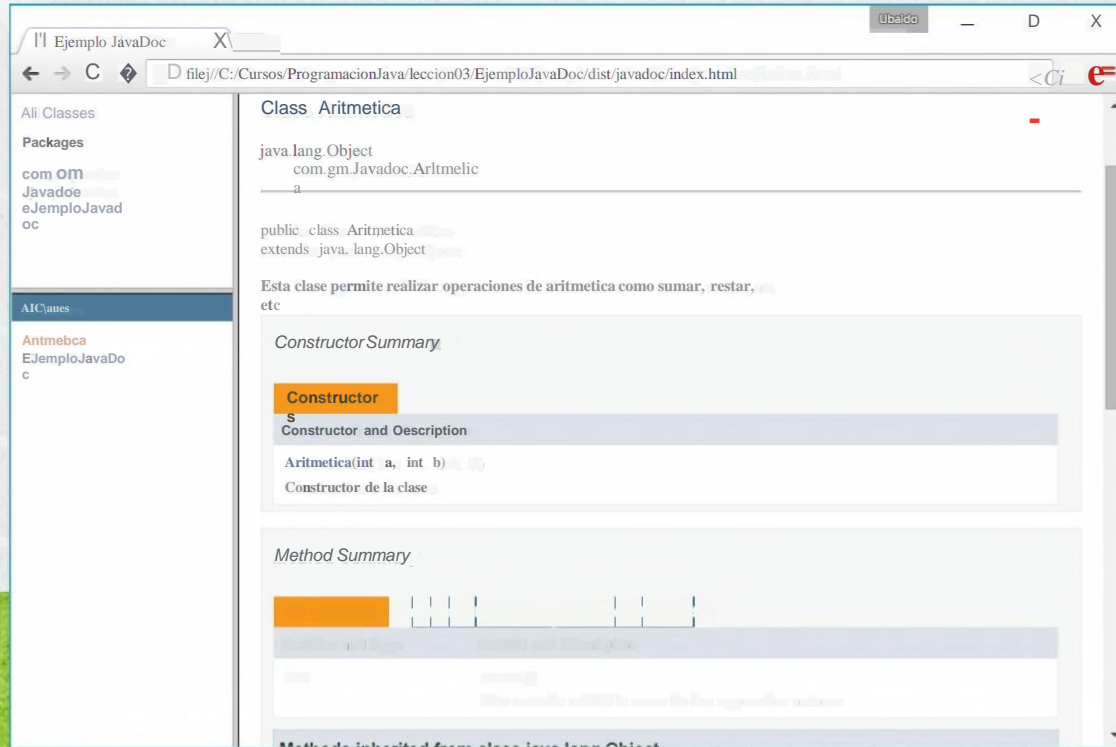


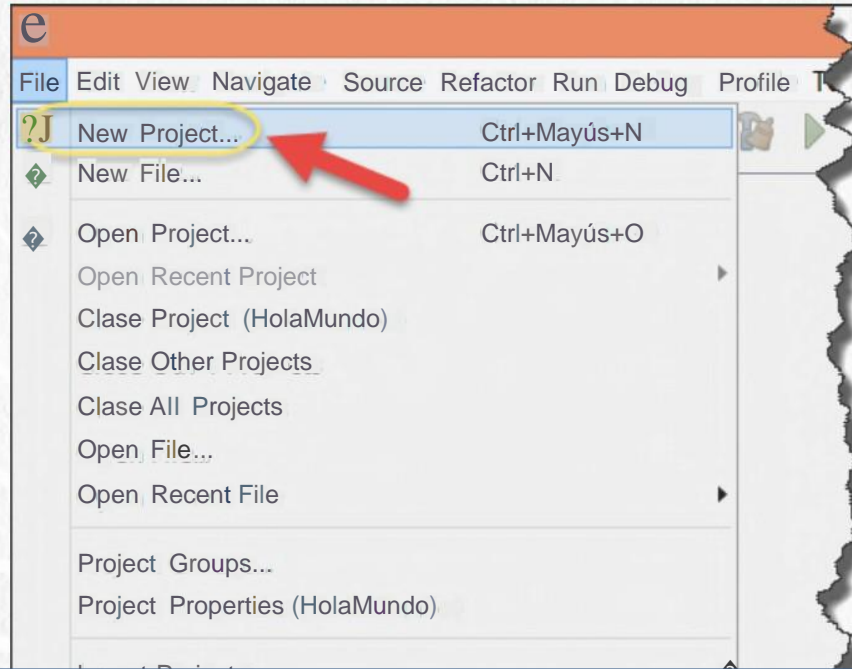
OBJETIVO DEL EJERCICIO

Crear un programa para practicar en manejo de Javadoc en Java. Al finalizar deberemos observar lo siguiente:



