

1. Avión.

1) Haz una clase llamada Avion que tenga lo siguiente:

- Sus atributos son: modelo, peso, pasajeros, autonomía.
- Los métodos que se implementaran son:
 - o gets y sets de sus atributos
 - o String categoriaPeso(): devuelve la categoría a la que pertenece el avión según su peso:
 - de 1 a 5 ton = ligero
 - de 5 a 10 ton = mediano
 - de 10 a 20 ton = pesado
 - más de 20 ton = superpesado
 - o boolean comprobarPasajeros(int numPasajeros): devuelve true si numPasajeros caben en el avión y false si no caben.
 - o int numeroRepostajes (int km): dados los km que tiene que volar el avión devuelve cuantas veces tendrá que parar el avión a repostar (según su autonomía).

Ahora, crea una clase principal que haga lo siguiente:

- Pide por teclado los atributos para crear un objeto de tipo avión.
- Para dicho objeto, deberá mostrar su categoría de peso.
- Indicar para el objeto si puede llevar 200 pasajeros o no.
- Mostrar cuantas veces tiene que repostar el objeto para ir de París a Nueva York. (5837 km)

2) Modifica la clase Avion para añadirle lo siguiente

- Atributos: altura y velocidad
- Métodos:
 - o boolean enTierra() devuelve true si el avión está posado (altura=0) y falso si no.

o `double tiempo (int km)`: devuelve el tiempo que tardará el avión en recorrer los km indicados (según su velocidad actual).

Ahora, modifica la clase principal anterior para que haga lo siguiente:

- crea dos aviones, dales valores y di cual va más rápido de los dos.
- calcula cuánto tarda cada avión en ir de París a Nueva York.

2. Reloj.

Diseñar una clase llamada 'Reloj' con los atributos enteros: hora (24 horas), minuto y segundo, y los métodos siguientes:

- Un constructor que inicialice los atributos a valores pasados como parámetros.
- Un método llamado `incrementa` que incremente la hora en un segundo.

Por ejemplo:

17:55:45 será 17:55:46

17:53:59 será 17:54:00

17:59:59 será 18:00:00

Un método llamado `decrementa` que disminuya la hora en un segundo.

Por ejemplo:

17:52:25 será 17:52:24

17:53:00 será 17:52:59

17:00:00 será 16:59:59

Un método llamado `hora12` que convierte a String la hora mediante la concatenación de sus atributos y devuelve la cadena con el formato de 12 horas (hh:mm:ss AM/PM)

AM: (hasta las 12 mediodía), PM (desde las 12 hasta las 24 horas)

En una clase PrincipalReloj, pon a prueba los métodos anteriores:

- Crea un reloj al que aplicar repetidamente los métodos `incrementa`, `decrementa` y `hora12`, dando valores que revisen aquellos casos "límite".

3. Complejo.

Un número complejo está formado por una parte real y una parte imaginaria: $a+bi$, a es la parte real y b es la parte imaginaria.

Suma de dos números complejos:

$(2+3i) + (3+5i)=5+8i$, se suman las partes reales por un lado y las partes imaginarias por otro.

Crea un proyecto que contenga la clase Complejo:

Construye una clase Complejo con dos atributos:

- real: parte real del número complejo.
- imag: parte imaginaria del número complejo.

A continuación, crea los siguientes métodos dentro de la clase:

- public Complejo(): Constructor por defecto que inicializa los atributos a cero.
- public Complejo(double real, double imag): Constructor que inicializa los atributos a los valores indicados por los parámetros.
- public double consulta_Real(): Devuelve la parte real del objeto.
- public double consulta_Imag(): Devuelve la parte imaginaria del objeto.
- public void cambia_Real(double real): Asigna a la parte real del objeto el valor indicado en el parámetro real.
- public void cambia_Imag(double imag): Asigna a la parte imaginaria del objeto el valor indicado en el parámetro imag.
- public String toString(): Convierte a String el número complejo, mediante la concatenación de sus atributos y devuelve como resultado la cadena de texto $3 + 4i$, si 3 es la parte real y 4 la parte imaginaria.
- public void sumar(Complejo c + di): Aplicado a un número $a + bi$, suma la parte real con la parte real del número complejo $c + di$, y la parte imaginaria con la parte imaginaria del número complejo $c + di$.
- Se pueden crear más métodos que se consideren oportunos.

Crea una clase Principal en el mismo paquete que pruebe todos los anteriores métodos:

- Creando un numero complejo a partir del constructor por defecto.
- Pidiendo por teclado la parte real e imaginaria del número y asignándole tales valores.
- Creando un numero complejo a partir del constructor con parámetros
- Describiendo una salida que muestre la parte real y la parte imaginaria del número anterior (pasado a cadena).
- Por último, suma un número complejo a uno de los “objetos” ya creados utilizando el método sumar.