

**I**nstituto **T**ecnológico de **O**axaca

**“*Tecnología Propia e Independencia Económica***”

**T**area: Ejercicios 5,6 y 7 Java

**C**arrera: **I**ngeniería en **S**istemas **C**omputacionales.

**E**studiante: **J**uárez **L**ucas **E**dson **R**ubén

**N**o. de **C**ontrol: 18161150

**G**rupo y **H**ora: 9-10 am

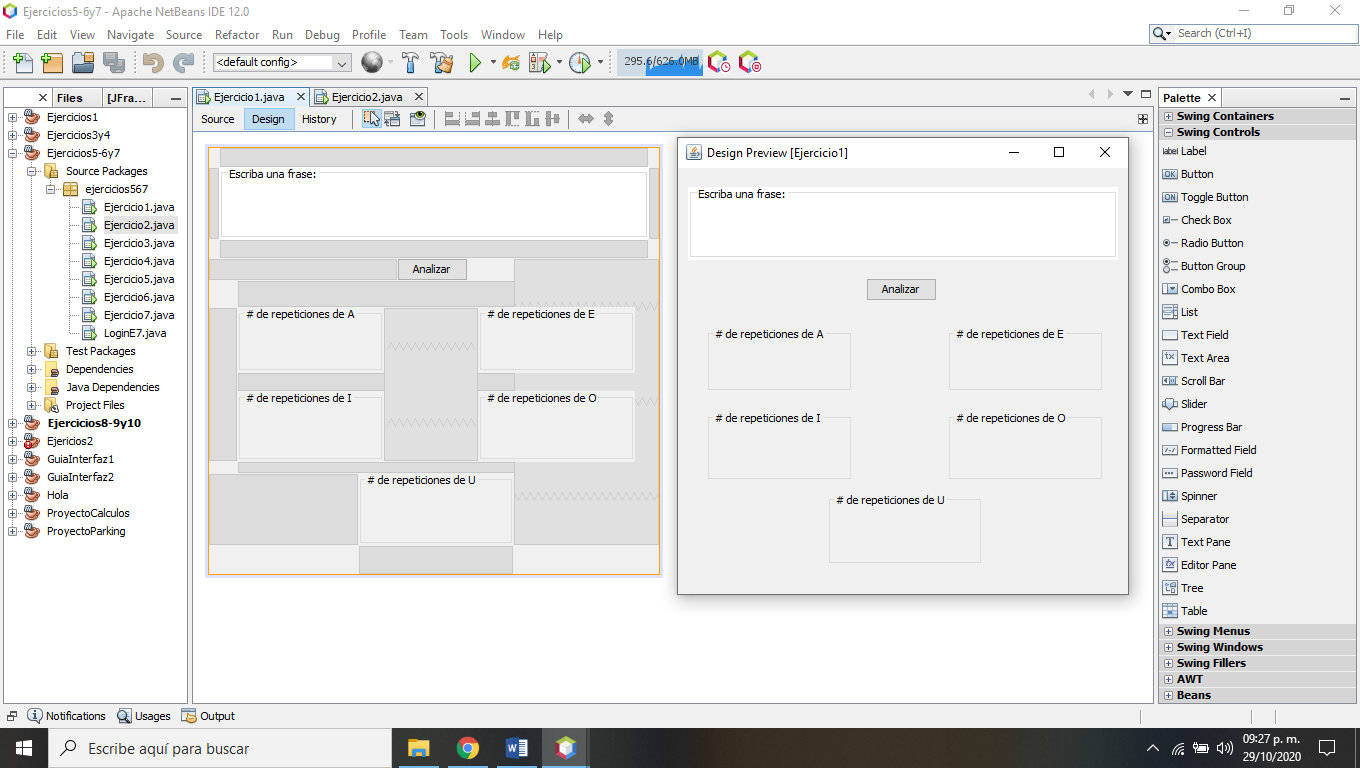
**P**rofesor: Luis Alberto Alonso Hernández

**O**axaca de **J**uárez, **O**axaca.

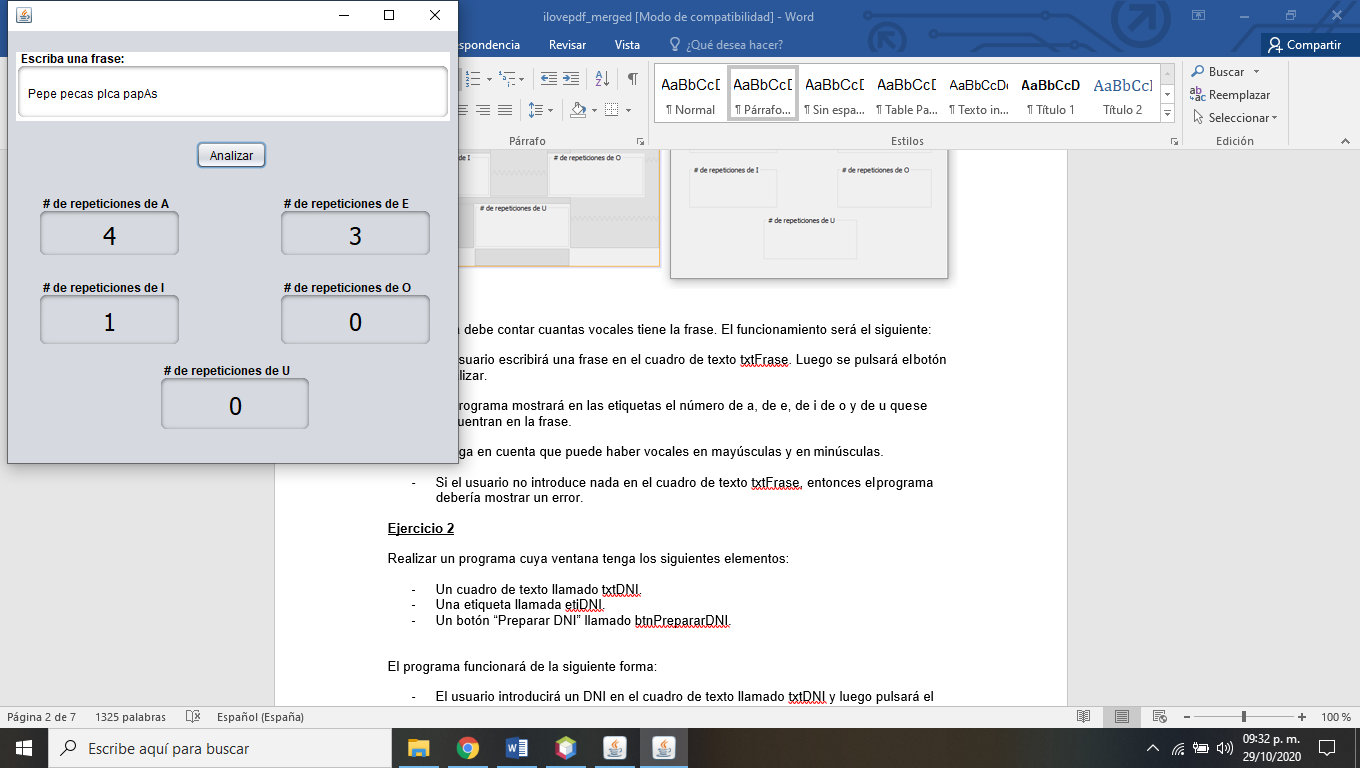
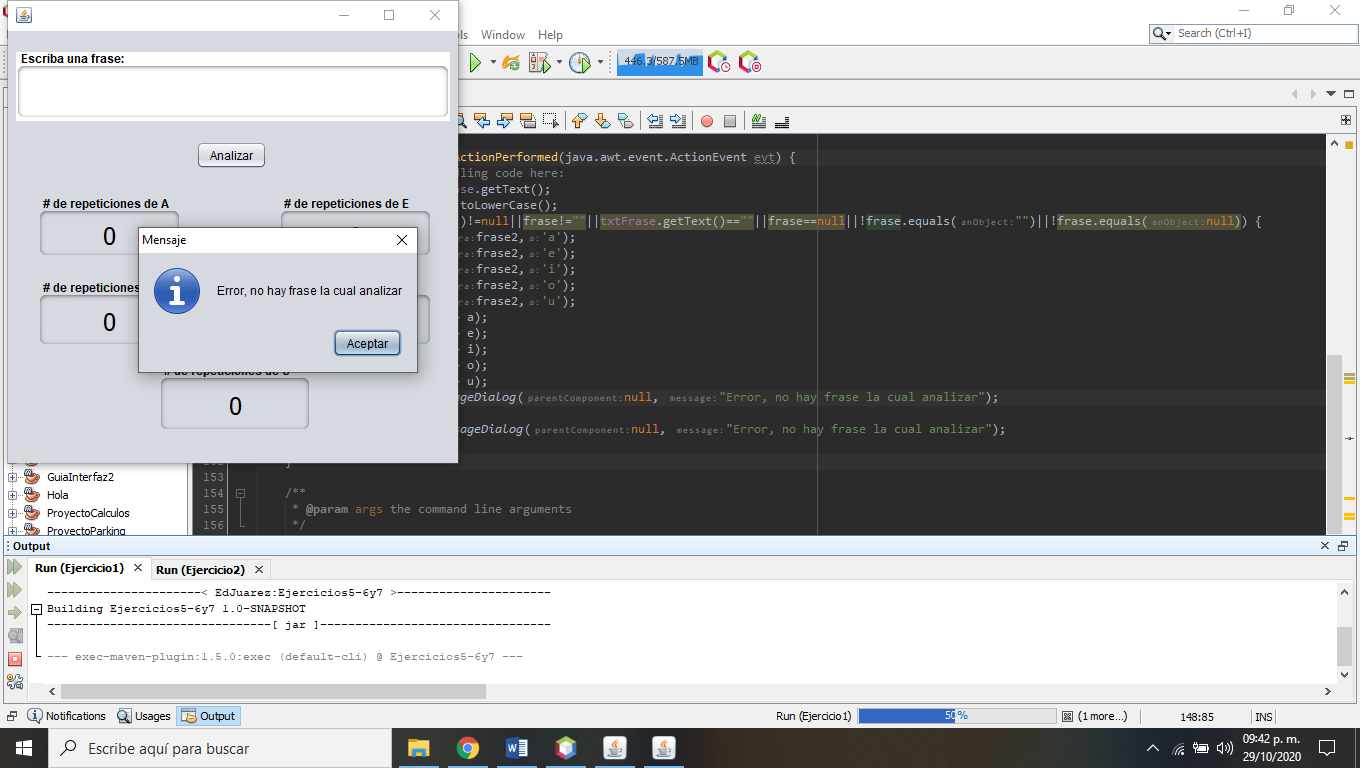
# CADENAS (CONTINUACIÓN) PDF #5

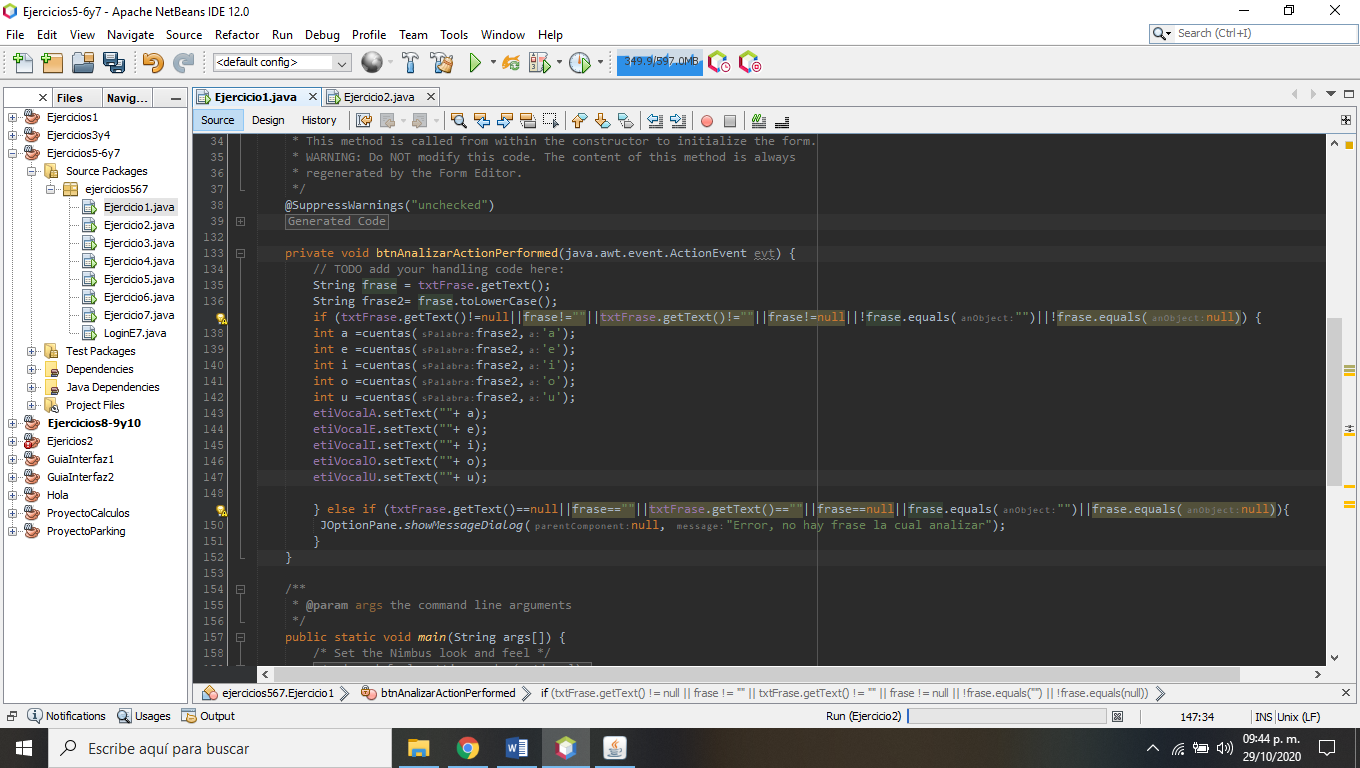
**Ejercicio 1**

Realizar un programa cuya ventana tenga los siguientes elementos:

* Un cuadro de texto llamado txtFrase.
* Varias etiquetas.
* Un botón “Analizar” llamado btnAnalizar.