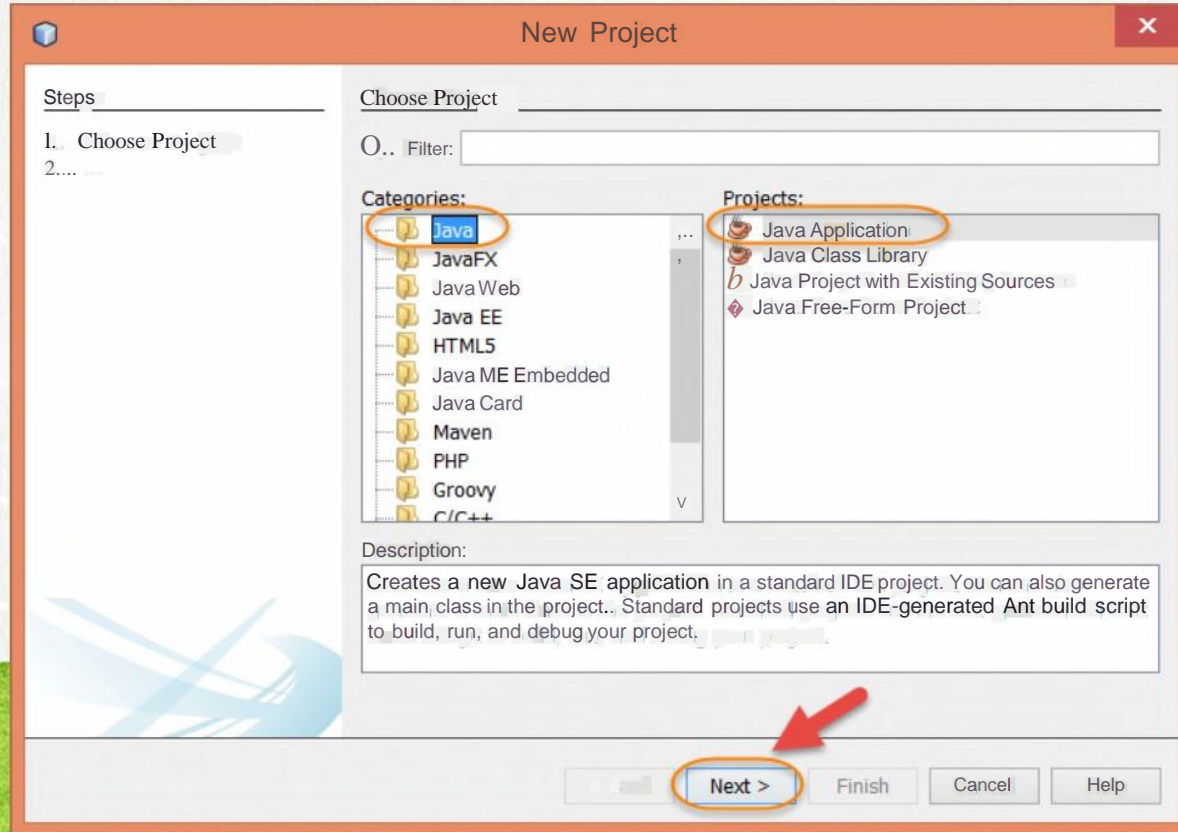
PASO I. CREACIÓN DEL PROYECTO

Vamos a crear el proyecto EjemploJavaDoc:



PASO I. CREACIÓN DEL PROYECTO (CONT)

Vamos a crear el proyecto EjemploJavaDoc:



PASO I. CREACIÓN DEL PROYECTO (CONT)

Vamos a crear el proyecto EjemploJavaDoc:

New Java Application

Steps

1. Choose Project
2. Name and Location

Name and Location

Project Name:

Project Location: Browse...

