

UD4 - Actividades 1 – Referencias estáticas
--

Actividad 0. Proyecto

- a) Crea un proyecto con el nombre “ActividadesReferenciasEstaticas” para realizar todas las actividades de este documento.

Actividad 1. Queremos crear una calculadora básica con una clase Java para realizar cálculos sencillos.

- a) Define una clase CalculadoraBasica en un fichero java con los siguientes atributos privados estáticos:
- operador uno (int)
 - operador dos (int)
 - operación (String o char)
 - resultado (double)
- b) Crea un método estático “sumar” que acepte dos parámetros y retorne el valor de sendos operandos sumados.
- c) Crea un método estático “restar” que acepte dos parámetros y retorne el valor de sendos operandos restados.
- d) Crea un método estático “multiplicar” que acepte dos parámetros y retorne el valor de sendos operandos multiplicados.
- e) Crea un método estático “dividir” que acepte dos parámetros y retorne el valor de la operación dividiendo siempre el mayor entre el menor.
- f) Crea una clase ejecutable “ActividadUno” y en el método “main”, realiza algunas operaciones con cada uno de los métodos anteriores para comprobar que el código funciona correctamente.
- g) Vuelve a la clase CalculadoraBasica y crea un método estático llamado “mostrarUltimaOperacion” en la que se debe mostrar por pantalla un String que indique la última operación realizada utilizando los atributos privados de la clase.
- h) Refactoriza cada uno de los métodos operatorios de la clase CalculadoraBasica para que guarde los operandos, el símbolo de la operación y el resultado obtenido.
- i) Accede al método “mostrarUltimaOperacion” desde la clase ejecutable tras cada una de las operaciones que hiciste en el apartado (f) para comprobar que se guardan correctamente los valores de la última operación realizada.

Actividad 2. Queremos crear una utilidad que nos permita calcular áreas y perímetros de figuras sencillas de dos dimensiones con una clase Java.

- a) Crea una clase Rectangulo en un fichero java con los métodos de acceso necesarios para los atributos enteros base y altura. Inicialízalos en el constructor.
- b) Crea una clase TrianguloRectangulo en un fichero java con los métodos de acceso necesarios para los atributos enteros base y altura. Inicialízalos en el constructor.
- c) Crea una clase Círculo en un fichero java con los métodos de acceso necesarios para el atributo entero radio. Inicialízalo en el constructor.
- d) Crea una clase utilidad llamada “CalculadoraDeAreas” en un fichero java.
- e) Crea en ella un método estático “calcularArea” que acepte como parámetro un objeto Rectangulo y devuelva el resultado del cálculo de su área.
- f) Sobrecarga el método estático “calcularArea” para que acepte como parámetro un objeto TrianguloRectángulo y devuelva el resultado del cálculo de su área.
- g) Vuelve a sobrecargar el método estático “calcularArea” para que acepte como parámetro un objeto Circulo y devuelva el resultado del cálculo de su área.
- h) Crea una clase utilidad llamada “CalculadoraDePerimetros” en un fichero java.
- i) Crea en ella un método estático “calcularPerimetro” que acepte como parámetro un objeto Rectangulo y devuelva el resultado del cálculo de su perímetro.
- j) Sobrecarga el método estático “calcularArea” para que acepte como parámetro un objeto TrianguloRectángulo y devuelva el resultado del cálculo de su perímetro.
- k) Vuelve a sobrecargar el método estático “calcularArea” para que acepte como parámetro un objeto Circulo y devuelva el resultado del cálculo de su perímetro.
- l) Crea una clase ejecutable llamada “ActividadDos” y en su método “main”, instancia al menos un objeto de cada figura geométrica.
- m) Utiliza la clase “CalculadoraDeAreas” para calcular el área de cada una de ellas y mostrar el valor por pantalla añadiendo el máximo de información posible. Por ejemplo, “El triangulo rectángulo de base 5 m y altura 2 m, tiene un área de 5m²”.
- n) Haz lo mismo para calcular el perímetro de las figuras instanciadas empleando la clase “CalculadoraDePerimetros”.
- o) Organiza toda la estructura de ficheros en una jerarquía de paquetes acorde con la naturaleza de cada clase.

Actividad 3. Queremos crear una calculadora avanzada con una clase Java para realizar cálculos más complejos.

- a) Define una clase CalculadoraAvanzada en un fichero java.
- b) Añade un atributo estático String llamado “ultimaSalida”.
- c) Crea un método estático que muestre por pantalla la tabla de multiplicar del uno al diez para un número que reciba como parámetro.
- d) Crea un método estático que calcule y retorne el factorial de un número que reciba como parámetro.
- e) Crea un método estático que retorne un booleano indicando si un número que recibe como parámetro es par.
- f) Haz que en el atributo estático “ultimaSalida” se almacene qué operación se hizo y con qué número se realizó. Por ejemplo:
 - “Se mostró la tabla de multiplicar para el número 6”
 - “Se calculó el factorial de 4, cuyo resultado es 2”
 - “Se comprobó si el número 10 era par. Es par”
- g) Crea una clase ejecutable llamada “ActividadTres” y en su método “main”, inicializa una variable del objeto “Scanner” para solicitar información al usuario.
- h) Realiza un bucle while, controlando la salida con la opción “-1”, para presentarle al usuario el “Menú de la calculadora avanzada” y solicitarle alguna de las siguientes opciones:
 1. **“Mostrar tabla de multiplicar para un número”**: Esta opción permite solicitar al usuario un número para llamar al método estático apropiado.
 2. **“Calcular factorial de un número”**: Esta opción solicitará al usuario un número para el que se calculará y mostrará por pantalla su factorial accediendo al método estático apropiado.
 3. **“Comprobar si un número es par”**: Esta opción solicitará al usuario un número para el que se mostrará por pantalla un mensaje “El número N es PAR/IMPAR” dependiendo del resultado.
 4. **“Ver última operación”**: Esta opción mostrará por pantalla la última operación realizada.
***Reto:** Busca la forma de guardar y mostrar las diez últimas operaciones y añade una opción adicional al menú que permita visualizarlas por pantalla.*
 5. **“Cerrar calculadora”**: Permite salir del programa.
***Reto:** Crea otro método estático que compruebe si un número es primo y añade esta opción al menú*