**Java EE**

Java EE es una plataforma construida sobre Java SE que proporciona una API y un entorno de ejecución para desarrollar y ejecutar aplicaciones de red a gran escala, de múltiples niveles, escalables, confiables y seguras. Estas aplicaciones se denominan aplicaciones empresariales porque resuelven los problemas que enfrentan las grandes empresas. Sin embargo, las grandes corporaciones y agencias gubernamentales no son las únicas que pueden beneficiarse de este tipo de aplicaciones y de los beneficios que proporciona Java EE. Las soluciones que ofrece la plataforma Java EE son útiles y, a veces, simplemente necesarias, para desarrolladores individuales y pequeñas organizaciones.

Java EE se desarrolla a través del Java Community Process (JCP), formado en 1998. Permite a las partes interesadas participar en la configuración de futuras versiones de las especificaciones de la plataforma del lenguaje Java. La base de este proceso son los JSR (Solicitud de especificación de Java), documentos formales que describen las especificaciones y tecnologías que se proponen agregar a la plataforma Java. Estas solicitudes las realizan miembros de la comunidad: desarrolladores y empresas comunes. Estos últimos incluyen Oracle, Red Hat, IBM, Apache y muchos otros. Aquellos. Los chicos proponen nuevas características y ventajas para su consideración que les gustaría incluir en Java.

**Java Server Faces (JSF)**

JavaServer Faces (JSF) es una infraestructura de interfaz de usuario o una interfaz de programación de aplicaciones (API) que facilita el desarrollo de aplicaciones web Java™ .

WebSphere® Application Server da soporte a JavaServer Faces 2.0 en un nivel de tiempo de ejecución. El tiempo de ejecución de JSF también:

Facilita la construcción de una interfaz de usuario a partir de un conjunto de componentes de interfaz de usuario reutilizables

Simplifica la migración de datos de aplicación hacia la interfaz de usuario o desde ésta. Ayuda a gestionar el estado de la interfaz de usuario durante las peticiones de servidor. Proporciona un modelo sencillo para enviar sucesos generados por el cliente al código de aplicación del lado del servidor.

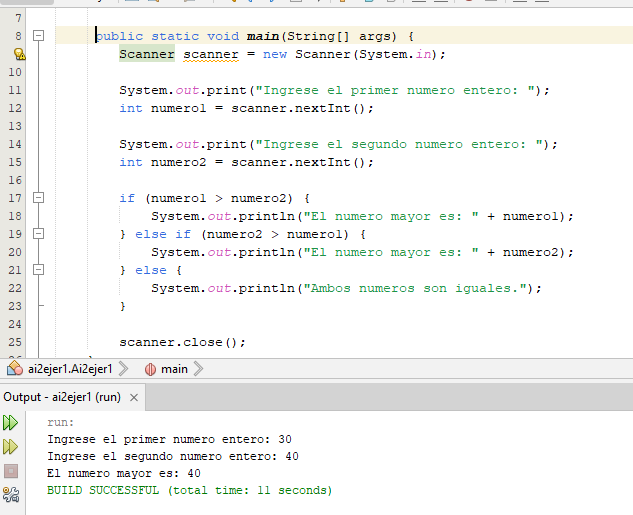
Permite crear y reutilizar con facilidad los componentes de una interfaz de usuario personalizada, Con el producto se envían la implementación de Sun Reference y la implementación de Apache MyFaces.

La implementación JSF de Apache MyFaces proporciona la base del código utilizado para el soporte JSF en WebSphere Application Server. La versión del tiempo de ejecución de JSF proporcionado por el servidor de aplicaciones reside en la ubicación de la biblioteca de tiempo de ejecución normal y está disponible para todas las aplicaciones Web que utilizan la API de JSF. La carga del servlet JSF funciona de la misma forma que si el tiempo de ejecución se empaquetara con la aplicación Web. La versión empaquetada incluye mejoras para una mejor integración con la exploración de anotaciones incorporada y otros componentes de tiempo de ejecución de WebSphere Application Server

**Link de repositirio GIF =** [**https://github.com/WuiliGusU/Asignaciones-individuales---TAREA-2.git**](https://github.com/WuiliGusU/Asignaciones-individuales---TAREA-2.git)

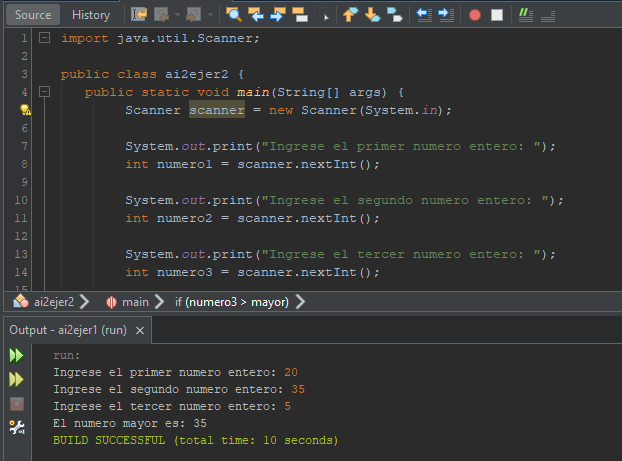
**Hoja de trabajo 2**

1. Escribir un programa que permita ingresar 2 números enteros, los compare y devuelva el mayor.



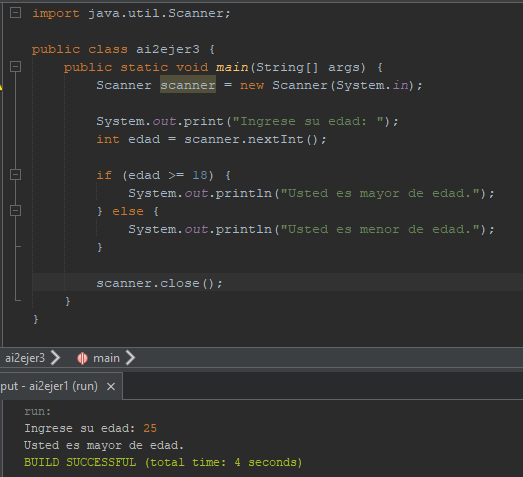
Solución: El programa se podrá realizar solicitando al usuario que ingrese los 2 números para almacenarlos, serán comparados mediante una sintaxis if para que puedan ser evaluados y mostrar el número mayor.

