

EXAMEN TEMAS 5 Y 6: ARRAYS Y POO

EJERCICIO 1 (2 PUNTOS)

Nos piden realizar una agenda telefónica de contactos.

Para ellos vamos a crear dos clases: La primera **clase** será **Contacto** y la crearemos en un fichero llamado **contacto.js**

Un contacto está definido por un **nombre** y un **teléfono** (No es necesario de validar).

Los métodos del contacto serán los siguientes:

- **Constructor**: que no recibe parámetros e inicializa ambos valores a ""
- **CrearContacto**: que recibe un nombre y un teléfono y los asigna correspondientemente.
- **MostrarTelefono**: que no recibe parámetros y **devuelve** el teléfono del contacto.
- **MostrarContacto**: que no recibe parámetros y **muestra** por pantalla los datos del contacto de la forma

Nombre: Pepe

teléfono: 666

EJERCICIO 2 (3 PUNTOS)

Habrás que crear otra **clase** llamada **Agenda (agenda.js)** que tendrá un atributo solamente llamado **contactos** que estará formado por un **array** de **contactos**.

Los métodos de la agenda serán los siguientes:

- **Constructor**: no recibe parámetros e inicializa array vacío.
- **añadirContacto**: Que recibe un parámetro una variable de la clase contacto y añade ese contacto a la agenda
- **listarContactos**: no recibe parámetros y en conjunción a **MostrarContacto** (de la **clase Contacto**), nos mostrará toda la agenda
- **buscaContacto**: que recibe como parámetro un nombre, buscará un contacto por ese nombre y mostrará su teléfono (usar **MostrarTelefono** de **Contacto**).

Crear un fichero **prueba.js**. donde haremos las siguientes operaciones:

- Crear **tres objetos** de tipo **Contacto**, **contacto1**, **contacto2** y **contacto3**.
- Usar **crearContacto** con los 3 objetos asignando valor a los mismos.
- Crear un **objeto** de tipo **Agenda**, **insertar contacto1 y contacto2** en la **agenda**
- **Listar** los **contactos** de la **Agenda**
- **Buscar contacto1** en la **agenda** (a ver si está)
- **Buscar contacto3** en la **agenda** (a ver si está)

EJERCICIO 3 (3 PUNTOS)

Como hace tiempo que no hacemos un juego vamos a realizar un simulador de partidas de ROL.

Cada una de las siguientes clases las haremos en ficheros independientes.

Por un lado, tendremos la **clase Item** con los siguientes **atributos**:

- **Nombre**: de tipo texto
- **Coste**: de tipo numérico
- **Tipo**: ataque o defensa
- **Potencia**: de tipo numérico

La clase **Item**, tendrá los siguientes **métodos**:

- **Constructor**, recibe como parámetros todos los valores de esta clase.
- **toString**, que no recibe parámetros y **devuelve** todos los valores del item.

Vamos a tener, también, la **clase Personaje**, esta clase tiene los siguientes **atributos**:

- **Nombre**: de tipo texto
- **Vida**: de tipo numero
- **Arma**: de tipo Item
- **Defensa**: de tipo Item
- **Dinero**: de tipo numero

Esta clase tendrá los siguientes **métodos**:

- **Constructor**: que recibe como parámetro el nombre, la vida y el dinero
- **AsignarArma**: que recibe como parámetro un **Item**(debe ser de ataque)
- **AsignarDefensa**: que recibe como parámetro un **Item**(debe ser de defensa)
- **Levear**, que no recibe parámetros, subirá la vida en 5 puntos, descontando 2 unidades de dinero, si no hay dinero suficiente no se subirá de nivel.
- **Farmear**, que recibe como parámetro un **Item**, si el **item** es mejor que el que tenemos nos equiparemos con dicho item, descontando el valor del item de nuestro dinero.
- **MostrarPersonaje**, que no recibe parámetros y mostrará por pantalla los valores del personaje y de los ítems que lo componen (mediante **toString**)

NOTA: Solo hay que hacer las clases, ya que en el ejercicio siguiente haremos la prueba de estas.

EJERCICIO 4 (2 PUNTOS)

Vamos a simular una partida utilizando las clases del ejercicio anterior.

Crear los siguientes objetos de tipo **Item**:

item1	<ul style="list-style-type: none">• Nombre: Espada Mágica• Coste:10• Tipo: ataque• Potencia:40
item2	<ul style="list-style-type: none">• Nombre: Hacha Élfica• Coste:20• Tipo: ataque• Potencia: 42
item3	<ul style="list-style-type: none">• Nombre: Escudo Negro• Coste:10• Tipo: defensa• Potencia:20
item 4	<ul style="list-style-type: none">• Nombre: Escudo Azul• Coste:15• Tipo: defensa• Potencia:25

Crear 2 objetos de tipo **Personaje**.

personaje1	<ul style="list-style-type: none">• Nombre: Vladimir el tuerto• Vida: 100• Dinero:200
personaje2	<ul style="list-style-type: none">• Nombre: Eldelbar Hans• Vida: 150• Dinero: 200

Utilizando los métodos **asignarArma** y **asignarDefensa**, asignar lo siguiente a los personajes:

personaje1	<ul style="list-style-type: none">• Arma: Hacha Élfica• Defensa: Escudo Azul
personaje2	<ul style="list-style-type: none">• Arma: Espada Mágica• Defensa: Escudo Negro

Utilizar **mostrarPersonaje** para ver los datos de uno y de otro personaje para prever quién ganará en la lucha (no implementar dicha lucha).