Project Folder:

Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

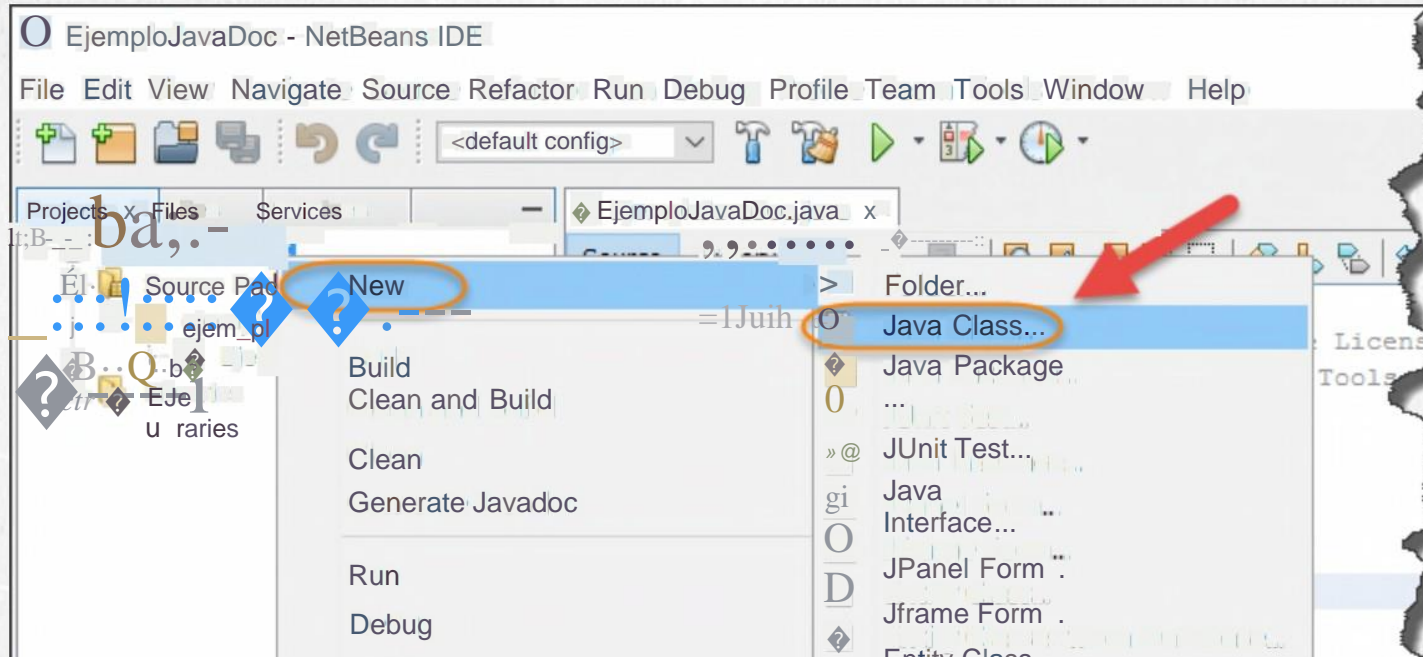
Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class

Buttons: Back, Next >, **Finish**, Cancel, Help

PASO 2. CREACIÓN DE UNA CLASE

Vamos a crear una nueva clase:



PASO 2. CREACIÓN DE UNA CLASE (CONT)

Vamos a crear una nueva clase:

New Java Class

steps

1. Choose File Type
2. Name and Location

Name and Location

Class Name: Artemetica

Project: EjemploJavaDoc

Location: Source Packages

Package: m.gm.javadoc

Created File: rogramacionJava\LeccionOJ\EjemploJavaDoc\src\com\gm\javadoc\Artemetica.java

Next >

PASO 3. MODIFICAMOS EL CÓDIGO

Archivo Aritmetica.java:

```
package com.gm.javadoc;

/**
 * Esta clase permite realizar operaciones de aritmetica como sumar, restar, etc
 *
 * @author: Ubaldo Acosta
 * @version: 1.0
 */
public class Aritmetica {

    /**
     * Primer Operando
     */
    int operandoA;
    /**
     * Segundo operando
     */
    int operandoB;

    /**
     * Constructor de la clase
     * @param a Recibe el valor para el operando A
     * @param b Reciboe el valor para el operando B
     */
    public Aritmetica(int a, int b) {
        this.operandoA = a;
        this.operandoB = b;
    }

    /**
     * Este metodo realiza la suma de dos operandos enteros
     * @return int resultado de la suma
     */
    public int sumar() {
        return operandoA + operandoB;
    }
}
```