El programa debe contar cuantas vocales tiene la frase. El funcionamiento será el siguiente:

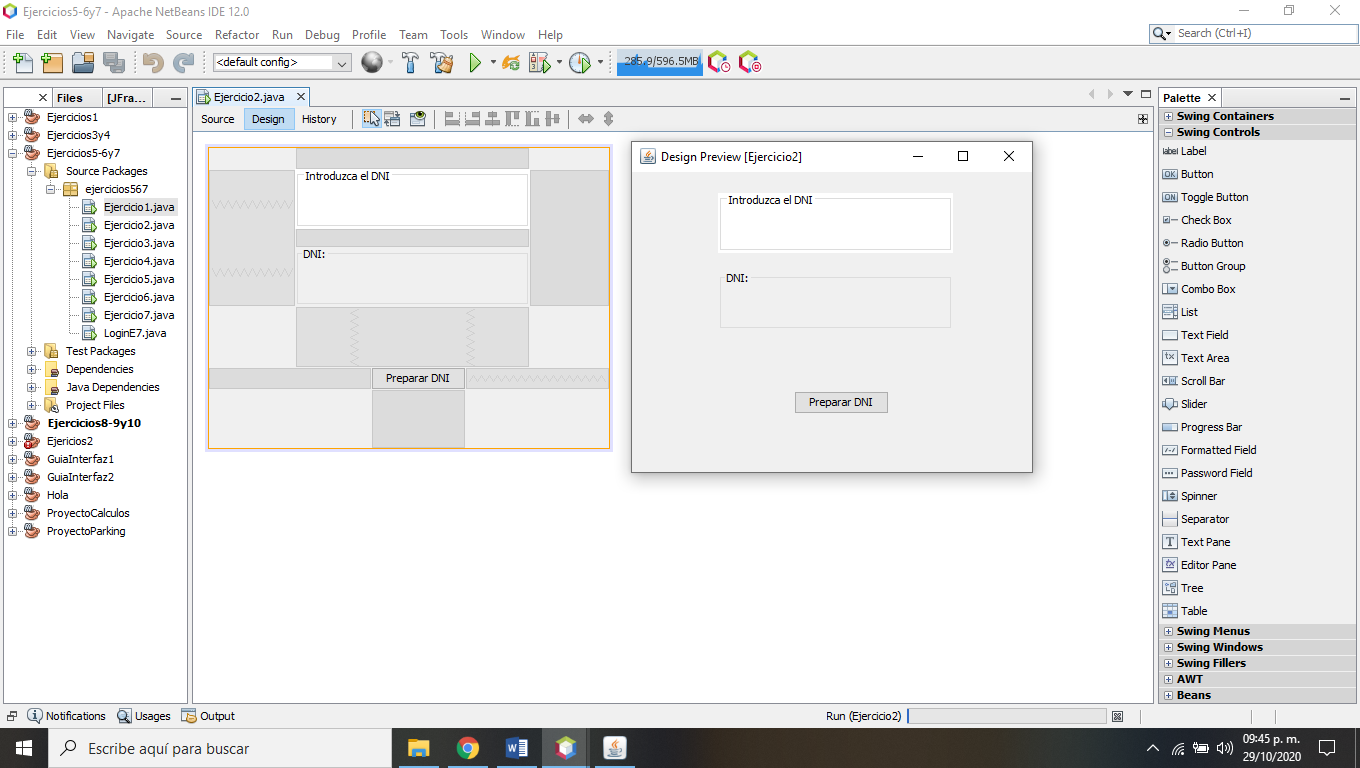
* El usuario escribirá una frase en el cuadro de texto txtFrase. Luego se pulsará el botón Analizar.
* El programa mostrará en las etiquetas el número de a, de e, de i de o y de u que se encuentran en la frase.
* Tenga en cuenta que puede haber vocales en mayúsculas y en minúsculas.
* Si el usuario no introduce nada en el cuadro de texto txtFrase, entonces el programa debería mostrar un error.



# Ejercicio 2

Realizar un programa cuya ventana tenga los siguientes elementos:

* Un cuadro de texto llamado txtDNI.
* Una etiqueta llamada etiDNI.
* Un botón “Preparar DNI” llamado btnPrepararDNI.



El programa funcionará de la siguiente forma:

* El usuario introducirá un DNI en el cuadro de texto llamado txtDNI y luego pulsará el botón Preparar DNI. El resultado será que el DNI introducido aparecerá “preparado” en la etiqueta etiDNI.
* A continuación, se explica cómo preparar el DNI:

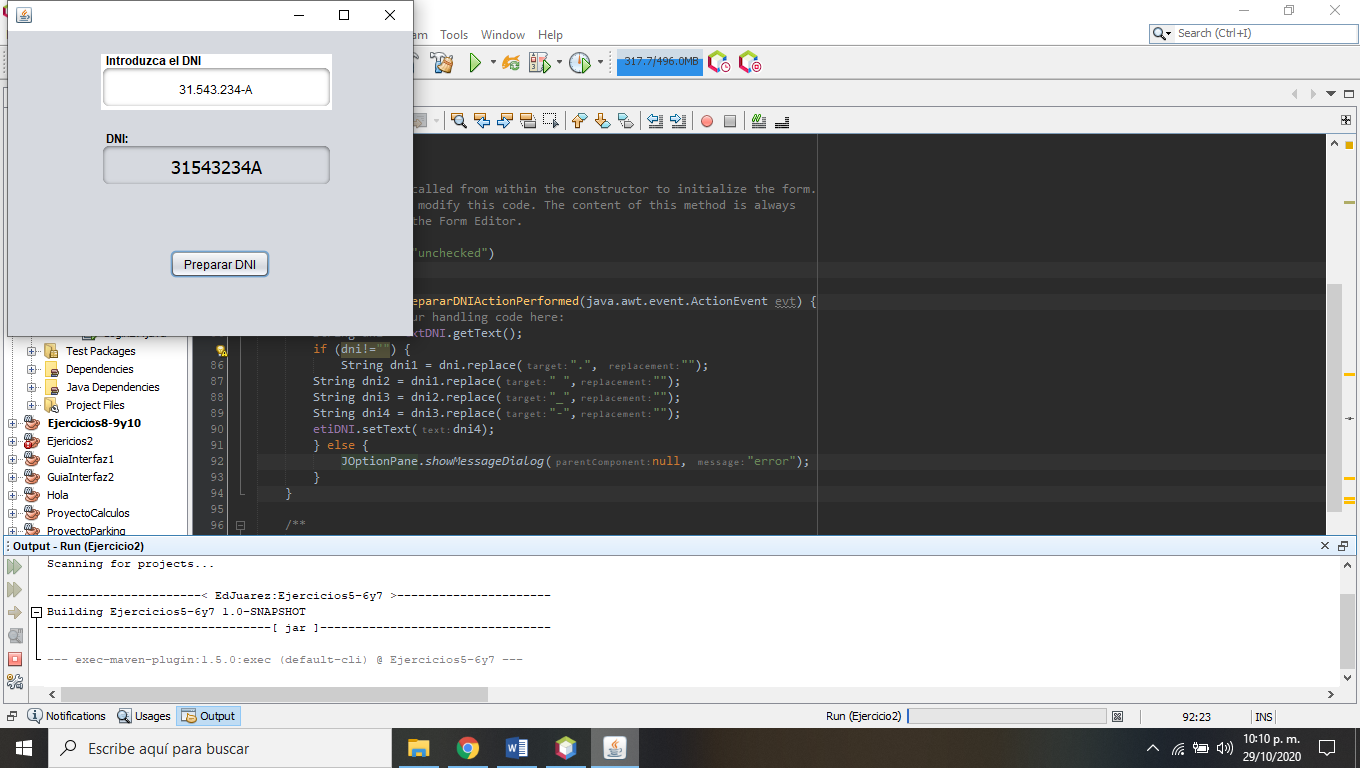
El usuario puede introducir un DNI con uno de estos formatos: 31.543.234-A

31.543.234 A 31.543.234A 31 543 234 A etc.

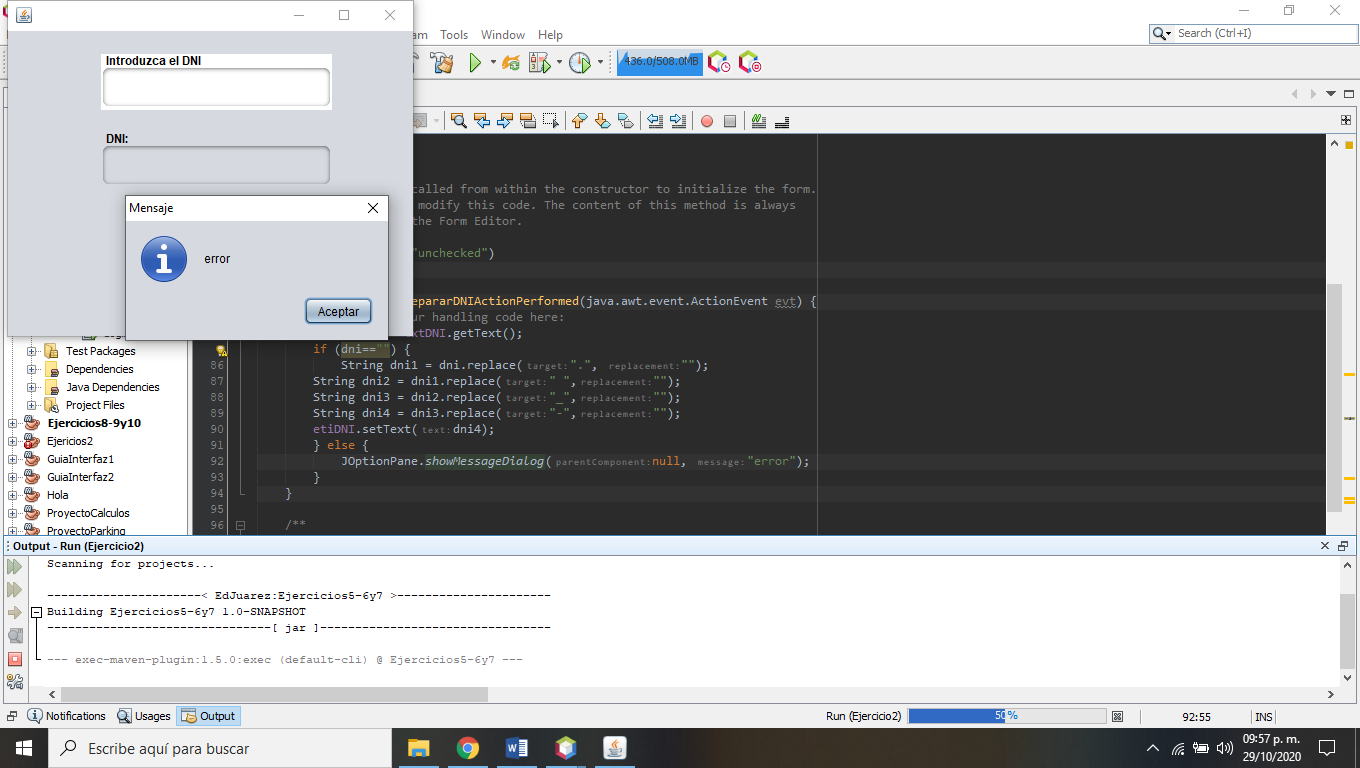
Sin embargo, cuando el usuario pulse el botón Preparar DNI, en la etiqueta etiDNI debe aparecer el DNI con el siguiente formato:

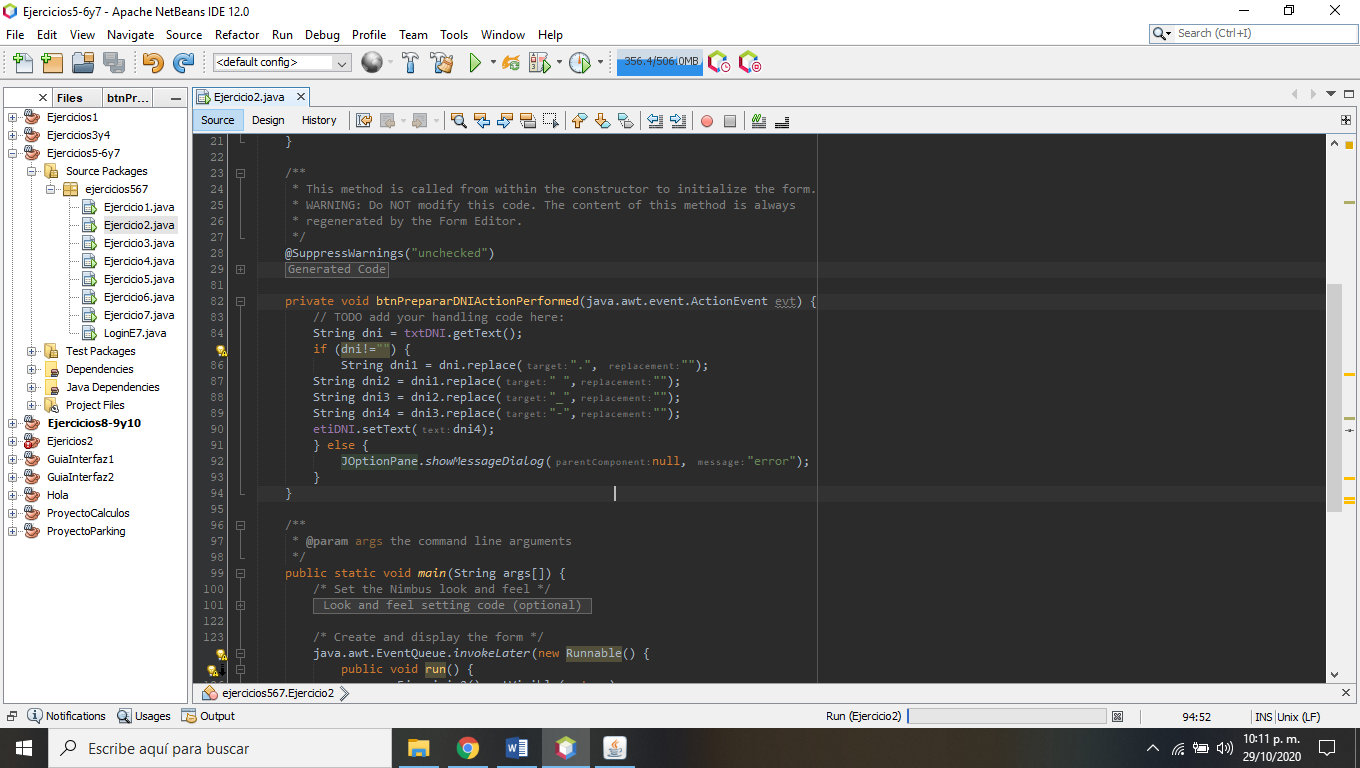
31543234A

Es decir, sin ningún espacio y sin puntos ni guiones.



* Si el usuario no introduce nada en el cuadro de texto del DNI y pulsa el botón, entonces debe aparecer un error emergente (JOptionPane).





# Ejercicio 3

Se pide hacer un programa que le permita al usuario introducir una palabra en latín de la primera declinación, y a continuación generar sus “casos” en plural y singular.

Las palabras de la primera declinación en latín son sustantivos femeninos (la mayoría), que terminan en a, como, por ejemplo: ROSA o ANIMA.

