Crear una aplicación, una aplicación Java contendrá uno o más proyectos con los recursos que solucionan el problema planteado.

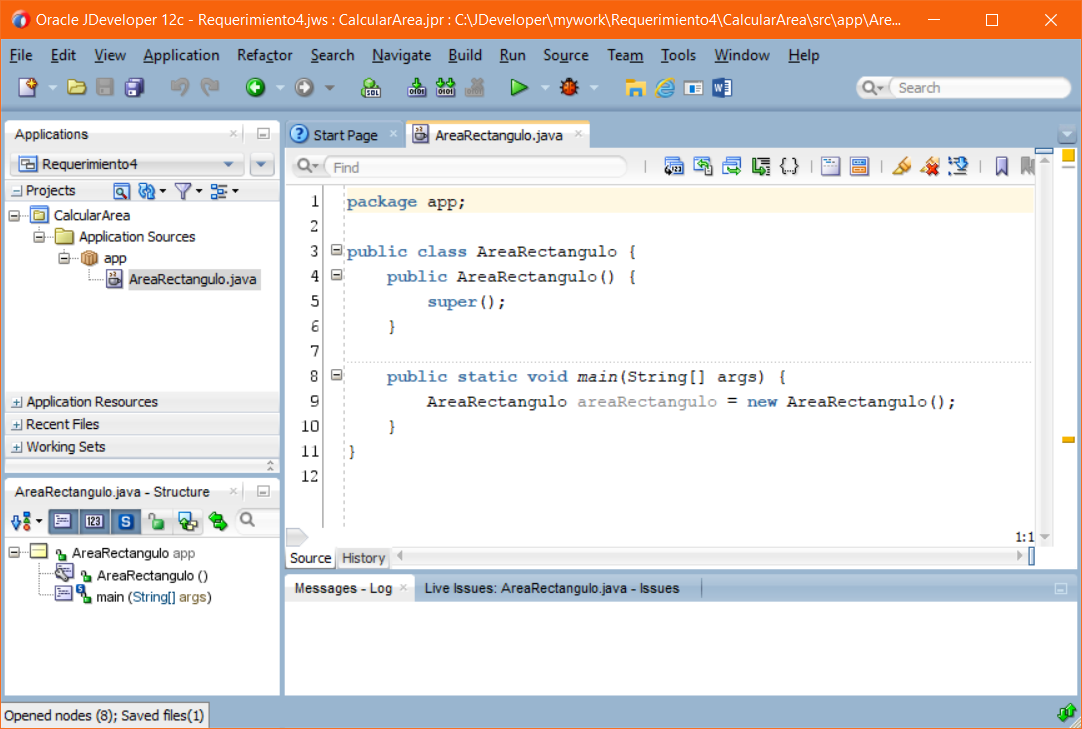
### Requerimiento del usuario para crear una aplicación Java:

Requerimiento 4: Crear el algoritmo para calcular el área de un rectángulo y un triángulo rectángulo de altura n metros y la base de n metros. Cuando el área supere los 789,5 metros cuadrados del rectángulo indicar que la superficie ha superado el valor permitido.

Nota: Realizar el programa usando clases y técnicas de la programación orientada a objetos.

Pasas para crear el programa:

1. Crear una aplicación llamada Requerimiento4
2. Crear un proyecto llamado CalcularArea
3. Crear una clase llamada AreaRectangulo
4. Continuar con los siguientes pasos que se detallan a continuación



Resultado de la tarea realizada

Agregar el siguiente código faltante al archivo AreaRectangulo.java

package app;

import java.util.Scanner;

public class AreaRectangulo {

public AreaRectangulo() {

super();

}

public static void main(String[] args) {

//AreaRectangulo areaRectangulo = new AreaRectangulo();

Scanner teclado = new Scanner(System.in);

Float base = teclado.nextFloat();

Float altura = teclado.nextFloat();

Float areaRectangulo = base \* altura;

Float areaTrianguloRec = areaRectangulo / 2;

System.out.println("Altura = " + altura);

System.out.println("base = " + base);

System.out.println("Área rectángulo = " + areaRectangulo);

System.out.println("Área triángulo rectángulo = " + areaTrianguloRec);

if(areaRectangulo.compareTo(789.5f) > 0){

System.out.println("El área de rectángulo es mayoe a: " + "789,5 m");

