

Actividades No 4 POO

AA1. Crea el diagrama de clases teniendo en cuenta los siguientes recuadros.

a.


<p>Pepito</p> 	<p>Características del burro:</p> <ul style="list-style-type: none">• Edad.• Color.• Peso.• Medidas. <p>Estas características son comunes a todos los burros.</p> <hr/> <p>Acciones que puede realizar el burro:</p> <ul style="list-style-type: none">• Comer.• Dormir.• Correr.• Rebuznar. <p>Estos comportamientos son comunes a todos los burros.</p>
--	---

Imagen 7.1. Burro como objeto. Pearson Scott Foresman (Autor). (2007). Donkey (PSF).png [Dibujo digitalizado]. Recuperado el 14 de marzo de 2014, de: [http://en.wikipedia.org/wiki/File:Donkey_\(PSF\).png](http://en.wikipedia.org/wiki/File:Donkey_(PSF).png)

b.

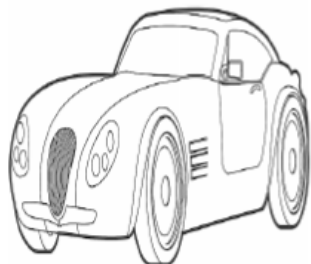
<p>Peugeot</p> 	<p>Características del carro:</p> <ul style="list-style-type: none">• Marca.• Modelo.• Chasis.• Cantidad de puertas.• Color <p>Estas características son comunes a todos los carros.</p> <hr/> <p>Acciones que puede realizar el carro:</p> <ul style="list-style-type: none">• Calcular velocidad máxima.• Calcular tiempo máximo de vida útil. <p>Estos comportamientos son comunes a todos los carros.</p>
---	--

Imagen 7.2. Carro como objeto. OpenClips (Responsable por subir imagen). (2013). Oldtimer Vintage Automobile Vintage Car Classic Car [Imagen digital]. Recuperado el 14 de marzo de 2014, de: <http://pixabay.com/en/oldtimer-vintage-automobile-151959>

c. Según el diagrama de la clase construye su código en java utilizando una interfaz de desarrollo.

AA2. Estudia el documento de 1_POO_Java.pdf y construye en java la siguiente clase:

Televisor
+ marca: String +color: String +tamaño: int +encendido: boolean +volumen: int +canal: int
+encenderTv(): void +apagarTv(): void +cambiarCanalMas(): void +cambiarCanalMenos(): void +subirVolumen(): void +bajarVolumen(): void

AA3. Estudia el documento 1_POO_Java.pdf y realiza el ejercicio 2 propuesto.

AA4. Haz una clase llamada Persona que siga las siguientes condiciones:

Sus atributos son: **nombre**, **edad**, **codigounico**, **sexo** (H hombre, M mujer), **peso** y **altura**. Tener en cuenta lo siguiente:

No queremos que se accedan directamente a ellos. Piensa que modificador de acceso es el más adecuado, igualmente su tipo. Si quieres añadir algún atributo puedes hacerlo.

Por defecto, todos los atributos menos el codigounico serán valores por defecto según su tipo (0 números, cadena null para String, etc.). Sexo será hombre por defecto, usa una constante para ello.

Se implantarán varios constructores:

Un constructor por defecto.

Un constructor con el nombre, edad y sexo, el resto por defecto.

Un constructor con todos los atributos como parámetro.

Los métodos que se implementaran son:

calcularIMC(): calculará si la persona está en su peso ideal (peso en kg/(altura² en m)), si esta fórmula devuelve un valor menor que 18.5, la función devuelve un -1 (Bajo de peso), si devuelve un número entre 18.5 y 25 (sin incluir 25), significa que su imc es ideal y la función devuelve un 0 y si devuelve un valor mayor o igual que 25 significa que tiene sobrepeso, la función devuelve un 1.

esMayorDeEdad(): indica si es mayor de edad, devuelve un booleano (false si es menor de edad o true si es mayor de edad).

comprobarSexo(char sexo): comprueba que el sexo ingresado es correcto (que sea 'H' o 'M'). Si no es correcto, el valor asignado al atributo será H. la asignación no será visible al exterior.

toString(): devuelve toda la información del objeto. Para la utilización de este método consulta como se utiliza el método toString y existente en java.

mostrarDatos(): muestra todos los datos de los atributos.

generaCodigoUnico(): genera un número aleatorio de 8 cifras de la siguiente manera:

- Al iniciar el código debe tener una letra mayúscula.
- En su segunda posición debe tener una letra en minúscula.
- Luego debe tener 6 dígitos numéricos.
- Por ultimo debe tener un carácter especial entre los siguientes, *, +, ¿, ¿, #, \$, %, &, !.

Este método será invocado cuando se construya el objeto. Puedes dividir el método para que te sea más fácil. No será visible al exterior.

Crea para este ejercicio métodos set y get de cada atributo, excepto de codigounico.

Ahora, crea una clase Principal que haga lo siguiente:

Pide por teclado el nombre, la edad, sexo, peso y altura.

Crea 3 objetos de la clase anterior, el primer objeto obtendrá las anteriores variables pedidas por teclado, el segundo objeto obtendrá todos los anteriores menos el peso y la altura y el último por defecto, para este último utiliza los métodos set para darle a los atributos un valor.

Para cada objeto, deberá comprobar si está en su peso ideal, tiene sobrepeso o por debajo de su peso ideal con un mensaje.

Indicar para cada objeto si es mayor de edad.

Por último, mostrar la información de cada objeto.

AA5. Haz una clase llamada Password que siga las siguientes condiciones:

Que tenga los atributos longitud y contraseña. Por defecto, la longitud será de 8.

Los constructores serán los siguiente:

Un constructor por defecto.

Un constructor con la longitud que nosotros le pasemos. Generará una contraseña aleatoria con esa longitud.

Los métodos que implementa serán:

esFuerte(): devuelve un booleano si es fuerte o no, para que sea fuerte debe tener más de 2 mayúsculas, más de 1 minúscula y más de 5 números.

generarPassword(): genera la contraseña del objeto con la longitud que tenga.

Método get para contraseña y longitud.

Método set para longitud.

Ahora, crea una clase clase ejecutable:

Crea un array de Passwords con el tamaño que tú le indiques por teclado.

Crea un bucle que cree un objeto para cada posición del array.

Indica también por teclado la longitud de los Passwords (antes de bucle).

Crea otro array de booleanos donde se almacene si el password del array de Password es o no fuerte (usa el bucle anterior).

Al final, muestra la contraseña y si es o no fuerte (usa el bucle anterior).

AA6. Crea una clase Fracción con métodos para sumar, restar, multiplicar y dividir fracciones.

AA7. Crea una clase Fecha. La clase tendrá además de constructores, métodos set y get y el método toString, un método para comprobar si la fecha es correcta y otro para modificar la fecha actual por la del día siguiente.

AA8. Crea una clase Persona. La clase contendrá un atributo fechaNacimiento que será un objeto de la clase Fecha creada en el ejercicio anterior.