Estas palabras tienen las siguientes variantes o “casos”:

Por ejemplo, para la palabra ROSA, sus casos son los siguientes:

|  |  |  |
| --- | --- | --- |
| **CASO** | **SINGULAR** | **PLURAL** |
| NOMINATIVO | Rosa | Rosae |
| VOCATIVO | Rosa | Rosae |
| ACUSATIVO | Rosam | Rosas |
| GENITIVO | Rosae | Rosarum |
| DATIVO | Rosae | Rosis |
| ABLATIVO | Rosa | Rosis |

Por ejemplo, para la palabra ANIMA, sus casos son los siguientes:

|  |  |  |
| --- | --- | --- |
| **CASO** | **SINGULAR** | **PLURAL** |
| NOMINATIVO | Anima | Animae |
| VOCATIVO | Anima | Animae |
| ACUSATIVO | Animam | Animas |
| GENITIVO | Animae | Animarum |
| DATIVO | Animae | Animis |
| ABLATIVO | Anima | Animis |

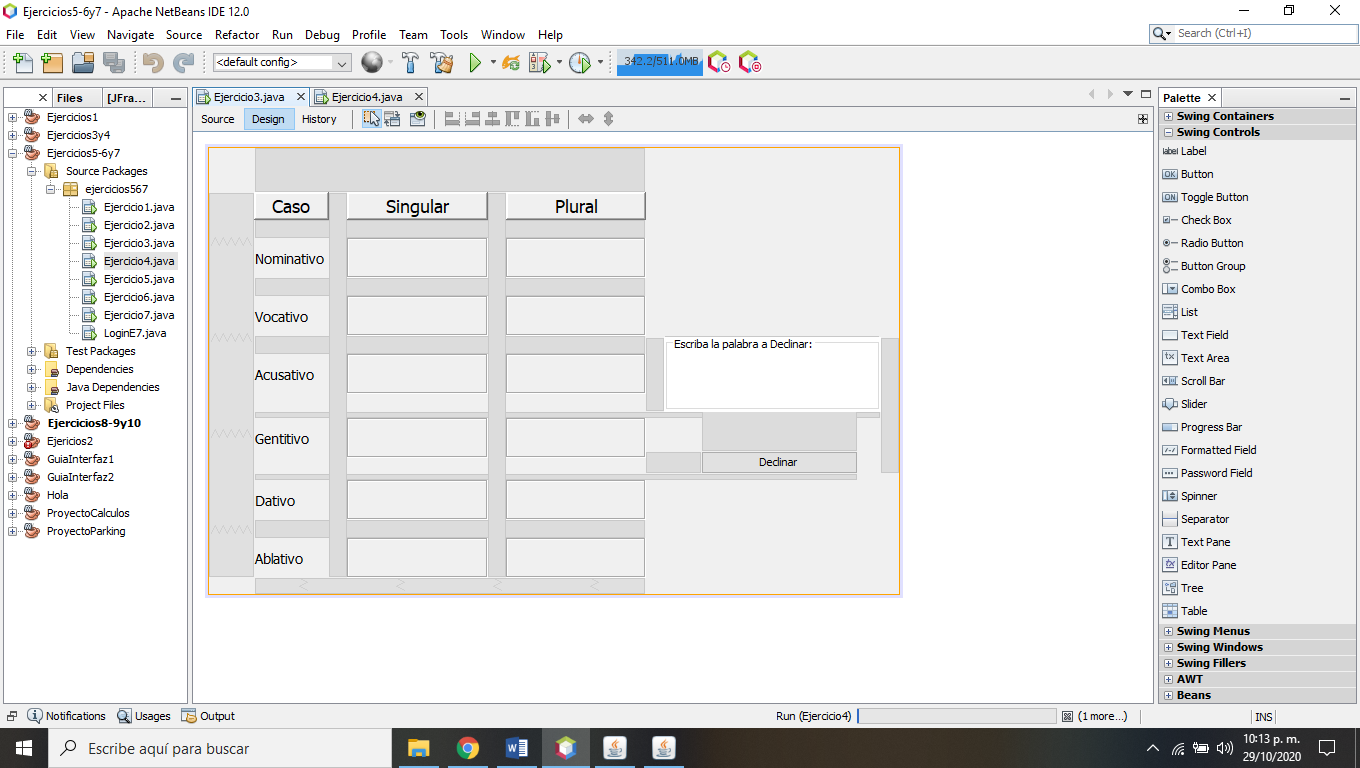
Debes observar que algunos casos son exactamente iguales a la palabra inicial, como por ejemplo el Nominativo Singular.

Otros casos, en cambio, se construyen añadiendo algunas letras al final de la palabra inicial. Por ejemplo, el Acusativo singular se construye añadiendo una “m” a la palabra inicial.

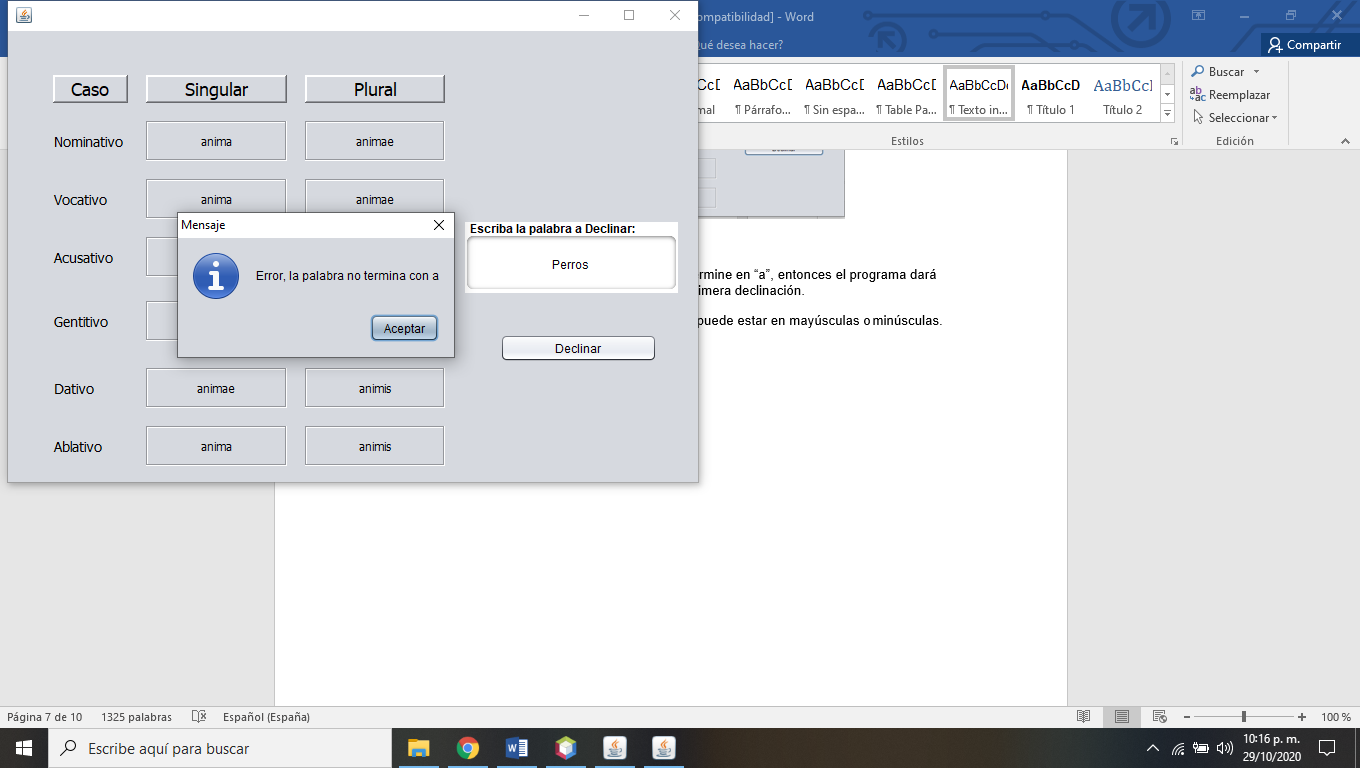
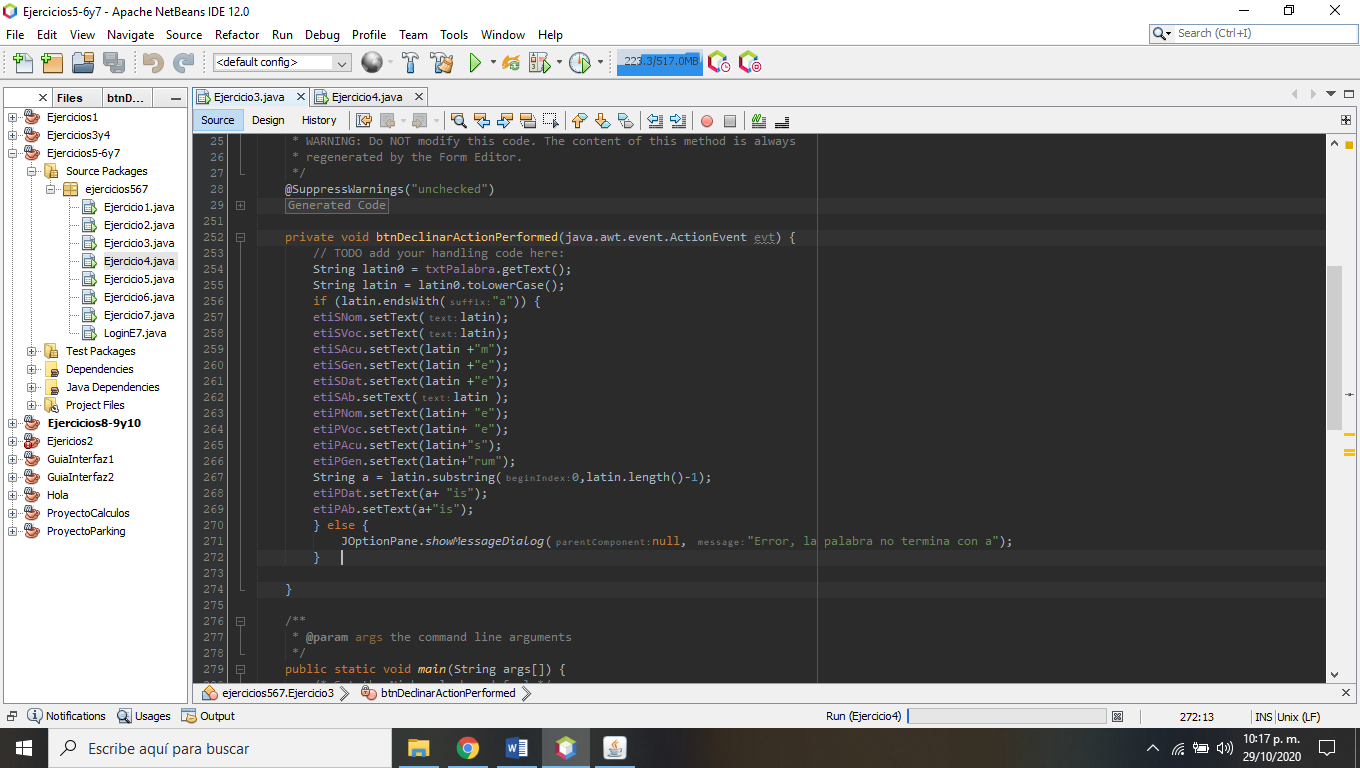
Para construir el Dativo y Ablativo plural, es necesario concatenar la palabra inicial (sin la a final) con “is”. Por ejemplo, en el caso de la palabra Rosa, se concatenaría: Ros + is.

TENIENDO EN CUENTA LO ANTERIOR, SE PIDE REALIZAR EL SIGUIENTE PROGRAMA:

Realice un programa cuya ventana tenga los siguientes elementos:

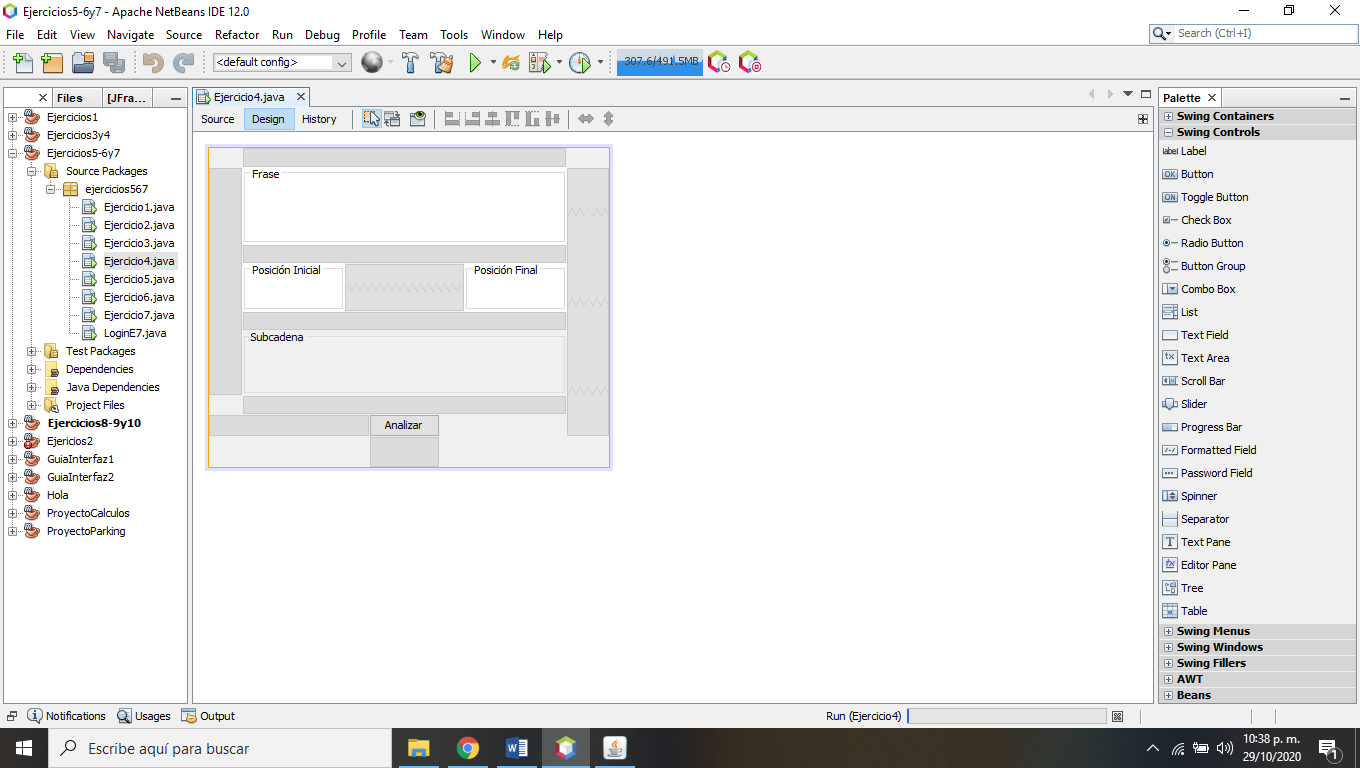
* Un cuadro de texto txtPalabra.
* Doce etiquetas al menos correspondientes a los 6 casos en singular y plural.
* Un botón “Declinar” llamado btnDeclinar.

El programa funcionará de la siguiente forma:

* El usuario introducirá una palabra en latín de la primera declinación en el cuadro de texto, y luego pulsará el botón “Declinar”. Entonces en las etiquetas aparecerán los casos declinados de la palabra.
* Si el usuario introduce una palabra que no termine en “a”, entonces el programa dará un error, ya que dicha palabra no es de la primera declinación.
* Tenga en cuenta que la palabra introducida puede estar en mayúsculas o minúsculas.

