

## Ejercicio clase Intervalo

Crea una clase llamada Intervalo que contenga dos atributos (mínimo y máximo) de tipo *double*. Esta clase se va a usar con fines matemáticos por lo que está pensada para simular un intervalo matemático que tiene un comienzo (atributo *minimo*) y un final (atributo *maximo*). Evidentemente el primer valor ha de ser menor o igual que el segundo. Ejemplos: intervalo de calificaciones [0.0, 10.0], intervalo de aprobado [5.0, 10.0].

1. Crea varios constructores con esta clase:
  - a. Uno sin parámetros en el que se inicializarán a cero los atributos.
  - b. Otro que acepte dos valores de tipo *double*.
  - c. Otro con un solo parámetro de tipo *double* que será el que se le asigne al atributo máximo, el otro atributo valdrá 0.
  - d. Otro que acepte un solo parámetro de tipo Intervalo y genere un Intervalo de iguales valores que el recibido.
2. Crea los métodos:
  - a. Uno llamado “valido” que verifique que los atributos cumplen con su condición de relación entre ellos.
  - b. Método “longitud” que devuelve la longitud del intervalo.
  - c. Método “puntoMedio” que devuelve el punto en la mitad del intervalo.
  - d. Método “desplazar” que tiene un parámetro de tipo *double* y hace que los valores de ambos atributos se muevan en la cantidad que indique el parámetro. No devuelve nada.
  - e. Método “copia” que devuelve un intervalo como el que recibe el mensaje.
  - f. Método “iguales” que recibe otro intervalo como parámetro y comprueba si sus atributos son iguales que los del objeto que recibe el mensaje.
  - g. Método “incluye” que recibe un *double* como parámetro y comprueba si ese valor está contenido en el intervalo.

- h. Método “incluye” que recibe otro intervalo como parámetro y comprueba si ese valor está contenido en el intervalo que recibe el mensaje.
- i. Método “troceado” que recibe un entero como parámetro y devuelve un array de intervalos de la longitud resultante de partir el intervalo original en tantas partes como indique el parámetro pasado.
- j. Método “escalar” que tiene un parámetro de tipo *double* y hace que la longitud del intervalo se multiplique por el valor del parámetro, lógicamente los valores de ambos atributos se alterarán. No devuelve nada.
- k. Método “desplazado” que recibe un parámetro de tipo *double* y devuelve otro intervalo distinto con los valores de ambos atributos del intervalo original modificados en la cantidad que indique el parámetro con respecto de los atributos del objeto que recibe el mensaje.
- l. Método “interseccion” que recibe un intervalo por parámetro y que devuelve un intervalo con la intersección de ambos. Si no tuvieran ningún tramo en común devolvería *null*.
- m. Método “simetrico” que no recibe parámetros y que devuelve un intervalo simétrico al objeto que recibe el mensaje. (Ej. Si el intervalo es [5.0, 10.0] su simétrico es [0.0, 5.0]).