

4.F. Actividad de refactorización en Eclipse

2. Solución a la actividad.

A continuación se muestran algunas indicaciones sobre como resolver la actividad.

	Opción a probar	Solución propuesta
1	Crear un proyecto Eclipse llamado "refactor".	Menú principal: Archivo/Nuevo/Proyecto/Proyecto Java. <i>Inglés: File / New / Java Project</i>
2	Crear un paquete "utilidades" y, dentro, una clase Circulo.	Pulsar el botón derecho del ratón sobre el proyecto refactor: Nuevo/paquete. Pulsar el botón derecho del ratón sobre el paquete utilidades: Nuevo/clase. <i>Inglés: New / Package</i> <i>Inglés: New / Class</i>
3	Crear un paquete "figuras" y, dentro, una clase Test.	Idem anterior. Con el código inicialmente ofrecido, el programa no compila.
4	Mediante las utilidades de "Código fuente" (<i>Inglés: Source</i>): <ul style="list-style-type: none"> • Generar métodos get y set para la clase Circulo. • Corregir la tabulación del código. • Dar formato al código. 	Generar métodos de obtención y establecimiento. Sangrado correcto. Formatear.
5	Mover la clase Circulo al paquete figuras.	Sobre el nombre de la clase, pulsar el botón derecho del ratón: refactorizar/mover. (<i>Inglés: Refactor / Move</i>) Continuar a pesar de la advertencia.
6	Renombrar la clase Circulo por Circunferencia. Observar si el cambio afecta a otras clases (en este caso Test).	Sobre el nombre, pulsar el botón derecho del ratón: refactorizar/renombrar. <i>Inglés: Refactor / Rename</i>
7	Renombrar el atributo "rad" por "radio". ¿Cómo afecta al método get?.	Actualiza el parámetro devuelto, pero no modifica la signatura del método.
8	Convertir la variable local "color" del método imprimir en un atributo, inicializando su valor en el mismo método imprimir.	Sobre la variable local, pulsar el botón derecho del ratón: refactorizar/convertir variable local en campo. <i>Inglés: Refactor / Convert Local Variable to Field</i>
9	En imprimir, en lugar de calcular y escribir el diámetro directamente en el println, extraer a una variable local "d" e imprimir dicha variable.	Sobre la variable area incluida en la instrucción pirntln, pulsar el botón derecho del ratón: refactorizar/extraer variable local. <i>Inglés: Refactor / Extract Local Variable</i>
10	Hacer que 3.1416 sea una constante llamada PI.	Sobre 3.1416, pulsar el botón derecho del ratón: refactorizar/extraer constante. <i>Inglés: Refactor / Extract Constant</i>
11	Extraer el cálculo del área a un método llamado calcularArea. No recibirá parámetros y devolverá un double.	Sobre la instrucción que realiza el cálculo , pulsar el botón derecho del ratón: refactorizar/extraer método <i>Inglés: Refactor / Extract Method</i>
12	Cambiar la firma o cabecera del método esIguar, invirtiendo el orden de los parámetros y cambiando el nombre de conDecimales por considerarDecimales. ¿Cómo afecta el cambio a la clase Test, en la que se usaba este método?.	Sobre la cabecera del método, pulsar el botón derecho del ratón: refactorizar/cambiar signatura del método. En los sitios donde es usado ha cambiado el orden de los parámetros automáticamente. <i>Inglés: Refactor / Change Method Signature</i>
13	Ahora usaremos "inline" para deshacer algunos cambios, es decir, hacer el código más concreto. Seleccionar la variable "d" (diámetro) y hacer que su valor se use en línea, desapareciendo por tanto la variable.	Sobre la variable d, pulsar el botón derecho del ratón: refactorizar/Incorporar. <i>Inglés: Refactor / Inline</i>

14	Seleccionar la llamada al método calcularArea y hacer que su código se incorpore en la misma línea, desapareciendo la necesidad de usar el método (puedes borrar el método después).	Sobre la llamada al método, pulsar el botón derecho del ratón: refactorizar/Incorporar. <i>Inglés: Refactor / Inline</i>
15	Seleccionar la constante PI y hacer que su valor se incorpore a las líneas en que se usa, desapareciendo por tanto la constante.	Sobre la aparición de PI en la función, pulsar el botón derecho del ratón: refactorizar/Incorporar. <i>Inglés: Refactor / Inline</i>
16	Intenta hacer la misma operación con el atributo "color". ¿Es posible eliminar un atributo y utilizar su valor en línea?	No, porque se trata de un valor variable que puede cambiar en otras partes del código, al contrario que los casos anteriores.