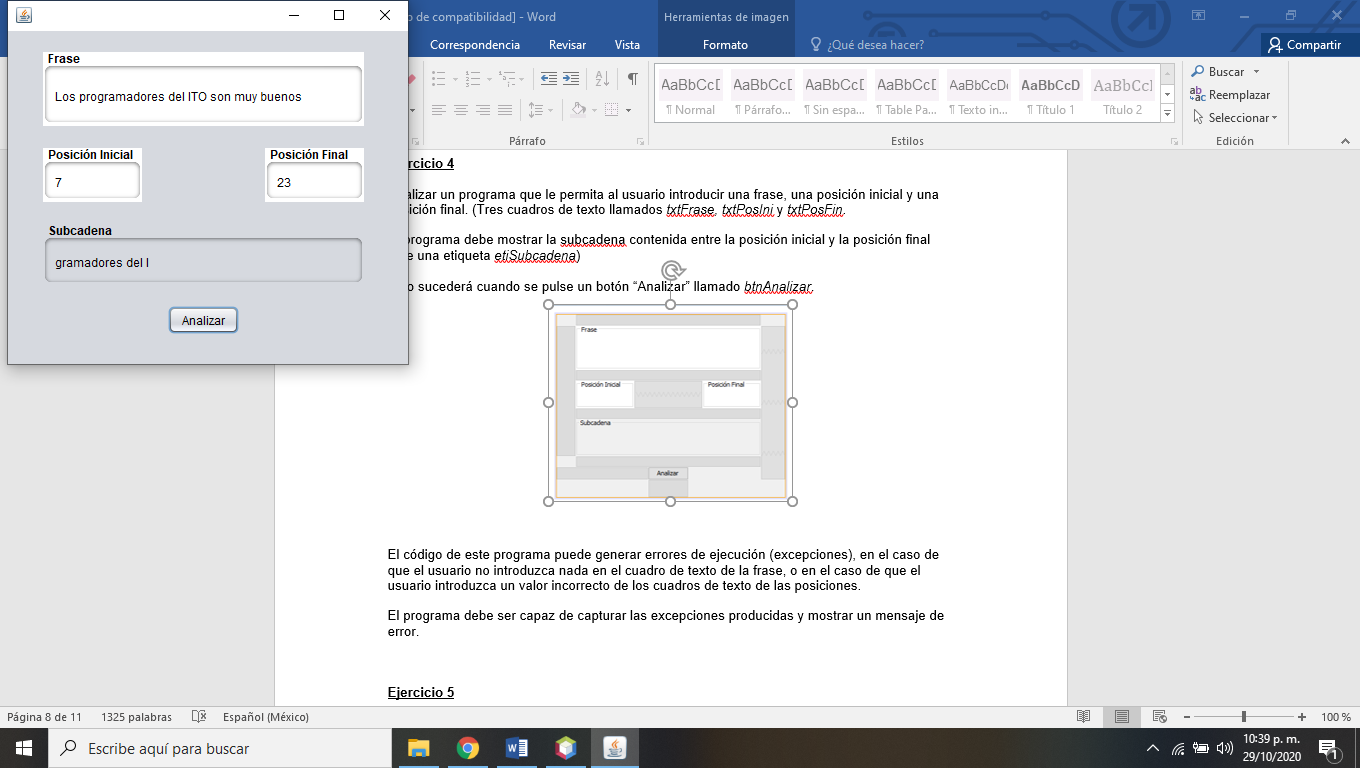
# EXCEPCIONES PDF #6

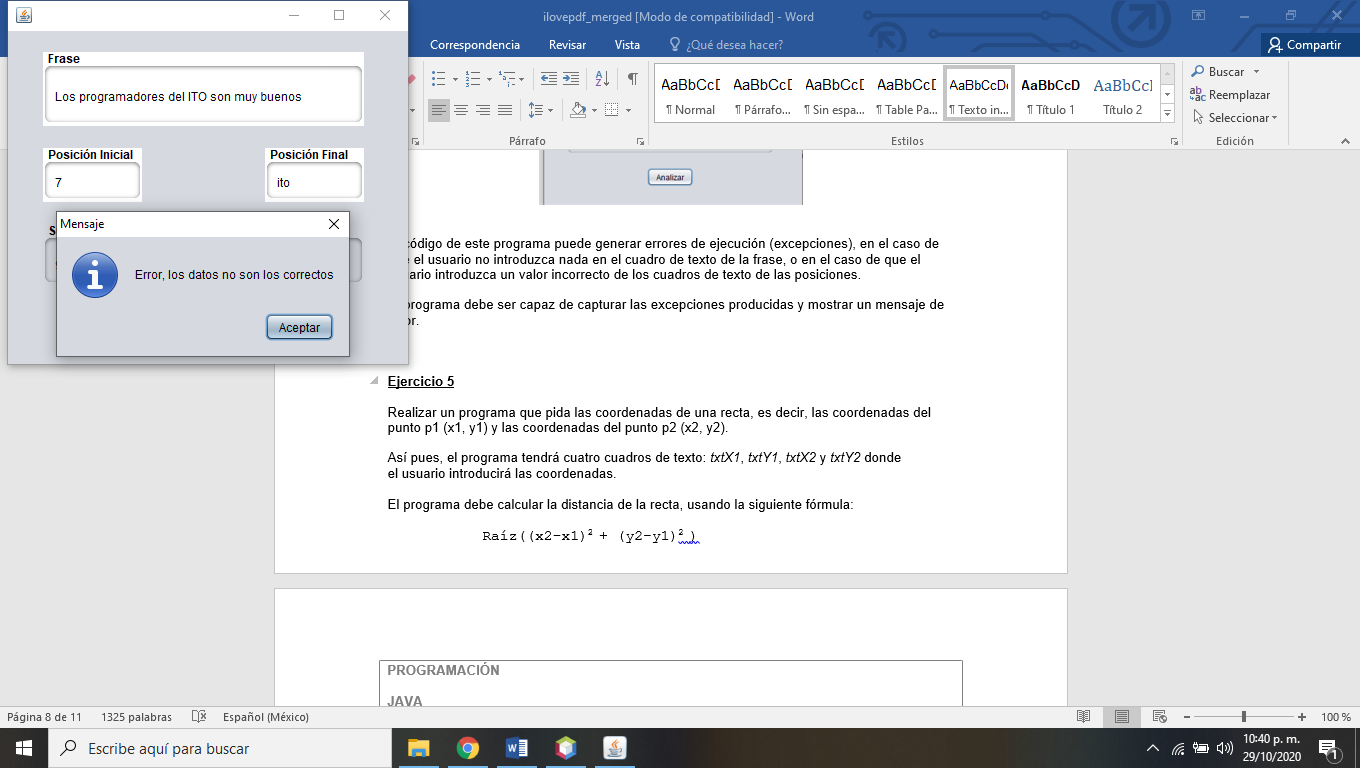
**Ejercicio 4**

Realizar un programa que le permita al usuario introducir una frase, una posición inicial y una posición final. (Tres cuadros de texto llamados *txtFrase, txtPosIni* y *txtPosFin*.

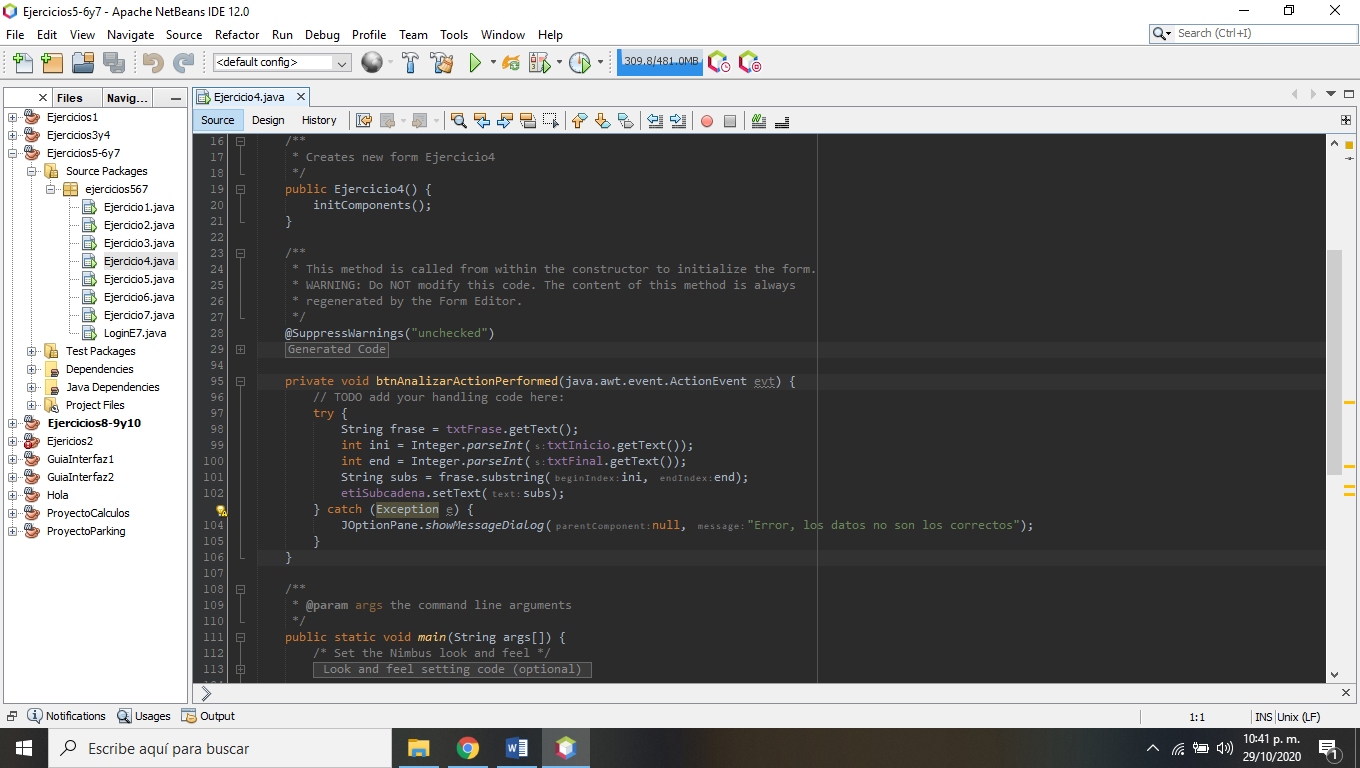
El programa debe mostrar la subcadena contenida entre la posición inicial y la posición final (use una etiqueta *etiSubcadena*)

Esto sucederá cuando se pulse un botón “Analizar” llamado *btnAnalizar*.



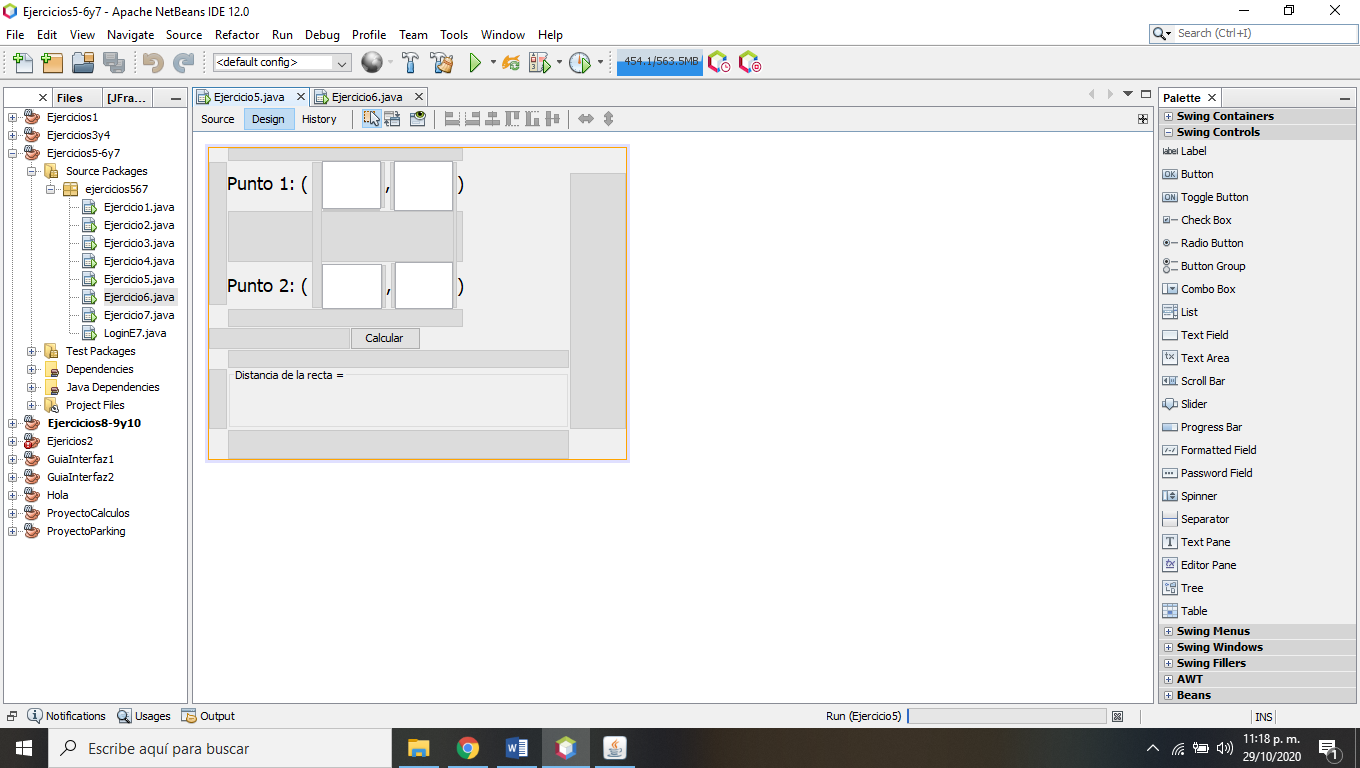
El código de este programa puede generar errores de ejecución (excepciones), en el caso de que el usuario no introduzca nada en el cuadro de texto de la frase, o en el caso de que el usuario introduzca un valor incorrecto de los cuadros de texto de las posiciones.

El programa debe ser capaz de capturar las excepciones producidas y mostrar un mensaje de error.