PASO 4. MODIFICAMOS EL CÓDIGO

Archivo EjemploJavaDoc:

```
/*
 * Derechos reservados GlobalMentoring.com.mx
 */
package ejemplojavadoc;

import com.gm.javadoc.Aritmetica;

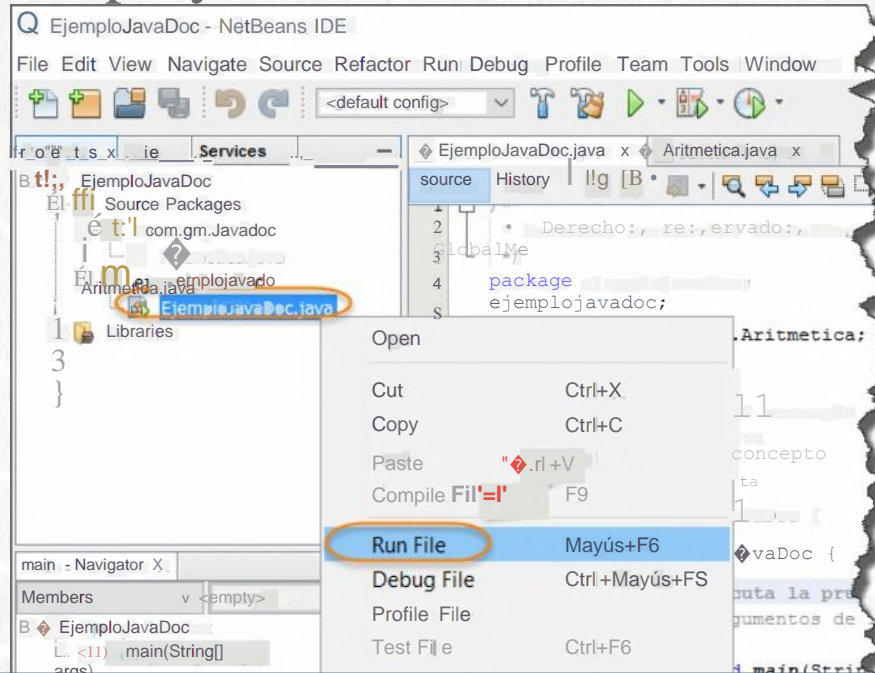
/**
 * Clase para probar el concepto de JavaDoc
 * @author Ubaldo Acosta
 * @version 1.0
 */
public class EjemploJavaDoc {

    /**
     * Metodo que ejecuta la prueba de la clase Aritmetica
     * @param args argumentos de la linea de comandos
     */
    public static void main(String[] args) {
        int resultado = new Aritmetica(3, 2).sumar();

        System.out.println("resultado:" + resultado);
    }
}
```

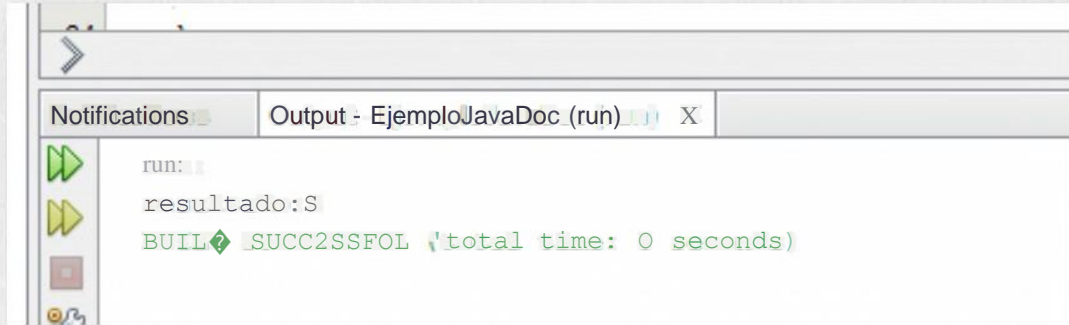

PASO 5. EJECUTAMOS EL PROYECTO

Ejecutamos nuestro proyecto. Damos click derecho -> Run:



PASO 5. EJECUTAMOS EL PROYECTO (CONT)

El resultado es como sigue:

