

Practica 1

Partimos de la clase llamada **Punto** desarrollada en ejercicios anteriores y que tiene las siguientes propiedades:

- Dos atributos `private` de tipo `double`.
- Un constructor con dos parámetros de tipo `double` que inicialice los dos atributos.
- Un constructor por defecto (sin parámetros) que inicialice los dos atributos al valor que se quiera.
- Un `getvalor` para cada uno de los atributos.
- Un método `calcularDistanciaDesde` que recibe un parámetro de tipo `Punto` y que devuelve un `double`.

1. Desarrollar una clase llamada `Circulo` que:

- Tenga dos atributos `private` de tipo `Punto` (`centro`) y `double`.(`radio`)
- Tenga un constructor con dos parámetros de tipo `Punto` y `double` que inicialice los dos atributos.
- Tenga un constructor por defecto (sin parámetros) que inicialice los dos atributos al valor que se quiera.
- Tenga un constructor con tres parámetros de tipo `double` que inicialice los dos atributos.
- Tenga un `getvalor` para cada uno de los atributos.
- Tenga un método `calcularDistanciaDesde` que recibe un parámetro de tipo `Punto` y que devuelve un `double`.

La distancia entre el punto y la circunferencia se calcula como la distancia entre el punto y el centro de la circunferencia más el radio

- Tenga un método `calcularArea` que no recibe ningún parámetro y devuelve un `double`.

$$\text{Área} = \pi \cdot r^2$$

- Tenga un método `calcularPerímetro` que no recibe ningún parámetro y devuelve un `double`.

$$\text{perímetro} = 2 \cdot \pi \cdot r$$

2. Desarrollar una clase llamada `Practica_1` que en su método `main`:

- Cree e inicialice dos objetos de la clase `Punto` y muestre la distancia entre ambos.
- Cree un objeto de la clase `Circulo` y muestre su área, perímetro y distancia a uno de los dos puntos creados al comienzo.

Practica 2

1. Desarrollar una clase llamada **Motor** que:
 - Tenga dos atributos `private`
 - de tipo `int` (litros de aceite)
 - de tipo `int` (caballos)
 - Tenga un constructor con un parámetro de tipo `int` para los CV. Los litros de aceite por defecto serán 0.
 - Tenga un `getvalor` para cada uno de los atributos.
 - Tenga un `setvalor` para los litros.
2. Desarrollar una clase llamada **Coche** que:
 - Tenga un atributo `private` de tipo `Motor`, un atributo de tipo `String` (marca), otro de tipo `String` (modelo) y otro de tipo `double` con el precio acumulado con las averías.
 - Tenga un constructor con dos parámetros de tipo `String` que inicialice la marca y el modelo.
 - Tenga un `getvalor` para cada uno de los atributos.
 - Un método `acumularAveria` que incrementará el importe gastado en averías.
3. Desarrollar una clase llamada **Garaje** que:
 - Tendrá tres atributos, un `Coche`, un `String` con la avería asociada y el número de coches que ha ido atendiendo.
 - El garaje solo podrá atender a un coche en cada momento. Controlar esta premisa.
 - Tenga un método `aceptarCoche` que recibe un parámetro de tipo `Coche` y la avería asociada. El garaje solo podrá atender a un coche en cada momento. Si ya está atendiendo uno, que devuelva un `false`.
 - Tenga un método `devolverCoche` que dejará al garaje en estado de aceptar un nuevo coche.
4. Desarrollar una clase llamada **Practica2** que en su método `main`:
 - Cree un garaje.
 - Cree 2 coches.
 - El garaje irá cogiendo los coches y devolviéndolos, acumulando un importe aleatorio (`Math.random()`) de la avería tratada.
 - Si la avería del coche es "aceite" incrementar en 10 los litros de aceite.
 - Los coches entrarán al menos 2 veces en el garaje.
 - Mostrar la información de los coches al final de `main`.