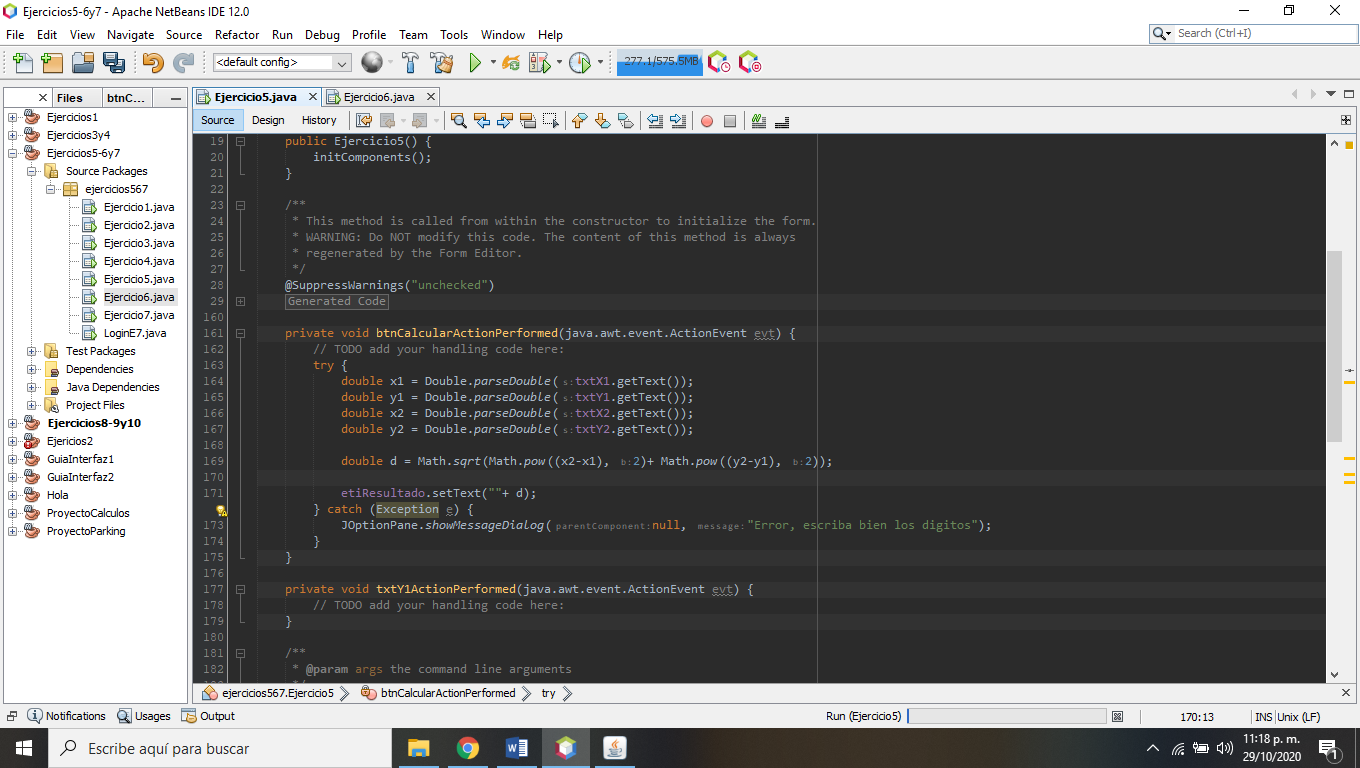


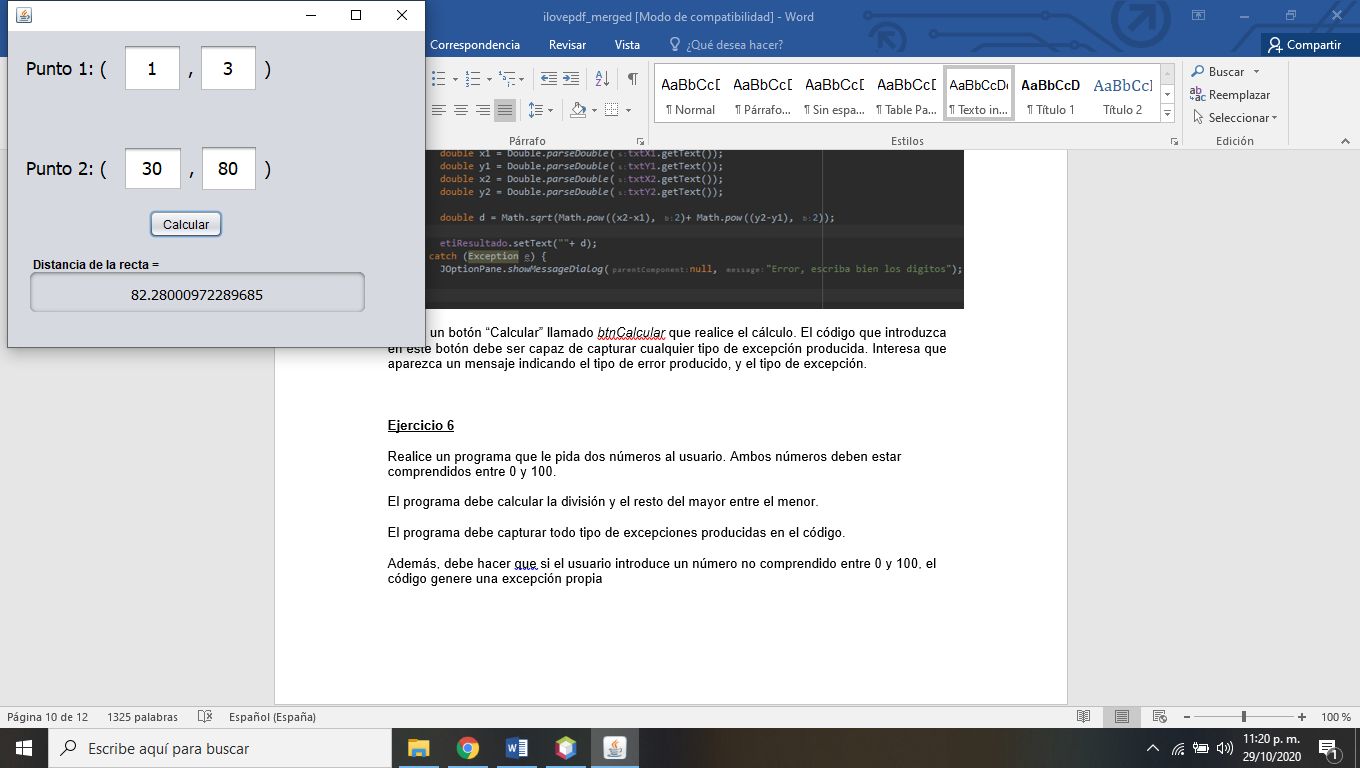
# Ejercicio 5

Realizar un programa que pida las coordenadas de una recta, es decir, las coordenadas del punto p1 (x1, y1) y las coordenadas del punto p2 (x2, y2).

Así pues, el programa tendrá cuatro cuadros de texto: *txtX1*, *txtY1*, *txtX2* y *txtY2* donde el usuario introducirá las coordenadas.

El programa debe calcular la distancia de la recta, usando la siguiente fórmula:

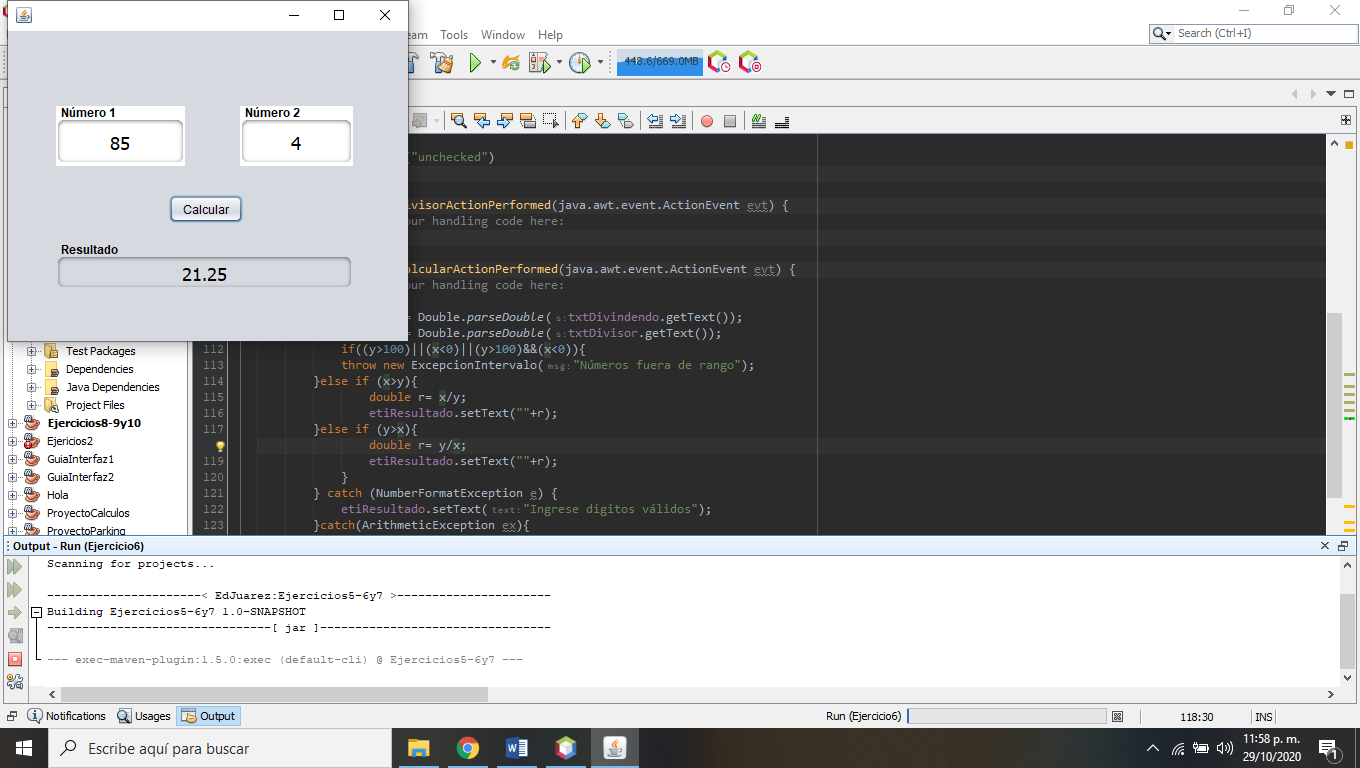
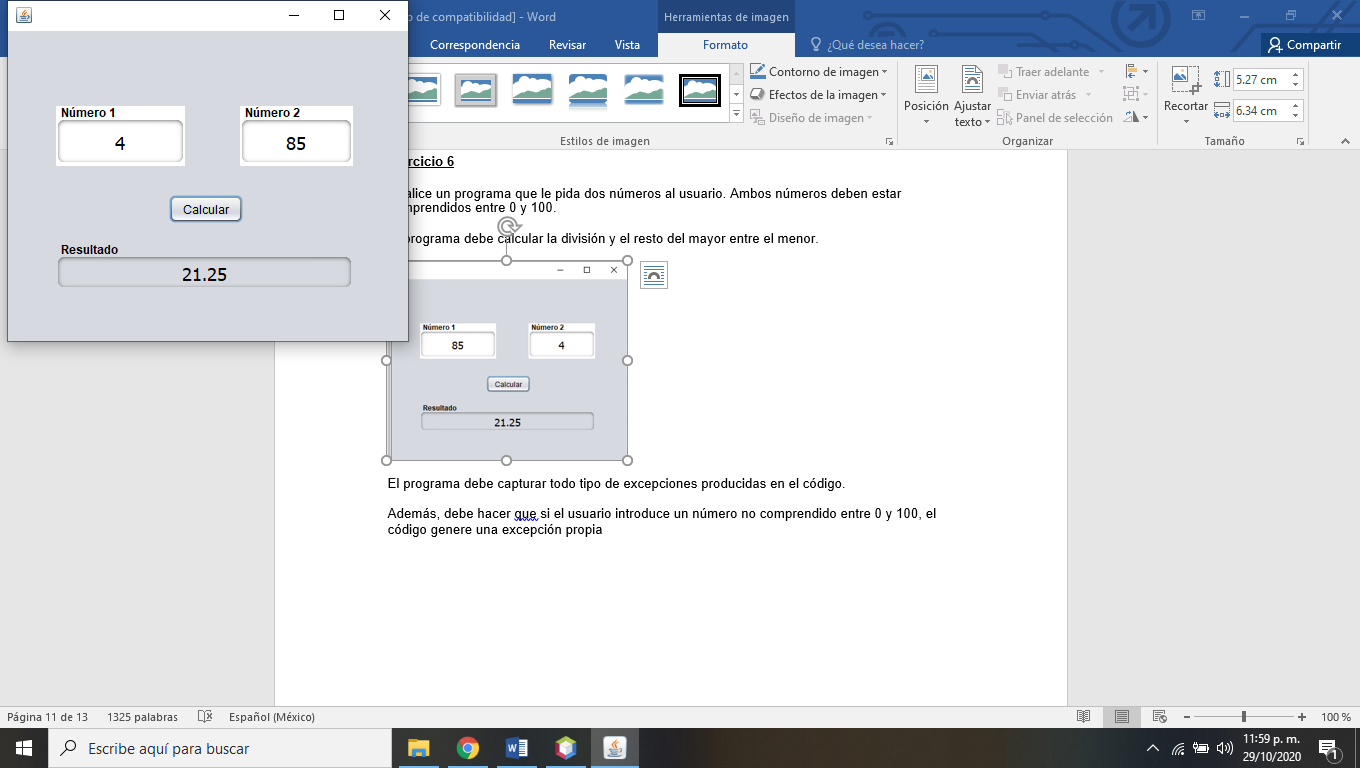
Raíz((x2-x1)2 + (y2-y1)2 )

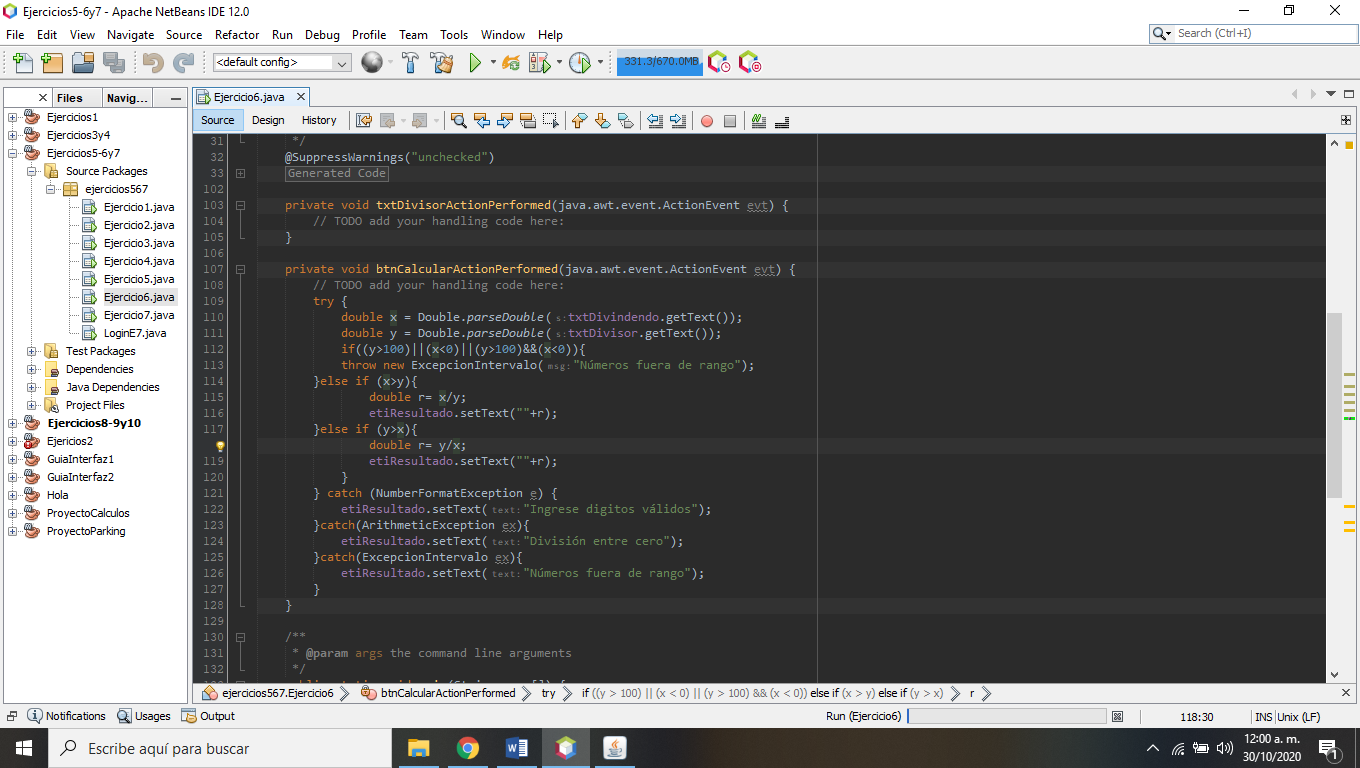
Añada un botón “Calcular” llamado *btnCalcular* que realice el cálculo. El código que introduzca en este botón debe ser capaz de capturar cualquier tipo de excepción producida. Interesa que aparezca un mensaje indicando el tipo de error producido, y el tipo de excepción.

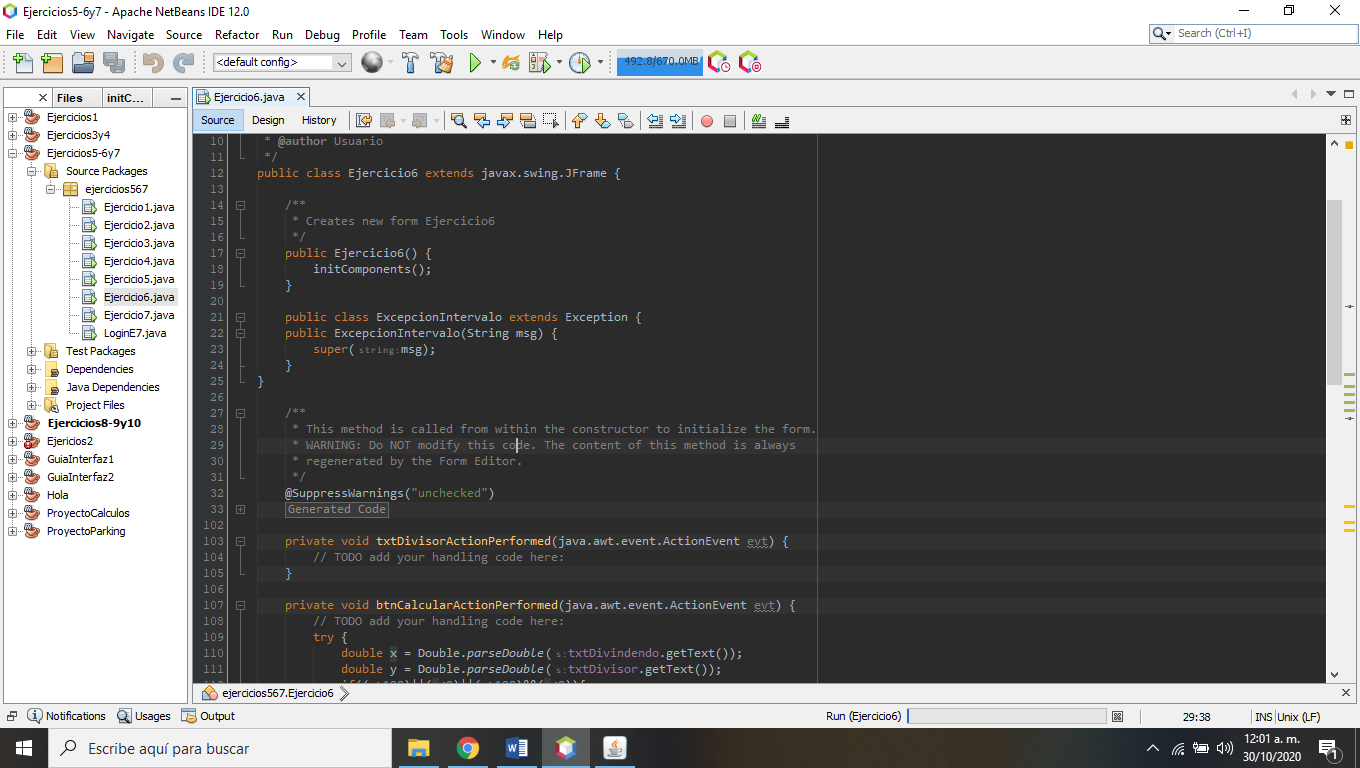
# Ejercicio 6

Realice un programa que le pida dos números al usuario. Ambos números deben estar comprendidos entre 0 y 100.