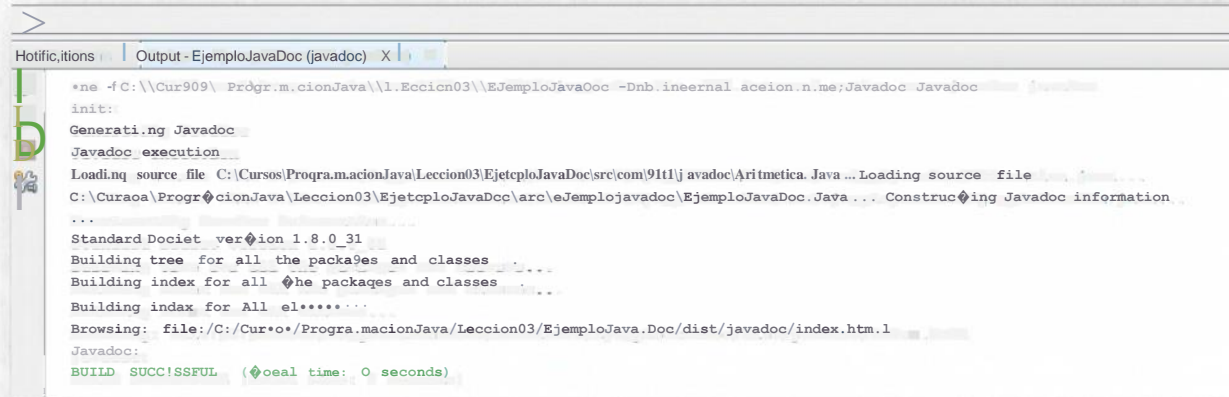
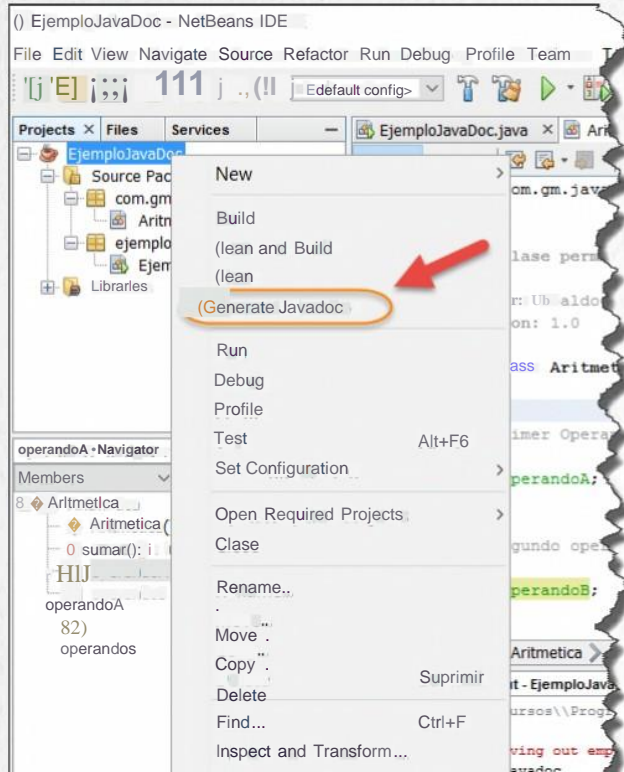


The screenshot shows an IDE's Output window with the title 'Output - EjemploJavaDoc (run)'. The window contains the following text:

```
run:
resultado:S
BUILD SUCCESSFUL (total time: 0 seconds)
```

On the left side of the output window, there are three green arrows pointing right, indicating successful execution, and a red square icon at the bottom.

PASO 6. CREAMOS EL JAVADOC



PASO 7. OBSERVAMOS EL JAVADOC

Cl Ejemplo x

JavaDoc

file:///C:/Cursos/ProgramacionJava/Leccion03/EjemploJavaDoc/dist/javadoc/index.html

Ubaldo

All Classes

Packages

com.gm.javadoc

ejemplojavadoc

All Classes

Aritmetica

EjemploJavaDoc

Class Aritmetica

java.lang.Object
com.gm.javadoc
Aritmetica

```
public class Aritmetica extends java.lang.Object
```

Esta clase permite realizar operaciones de aritmetica como sumar, restar, etc

Constructor Summary

Constructors

Constructor and Description
Aritmetica(int a, int b) Constructor de la clase

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
int	sumar() Este metodo realiza la suma de dos operandos enteros	

www.globalmentoring.com.mx

CONCLUSION DEL EJERCICIO

- Con este ejercicio hemos puesto en práctica en concepto de JavaDoc.
- Hemos visto que se utiliza para documentar nuestras clases, y a diferencia de un comentario común como son `//` o `/* */`, el comentario de JavaDoc es `/** */` y este comentario se agrega a la documentación en HTML generada para el proyecto en cuestión.