



Tarea 5_1

CLASES

ESTUDIOS

Ciclo Formativo: **Desarrollo de aplicaciones Web**

Modalidad: **Distancia**

Módulo: **Programación**

ORGANIZACIÓN

Departamento: **Informática y Comunicaciones**

Instituto de Educación Secundaria “José Planes”

C/ Maestro Pérez Abadía, 2

30100 Espinardo – Murcia

T. 968 834 605

30010577@murciaeduca.es

www.iesjoseplanes.es

AUTORÍA

Isabel Puertas Toboso

Juan Pedro Martínez Gallar

LICENCIA



Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial 4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/)



Ejercicio 1



Se desea crear el proyecto: **Proyecto_StarWars**, que va a contener 1 paquete **paqueteStarWars1** y dentro de éste 3 clases: **Principal1** que contendrá el método `main()` y la clase **Personaje**.

CLASE **Personaje** (de carácter público)

ATRIBUTO	TIPO DE ATRIBUTO	ACCESO	PROPIETARIO	DESCRIPCION
nombre	String	privado	objeto	
salud	int	privado	objeto	Cuando salud sea ≤ 0 se considera muerto
fisico	int	privado	objeto	
fuerza	int	privado	objeto	
nivel	int	privado	objeto	Nivel de destreza
ph	int	privado	objeto	Puntos de habilidad

METODO	ACCESO	DESCRIPCION
Constructor	público	Con todos los atributos
getNombre	público	Devuelve el nombre del personaje actual
getSalud	público	Devuelve la salud del personaje actual
getFisico	público	Devuelve el físico del personaje actual
getFuerza	público	Devuelve la fuerza del personaje actual
getNivel	público	Devuelve el nivel de destreza del personaje actual
getPh	público	Devuelve los puntos de habilidad del personaje actual
setNombre	público	Cambia el nombre del personaje actual por otro considerado como dato de entrada
setSalud	público	Cambia la salud del personaje actual por otra considerada como dato de entrada
setFisico	público	Cambia el físico del personaje actual por otro considerado como dato de entrada
setFuerza	público	Cambia la fuerza del personaje actual considerada por otra como dato de entrada
setNivel	público	Cambia el nivel de destreza del personaje actual por otro considerado como dato de entrada
setPh	público	Cambia los puntos de habilidad del personaje actual por otros considerados como dato de entrada
toString	público	Devuelve una cadena que contiene TODOS los atributos del personaje
ataque_fisico	público	Devuelve un valor resultante del producto de un número al azar entre 0 y 2 (ambos incluidos) con el nivel y físico del personaje actual
ataque_fuerza	público	Devuelve un valor entero, sería 0 si los puntos de habilidad son 0 o negativos. Si son positivos, sería: decrementar los puntos de habilidad en 1 y devolver el producto del nivel con la fuerza
defensa	público	Se decrementa la salud en función de un número entero positivo considerado como dato de entrada
isVivo	público	Devuelve true si la salud es positiva, false si la salud es 0 o negativa del personaje actual

En la clase **Principal1**, método `main()` se pide:

- a) Crea los personajes: luke, darthVader y yoda.

PERSONAJE	nombre	salud	fisico	fuerza	nivel	ph
luke	Luke	500	300	100	2	1
darthVader	Darth Vader	500	200	500	3	4
yoda	Yoda	400	100	600	4	5

- b) Muestra los valores de los atributos de los 3 personajes.
 c) Cambia el físico de luke a 55. Después muestra el físico de luke.
 d) Daña la salud de luke con 10.
 e) Se produce un ataque físico de luke, con el valor generado daña la salud de darthVader. Intenta hacerlo en 1 sola línea ¿?
 f) Muestra la salud de darthVader
 g) Cambia la fuerza de darthVader a 600.
 h) Muestra los datos completos de los 3 personajes.
 i) Se produce un ataque de fuerza de darthVader dañando la salud de yoda.
 j) Muestra la salud de yoda.
 k) ¿yoda está vivo?
 l) Muestra los valores de los 3 personajes



Ejercicio 2



Dentro del proyecto anterior: **Proyecto_StarWars**, añade el paquete **paqueteTriangulo** y dentro de éste 2 clases: **Principal2** que contendrá el método `main()` y la clase **Triangulo**.

CLASE **Triangulo** (de carácter público)

ATRIBUTO	TIPO DE ATRIBUTO	ACCESO	PROPIETARIO
altura	int	privado	objeto
base	int	privado	objeto

MÉTODO	ACCESO	PROPIETARIO	DESCRIPCION
Constructor	público	objeto	Con todos los atributos
getBase	público	objeto	Devuelve la base del triángulo actual
getAltura	público	objeto	Devuelve la altura del triángulo actual
setBase	público	objeto	Cambia la base del triángulo actual por otra considerada como dato de entrada
setAltura	público	objeto	Cambia la altura del triángulo actual por otra considerada como dato de entrada
toString	público	objeto	Devuelve una cadena que contiene: altura, base e hipotenusa del triángulo actual
equals	público	objeto	Devuelve true o false si los atributos: base y altura del triángulo actual son iguales a la base y altura de otro Triángulo considerado como dato de entrada
area	público	objeto	Devuelve el área del triángulo actual
perimetro	público	objeto	Devuelve el perímetro del triángulo actual
hipotenusa	privado	objeto	Devuelve la hipotenusa del triángulo actual
duplicaTamaño	público	objeto	Método que amplía tanto la base como la altura del triángulo actual por 2
ampliaTamaño	público	objeto	Método que amplía tanto la base como la altura del triángulo actual por una cantidad entera considerada como dato de entrada

En la clase **Principal2**, en el método `main()` se pide:

- a) Crea los triángulos: `t1` y `t2`

TRIANGULO	base	altura
t1	12	10
t2	5	7

- Muestra los valores de los dos triángulos (base, altura e hipotenusa).
- Duplica el tamaño de `t1`, y muestra la base de `t1`.
- Amplía el tamaño de `t2` en 7 unidades, muestra la altura de `t2`.
- Muestra el valor del área, perímetro e hipotenusa de `t1`.
- Cambia la base de `t2` por el valor de la altura de `t1` multiplicado por 2.
- Muestra los valores de los atributos de los 2 triángulos.
- Comprueba si el triángulo `t1` es igual al triángulo `t2`.
- Crea un Personaje: `leia`, con los valores: "Leia", 500,100,50,1,1. ¿Lo puedes hacer? Explica por qué y qué es necesario para ello, si es que se puede.