1. Modifique el programa anterior para que pueda evaluar 3 números enteros.



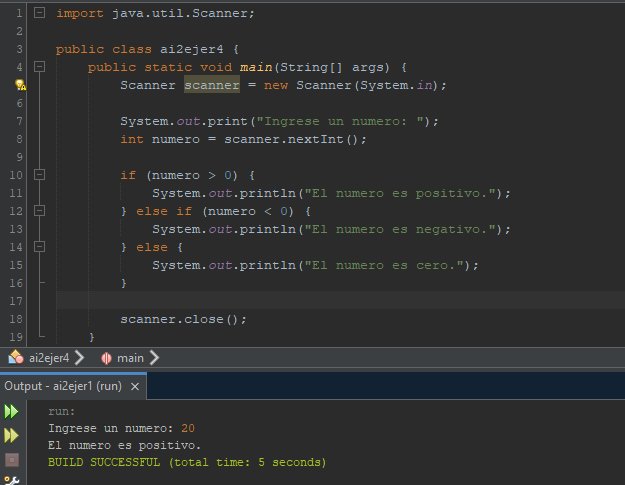
Solución: Se compara con los otros dos números el primer numero el cual se asume como mayor, se actualiza el valor mayor su uno de ellos 3 es mayor, mediante secuencias if.

1. Escribir un programa que permita ingresar la edad del usuario, y muestre en pantalla sí es mayor o menor de edad.



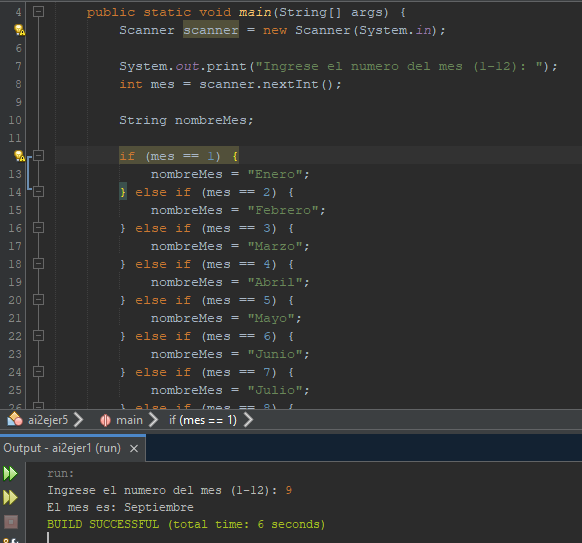
Solución: El programa solicitara ingresar una edad la cual es comparada mediante la secuencia de “if-else” por lo cual compara la edad ingresada si es menor de 18 años mostrara “menor de edad” si fuere mayor a 18 años mostrara “mayor de edad”.

1. Escribir un programa que permite ingresar un número entero y devuelva sí es positivo o negativo.



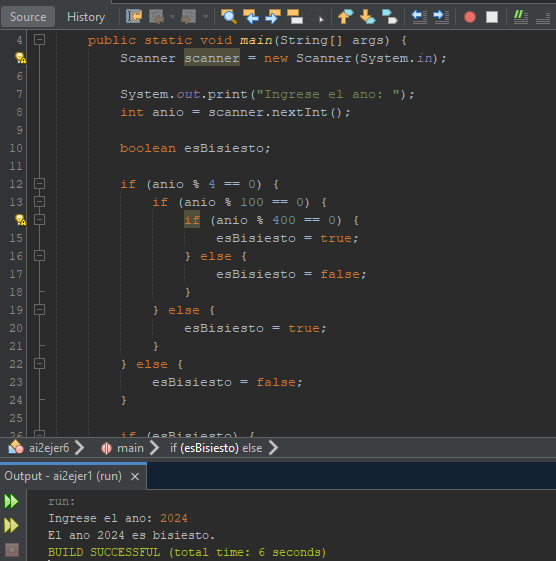
Solución: El programa se realizara ingresando un numero entero el cual se evaluara según las condiciones requeridas mediante if y else if, evaluando de manera que reconocerá si el numero es negativo (menor a 0) o positivo (mayor a 0) e igualmente reconocerá si el numero ingresado es 0.

1. Escribir un programa que permita ingresar el número del mes y devuelva el nombre del mismo (no utilizar switch).



Solución: Se solicitara al usuario que ingrese un numero entre 1 y 12 utilizando nextInt(), se realizara comparaciones mediante “if-else” para determinar el nombre del mes según el numero ingresado, si el numero esta fuera del rango solicitado mostrara que el número no es válido.

1. Escriba un programa que permita ingresar el dato año y devuelva, sí es bisiesto o no.



Solución: Se solicita al usuario que ingrese un año utilizando nextInt(),

Un año es bisiesto si es divisible por 4, sin embargo, si también es divisible por 100, entonces no es bisiesto, a menos que también sea divisible por 400.

Mediante “if-else” se comienza verificando si el año es divisible por 4.

Luego, si es divisible por 100, se verifica si también es divisible por 400.

Si cumple con las condiciones, se marca como bisiesto; de lo contrario, no lo es.