El programa debe calcular la división y el resto del mayor entre el menor.



El programa debe capturar todo tipo de excepciones producidas en el código.

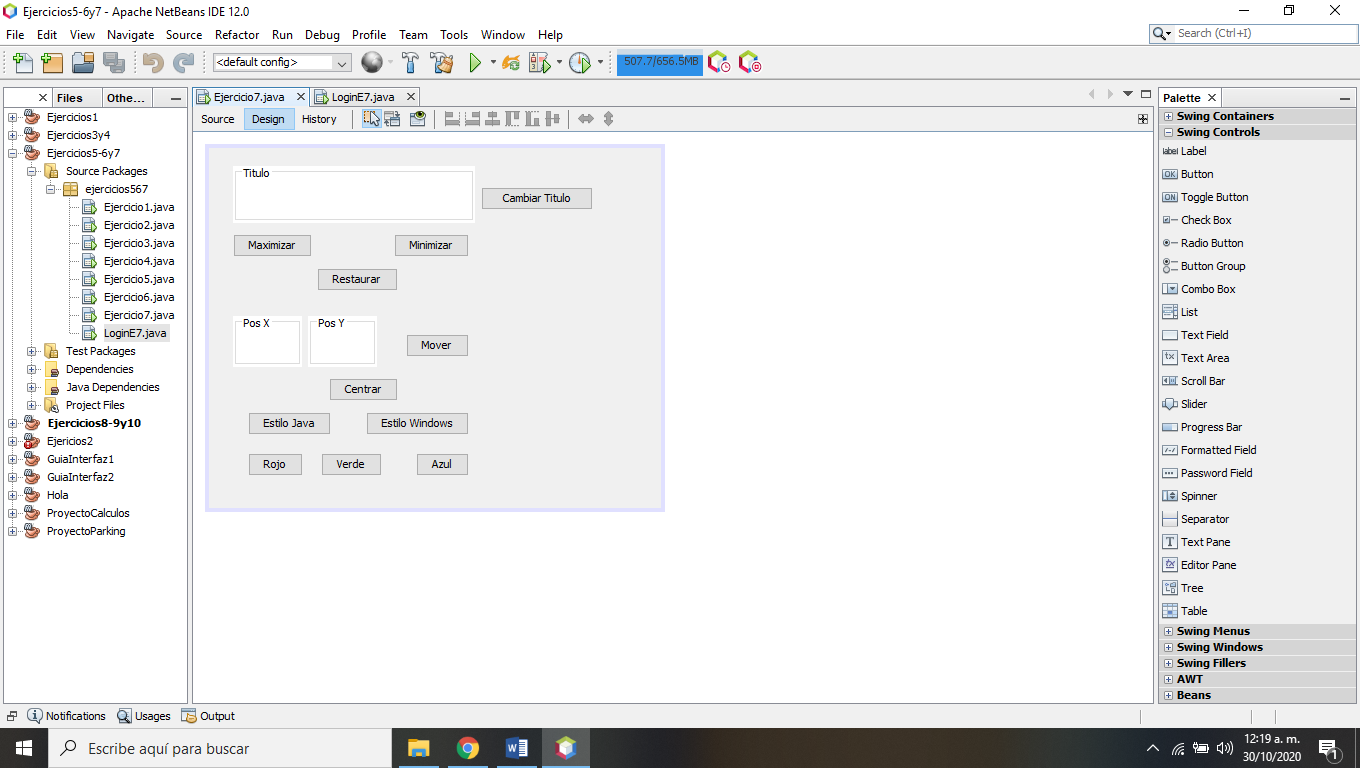
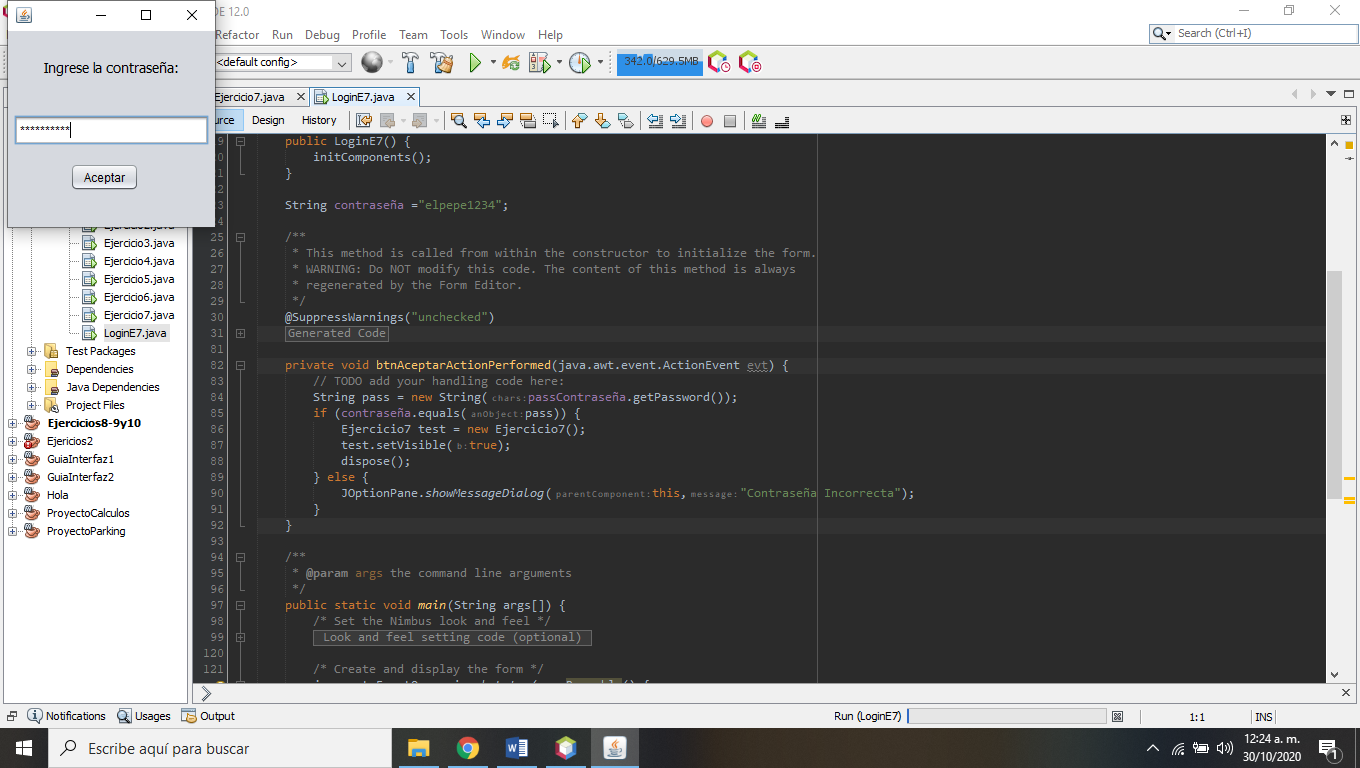
Además, debe hacer que, si el usuario introduce un número no comprendido entre 0 y 100, el código genere una excepción propia.

# LA CLASE JFRAME PDF #7

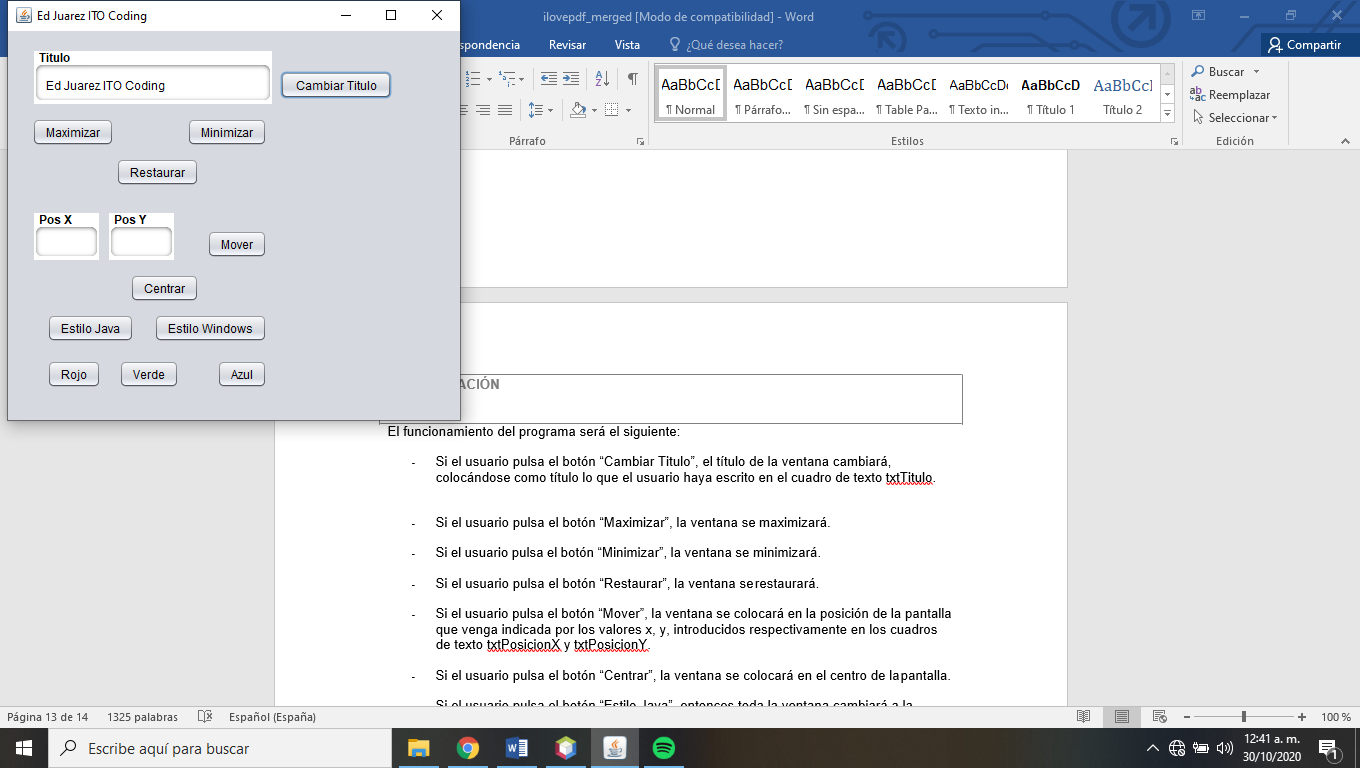
**Ejercicio 7**

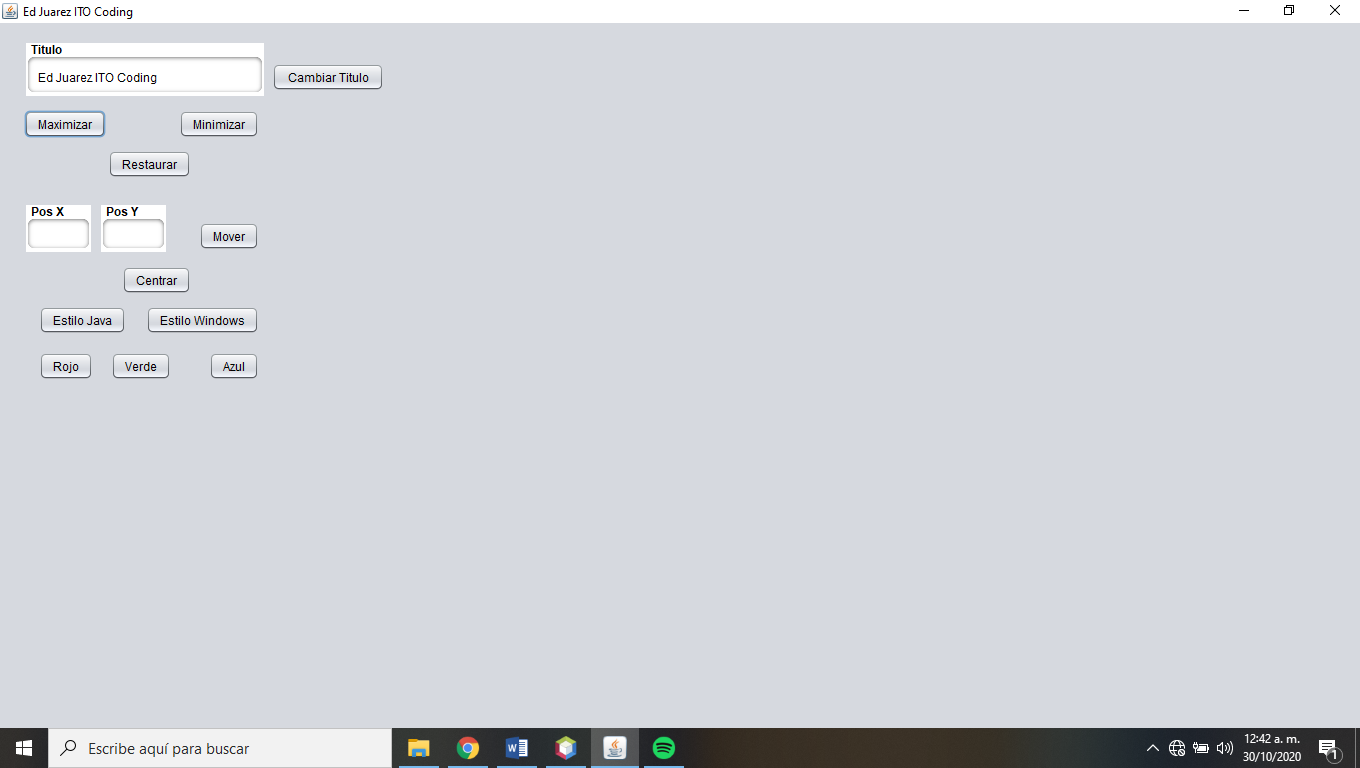
Realizar un programa cuya ventana contanga los siguientes elementos:

