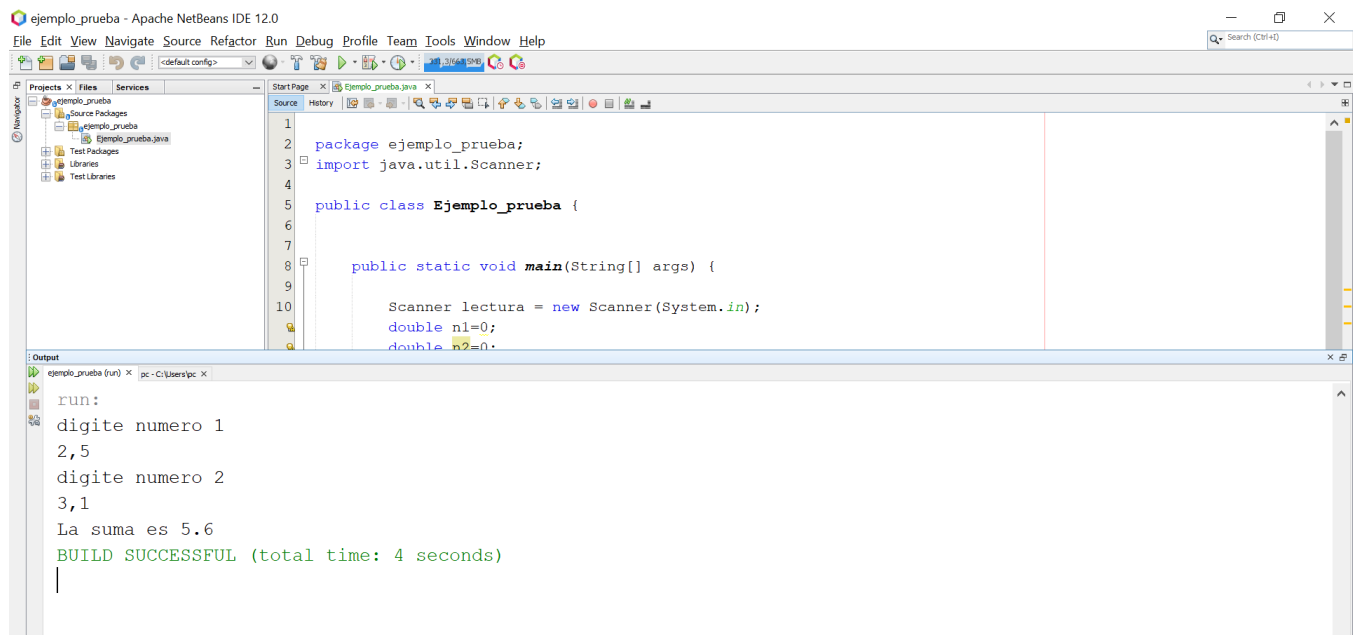
public class Ejemplo_prueba {

    public static void main(String[] args) {

        Scanner lectura = new Scanner(System.in);
        double n1=0;
        double n2=0;
        double s=0;

        System.out.println("digite numero 1 ");
        n1=lectura.nextDouble();
        System.out.println("digite numero 2 ");
        n2=lectura.nextDouble();
        s=n1+n2;
        System.out.println("La suma es "+s);
    }
}
```

Al momento de hacer una prueba se debe hacer clic en Run o presionar F6



## Paso 10

Crear un proyecto en Java con NetBeans para cada algoritmo y codificarlo

```
1 Proceso salario
2   Escribir "programa para determinar el salario neto";
3   Escribir "digite el nombre del trabajador:";
4   Leer Nombre;
5   Escribir "numero de horas:";
6   Leer horas;
7   Escribir "valor de las horas:";
8   Leer valor;
9   salario<-valor*horas;
10  Escribir "el trabajador:",Nombre,"tienen un salario neto de:",salario-(salario*7/100);
11 FinProceso
```

## Algoritmo 2

```
1 Proceso Sueld
2   Escribir 'Calcular Sueldo y horas extra: ';
3   Escribir 'Ingrese el nombre del trabajador: ';
4   Leer trabajador;
5   Escribir 'Ingrese el numero de horas de trabajo al dia: ';
6   Leer horas_dia;
7   Escribir 'ingrese el valor de cada hora de trabajo: ';
8   Leer valor_hora;
9   Escribir 'Ingrese el numero de dias trabajados en el mes: ';
10  Leer Dias_mes;
11  Escribir 'Ingrese el numero de horas extra: ';
12  Leer horas_extra;
13  Escribir 'Ingrese el valor de cada hora extra: ';
14  Leer valor_extra;
15  Sueldo_dia<-horas_dia*valor_hora;
16  Sueldo_mes<-Sueldo_dia*Dias_mes;
17  Sueldo_extra<-horas_extra*valor_extra;
18  Sueldo_total<-Sueldo_mes+Sueldo_extra;
19  Escribir 'El sueldo total es: ',Sueldo_total;
20 FinProceso
```



### Algoritmo 3

```
1  Proceso volumen
2      Escribir "escribir la longitud:";
3      Leer A;
4      Escribir "escribir el ancho:";
5      Leer B;
6      Escribir "escribir el espesor:";
7      Leer C
8      volumen<-A+B+C;
9      Escribir "el volumen del trozo de metal es:",volumen;
10 FinProceso
11
```



Docente: Mg. Bryan Fernández Ebrath.

## Referencias Bibliográficas

Martinez,J. (2011)Fundamentos de programación en Java. Madrid España: EME.

Deitel, P., & Deitel, H. (2012). Cómo programar Java (9a. ed. --.). México: Pearson.