**CLASES , OBJETOS y VECTORES de objetos**

1. Repasar los ejemplos del proyecto PRIMEROSOBJETOSALUMNOS: comparar PersonaStaticpublic, PersonaStatic y Triangulo.
2. Crear y manejar un nuevo triangulo en la clase Pruebatriangulo
3. Crear la clase **Persona (no estatica)**  siguiendo el ejemplo de Triangulo. Los datos miembro privados serán nombre, edad y altura (se puede hacer una copia de PersonaStatic y modificar o añadir lo necesario)
4. Programar la clase de **Prueba** .

En el main :

Se crean dos personas diferentes .

se visualizan los datos de la más mayor en edad

se calcula la media de edad entre ambas

**Añadir a la clase Persona el dato DNI con sus métodos set y get . En el método setDni se debe comprobar que es un DNI correcto, 9 caracteres entre el 0 y el 9 y una letra acorde a los números. Si no es un valor correcto no se cambia el DNI. Para saber desde fuera del objeto si se ha podido realizar o no la operación, el método setDni puede devolver un booleano .**

**Probar lo anterior en el main de la clase de prueba.**

1. Modificar la clasede **Prueba** añadiendo los **métodos mayor** (devuelve cual es la mayor de dos personas) y **media** (devuelve la media de edad entre dos personas)
2. Añade a la clase **Persona** el **método mayor** como método miembro de la clase y pruébalo en la clase de prueba comparando con el método mayor del punto anterior
3. Programar la clase **LineaPuntos**

Datos miembro o atributos privados : **puntos** (cantidad de puntos de los que consta la línea)

Crear los métodos públicos **setpuntos, getpuntos ,subepuntos**(aumenta en 5 la cantidad de puntos) , **bajapuntos** (baja en 5 la cantidad de puntos) **dibujalinea** (visualiza una línea de tantos puntos como indica su atributo puntos)

Crear la clase de prueba donde se creen 3 líneas y se dibujen. Aumentar o disminuir y volver a dibujar

1. Programar la clase **Linea** o modificar la anterior para que tenga también tenga el atributo o dato miembro **tipo** (0 para puntos , 1 para asteriscos). Añadir los métodos **settipo** ( si el valor recibido no es 0 o 1 no debe hacerlo), **gettipo, cambiatipo** (alterna entre 0 y 1). Modificar **dibujalinea** para que dibuje la línea de puntos o asteriscos según el tipo.

Crear una línea y dibujarla en puntos y asteriscos

1. Mejorar la clase Linea para que no se le pueda dar valores negativos ni superiores a 80 al número de puntos en **setpuntos**. El método debe devolver si se ha realizado o no la operación.

Añadir el método **dibujalineaderecha** , como **dibujalinea** pero alineada en la parte derecha hasta 80 caracteres (dibujando blancos).

**Añadir los constructores sin parámetros y con parámetros a las clases anteriores.**

10.Crear en una clase de prueba **un vector de Lineas (o de LineaPuntos).**

Añadir líneas al vector

Mostrar el tipo y la cantidad de puntos o asteriscos de cada línea

Comprobar cual de todas ellas tiene más puntos o asteriscos

11**.** Modificar la clase anterior añadiendo los métodos **mostrar (**muestra el tipo y la cantidad de puntos o asteriscos de cada línea de un vector de líneas) y **comprobar (**devuelve cual de todas las líneas de un vector) tiene más puntos o asteriscos. Probar los nuevos métodos desde el main.

**12. Modificar la clase de prueba de persona sobrecargando los métodos mayor y media para que operen con un vector de personas**

**13. Hacer una copia de Persona en otra clase llamada BuenaPersona. Se añadirá un dato miembro estatico llamado fondo. Cada BuenaPersona que se cree tendrá la posibilidad de aportar dinero a ese fondo común (investigar sobre esto y resolverlo)**

**Como mínimo se debe tener completa las clases Persona y Prueba (todo lo que está en rojo) y la clase LineaPuntos**