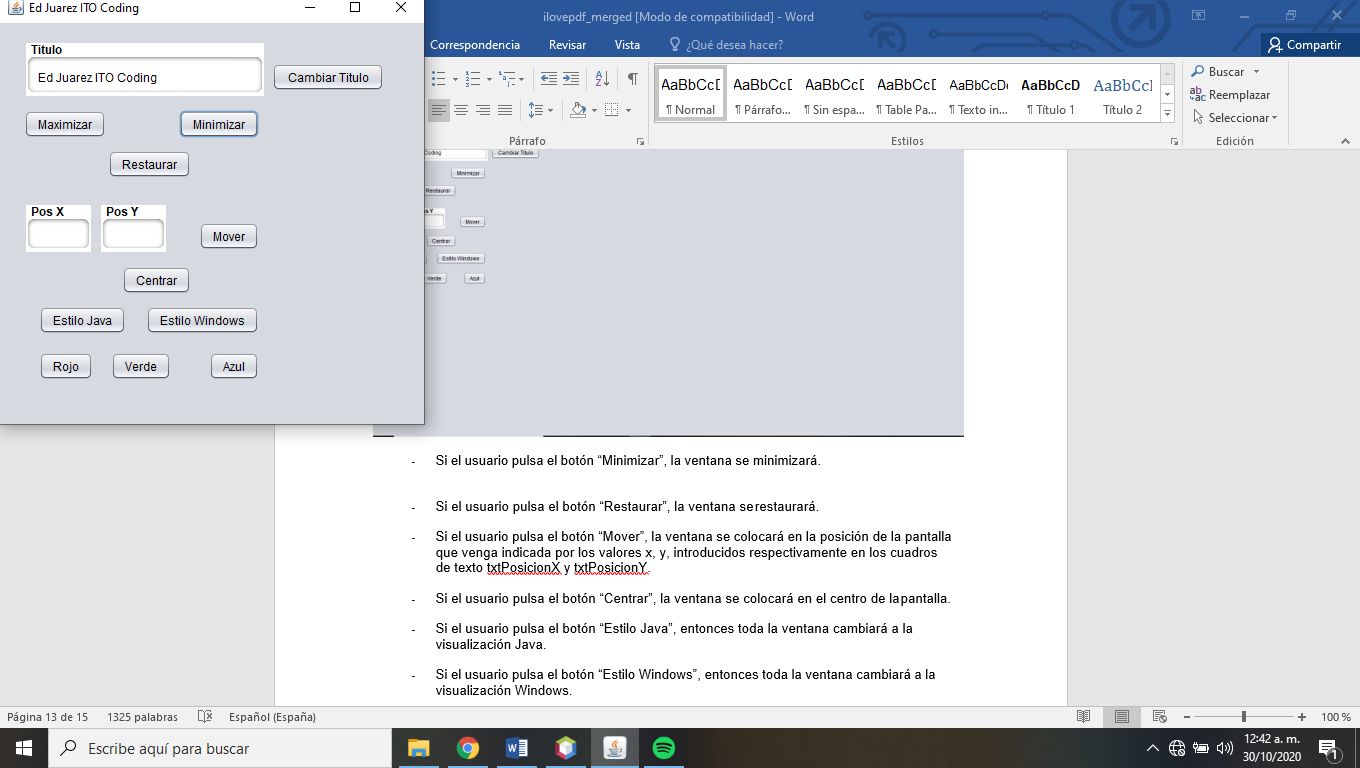
* Un cuadro de texto llamado txtTitulo.
* Un botón “Cambiar Título” llamado btnCambiarTitulo.
* Un botón “Maximizar” llamado btnMaximizar.
* Un botón “Minimizar” llamado btnMinimizar.
* Un botón “Restaurar” llamado btnRestaurar.
* Un cuadro de texto llamado txtPosicionX.
* Un cuadro de texto llamado txtPosicionY.
* Un botón “Mover” llamado btnMover.
* Un botón “Centrar” llamado btnCentrar.
* Un botón “Estilo Java” llamado btnEstiloJava.
* Un botón “Estilo Windows” llamado btnEstiloWindows.
* Un botón “Rojo” llamado btnRojo.
* Un botón “Verde” llamado btnVerde.
* Un botón “Azul” llamado btnAzul.

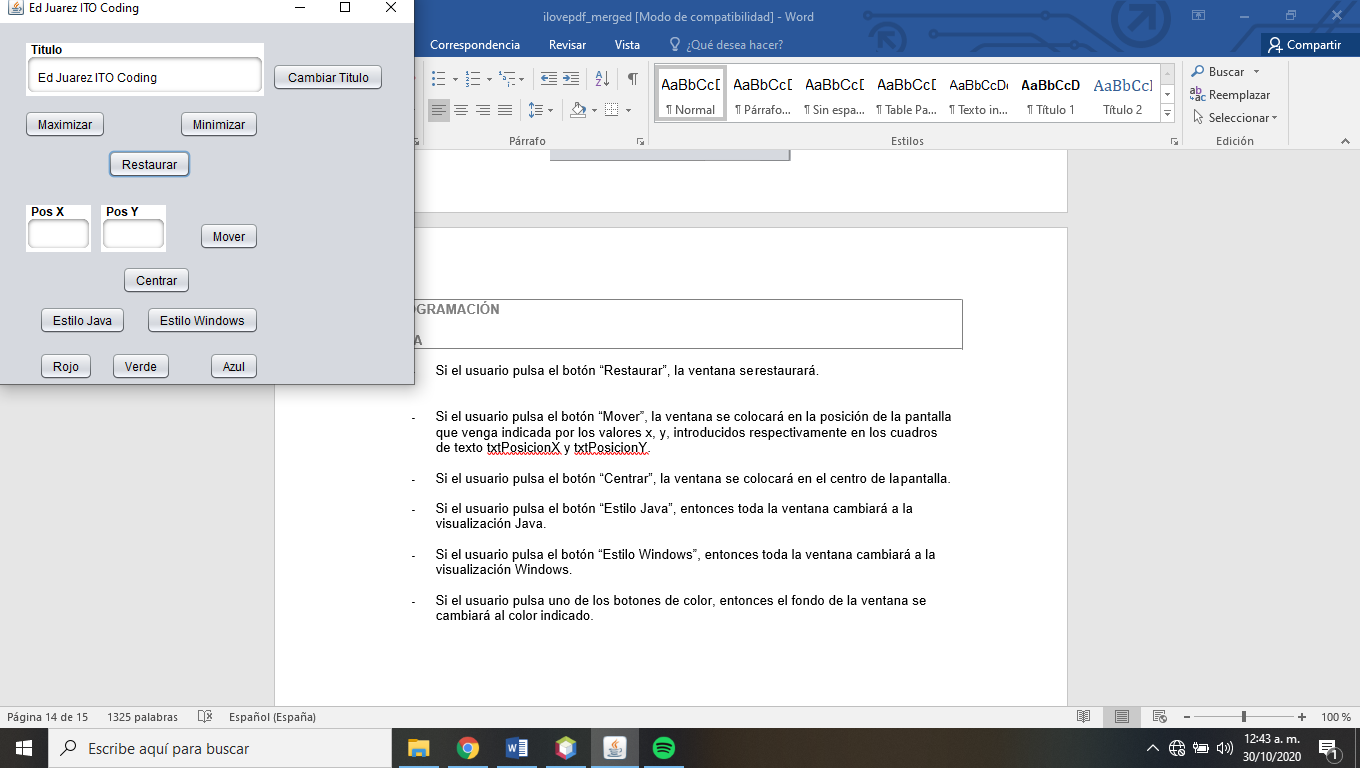


El funcionamiento del programa será el siguiente:

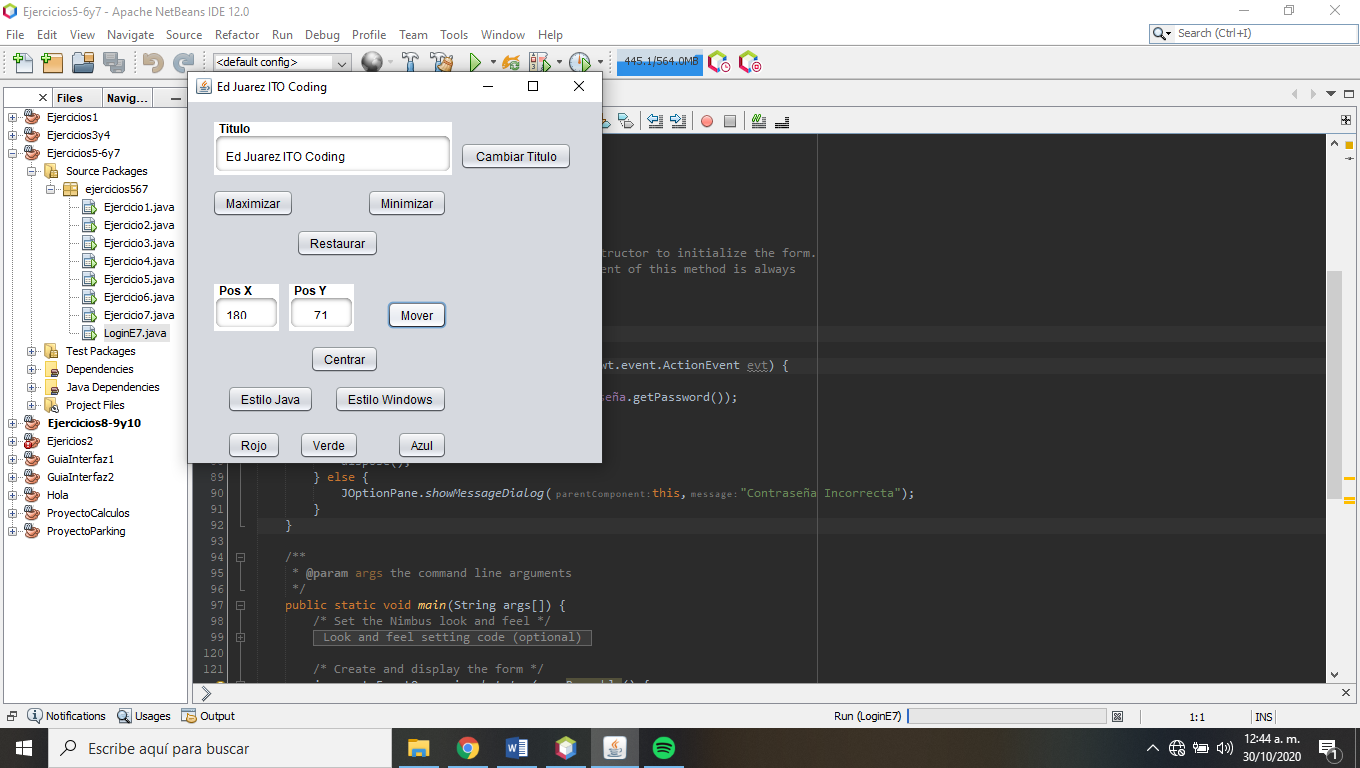
* Si el usuario pulsa el botón “Cambiar Titulo”, el título de la ventana cambiará, colocándose como título lo que el usuario haya escrito en el cuadro de texto txtTitulo.
* Si el usuario pulsa el botón “Maximizar”, la ventana se maximizará.



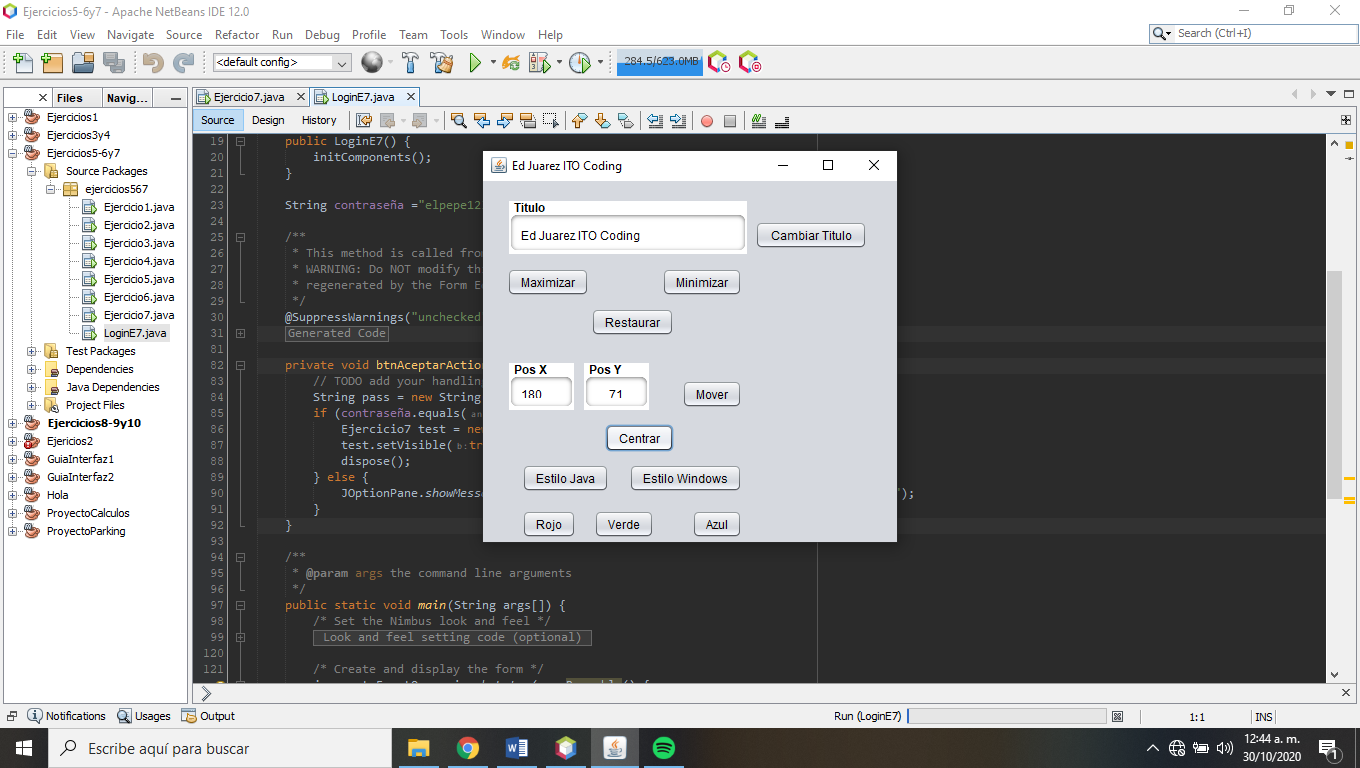
* Si el usuario pulsa el botón “Minimizar”, la ventana se minimizará.
* Si el usuario pulsa el botón “Restaurar”, la ventana se restaurará.



* Si el usuario pulsa el botón “Mover”, la ventana se colocará en la posición de la pantalla que venga indicada por los valores x, y, introducidos respectivamente en los cuadros de texto txtPosicionX y txtPosicionY.