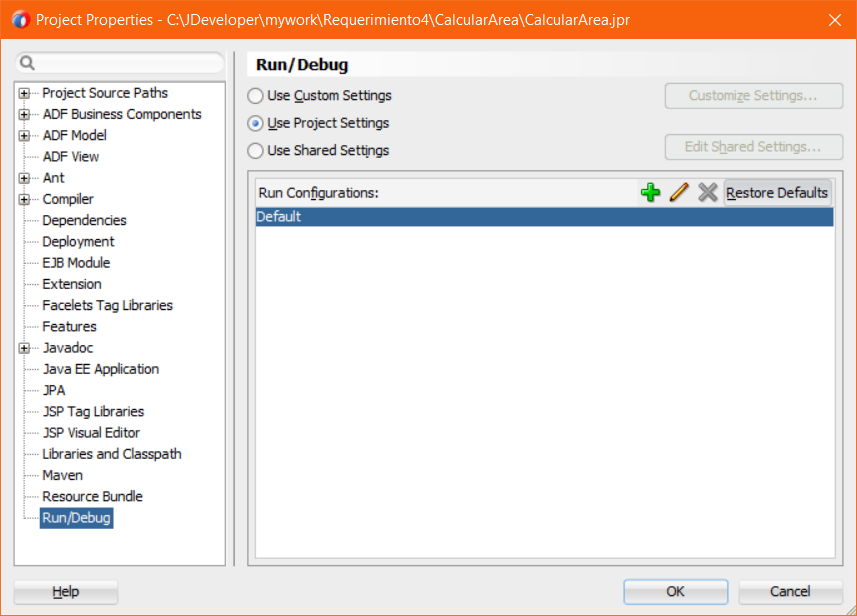
}

}

}

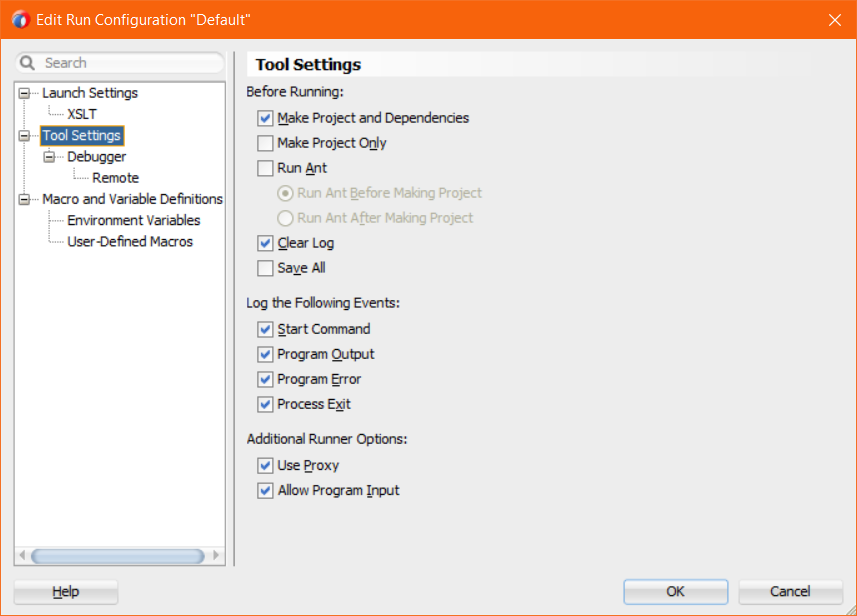
Activar la entrada por teclado en el JDeveloper

Hacer click con el botón derecho del ratón sobre el proyecto y seleccionar la opción ***Project Properties …*** y seleccionar las siguientes opciones en la ventana emergente.