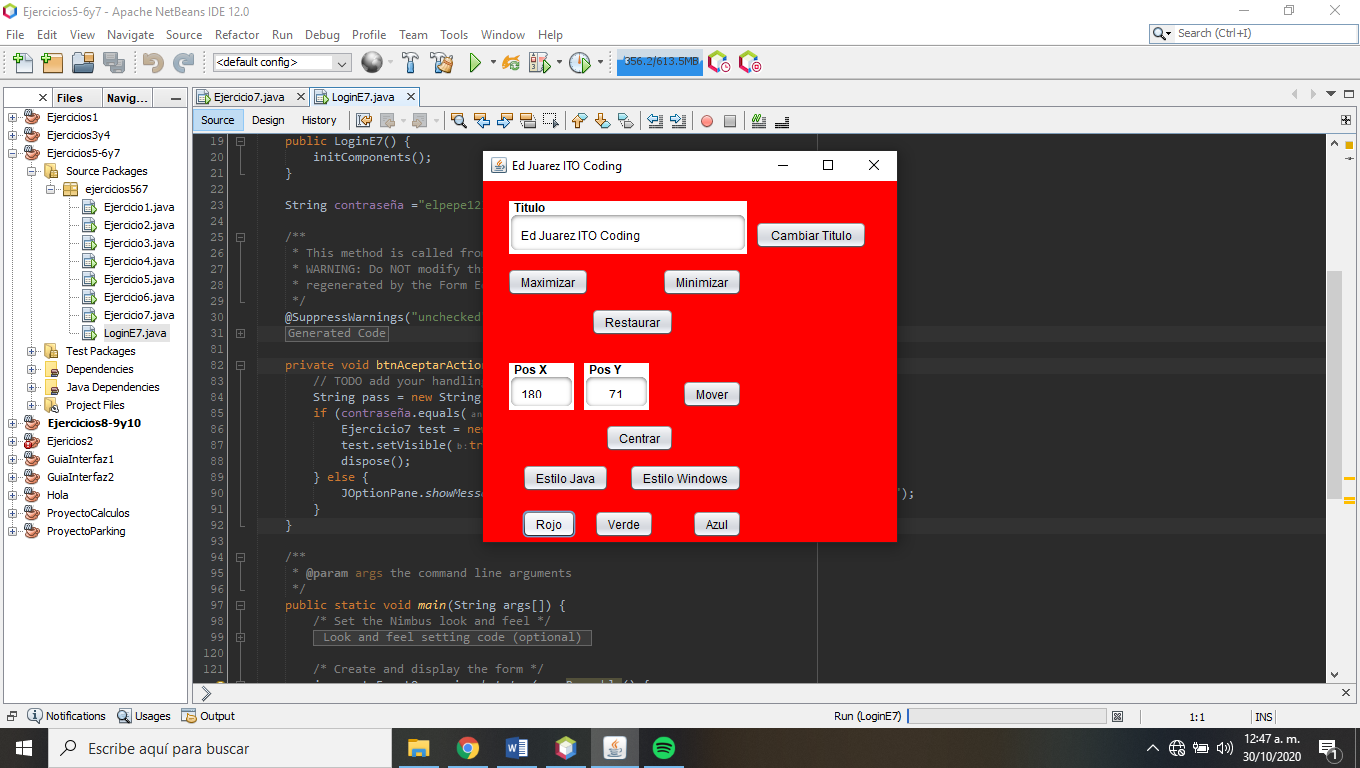
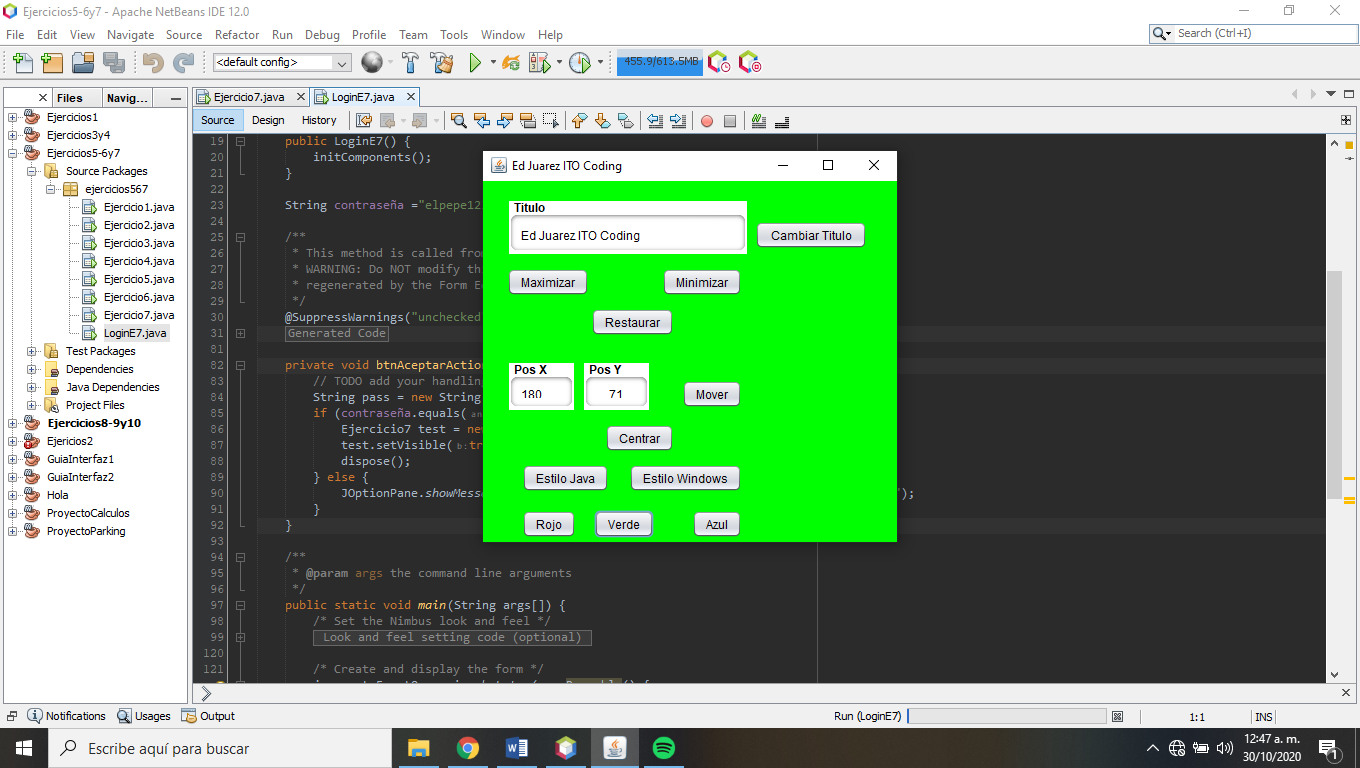
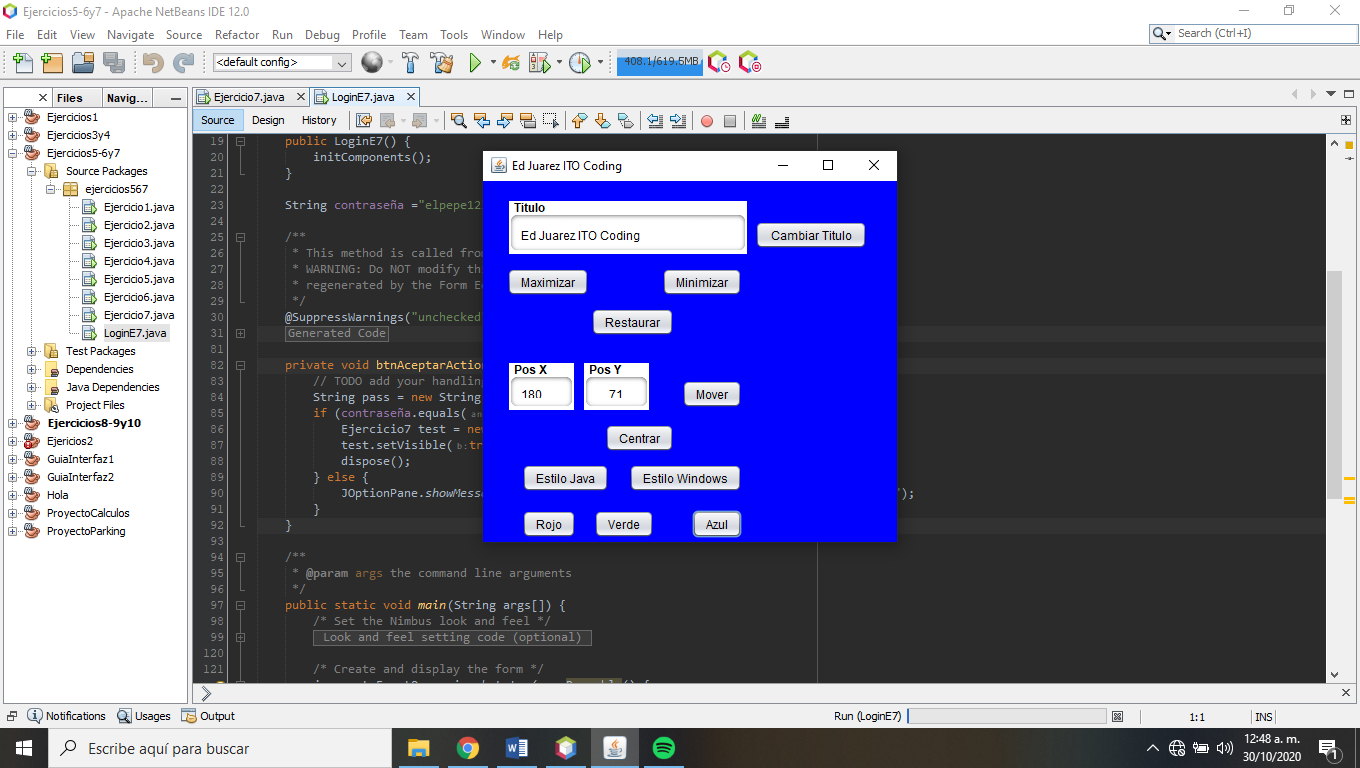
* Si el usuario pulsa el botón “Centrar”, la ventana se colocará en el centro de la pantalla.



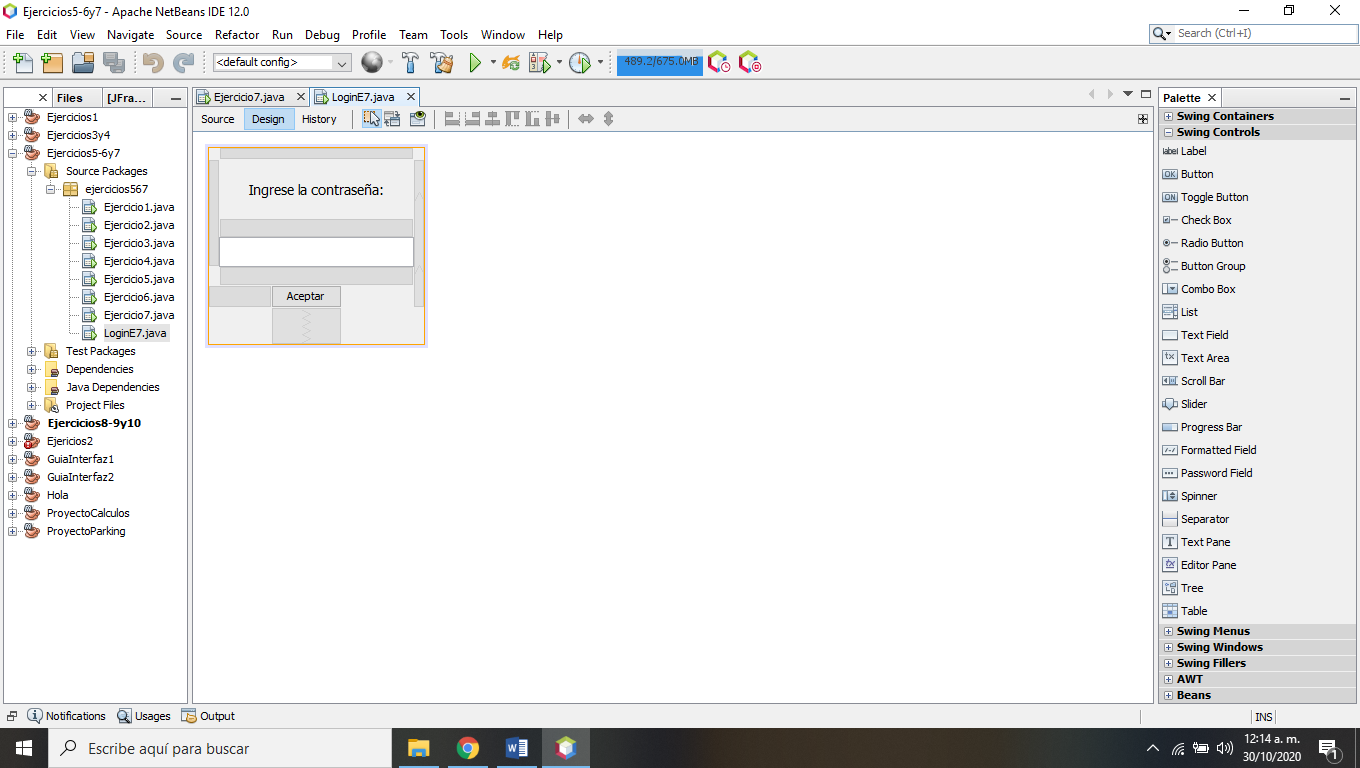
* Si el usuario pulsa el botón “Estilo Java”, entonces toda la ventana cambiará a la visualización Java.
* Si el usuario pulsa el botón “Estilo Windows”, entonces toda la ventana cambiará a la visualización Windows.

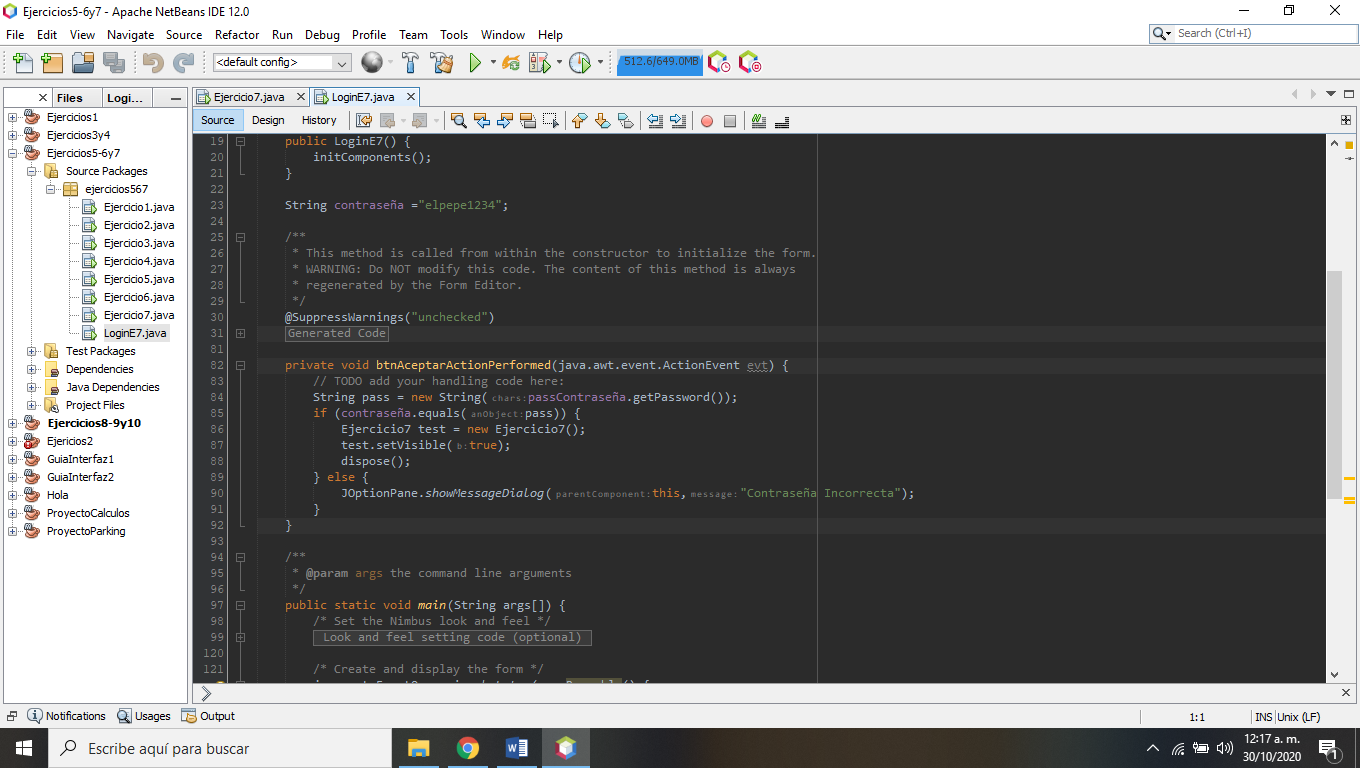
Este si no pude hacerlos con el LookandFeel lo intenté, pero no sé porque no cambiaba, de todas maneras ahí dejaré el código para revisarlo

* Si el usuario pulsa uno de los botones de color, entonces el fondo de la ventana se cambiará al color indicado.



Además, el programa contará con las siguientes características:

* Al iniciarse el programa, este debe pedir una contraseña al usuario. Si el usuario introduce la contraseña correcta, entonces el usuario entrará en el programa. En caso contrario el programa se cerrará.



* Al finalizar el programa, este debe pedir una confirmación al usuario:

¿Desea salir del programa?

Si el usuario responde Sí, entonces el programa finalizará. En caso contrario, el programa seguirá funcionando.