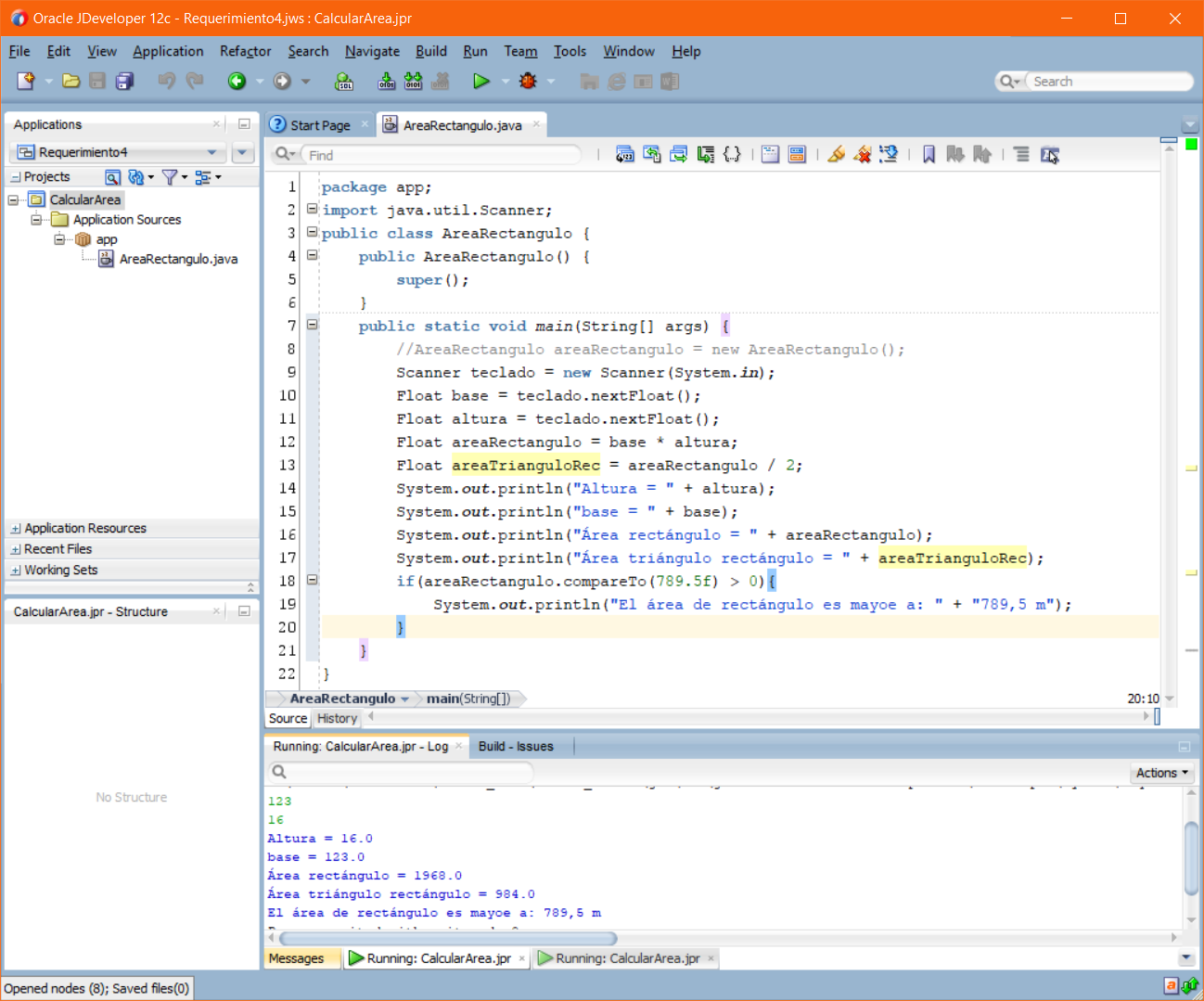
Seleccionar el lápiz

Seleccionar la opción ***Run/Debug***



Marcar la opción ***Allow Program Input*** en la configuración de ***Tool Settings***.

Correr el programa con el comando **Run** y verificar la salida correcta.



Resultado de la tarea realizada.

La línea de código: Float base = teclado.nextFloat(); declara una referencia de memoria llamada “base”, dicha referencia apuntará a un objeto del tipo Float entrado por teclado. El valor entrado por teclado es un objeto del tipo Float, y ocupará toda la memoria RAM necesaria para sus atributos y métodos.

Representación gráfica de la forma que Java almacena un objeto en memoria.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | base  14:33 |  |  | 45:56 |  |  |  |  |  |
| … | 45:56 |  |  | Objeto | Atributos y métodos del objeto del tipo Float | | |  | … |

La etiqueta “base” representa una dirección de memoria que tiene como contenido otra dirección de memoria que apunta al lugar de la memoria RAM donde se aloja la instancia de un objeto.

Representación gráfica de la siguiente línea de código: Float base = null;

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | base  14:33 |  |  | 45:56 |  |  |  |  |  |
| … | null |  |  | Objeto | Atributos y métodos del objeto del tipo Float | | |  | … |

La etiqueta “base” representa una dirección de memoria que no tiene asignado el valor de otra dirección de memoria. Cuando un objeto ha perdido su referencia el recolector de basura de la JVM limpiará la memoria de forma automática. Los programadores no deben preocuparse por fugas de memoria en Java.