

JAVA: EJERCICIOS DE CLASE

Crea un proyecto nuevo llamado primeraPruebaNombrePrimerApellido, el mío sería primeraPruebaInmaculadaOlias. Pon tus datos no los míos.

Dentro de este proyecto tendrás que resolver los ejercicios y subirme un fichero comprimido del proyecto.

CONTESTADOR

El contestador del gimnasio Jacafit tiene memoria para almacenar hasta dos mensajes (en nuestro caso será una cadena de texto cada mensaje) que pueden dejar los usuarios cuando el teléfono no está disponible. Este contestador permite realizar las siguientes operaciones marcando el número adecuado en la estructura del menú:

1. Consultar número de mensajes
2. Consultar mensaje
 1. Mensaje número 1
 1. Oír
 2. Borrar
 2. Mensaje número 2
 1. Oír
 2. Borrar
- 3.- Añadir mensaje
4. Borrar todos los mensajes
5. Salir

Si al añadir mensaje el buzón esta lleno, nos debe aparecer el mensaje “Buzón completo” y no pedir nuevo mensaje. En caso contrario deberá añadir el mensaje que introduzca el usuario en el primer mensaje que encuentre vacío.

Si intenta oír un mensaje que no existe se deberá imprimir un mensaje de error.

En nuestro caso, cuando se seleccione la opción oír se deberá imprimir el mensaje.

Gestionar el menú en el programa principal y realizar funciones para cada opción del menú.

Comprimir descomprimir clave

Nuestra empresa nos dice que usan claves muy largas y que le hagamos una función para comprimir y otra para descomprimir la clave. Para ello nos dicen que la forma de comprimir la clave es ir mirando cada uno de los caracteres y generar una nueva clave comprimida que contendrá el carácter y el número de veces que se repite el carácter. Por ejemplo si tengo la clave “aaahhhhuuqaa” la clave comprimida será “a3h4u2q1a2” porque tengo 3 ‘a’, 4 ‘h’, 2 ‘u’, 1 ‘q’ y 2 ‘a’.

Para descomprimir tendré que hacer lo contrario, es decir, escribir el carácter el número de veces que se me pide.

Hemos tenido suerte y nos dicen que nuestras claves sólo contienen caracteres del alfabeto (es decir no tienen números) y que el número de veces que se repiten no es superior a 9.

Ejemplos:

Compresión:

kkkkkaaaaabbbbbessaaaaaa -> k5a5b5e1s2a6

hola -> h1o1l1a1

Descompresión:

a3z2q3 -> aaazzqqq

b1o2m1e1r1a1n1g1 -> boomerang

Criterios de evaluación:

0,20%	2.a) Se han identificado los fundamentos de la programación orientada a objetos.
0,20%	2.b) Se han escrito programas simples.
0,80%	2.c) Se han instanciado objetos a partir de clases predefinidas.
3,25%	2.d) Se han utilizado métodos y propiedades de los objetos.
3,25%	2.e) Se han escrito llamadas a métodos estáticos.
1,00%	2.f) Se han utilizado parámetros en la llamada a métodos.
0,40%	2.g) Se han incorporado y utilizado librerías de objetos.
0,10%	3.g) Se ha comentado y documentado el código.
0,05%	2.i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples
3,10%	3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
2,00%	3.d) Se ha escrito código utilizando control de excepciones.
0,20%	4.a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
0,20%	4.b) Se han definido clases.
3,50%	4.c) Se han definido propiedades y métodos.
2,00%	4.d) Se han creado constructores.
3,00%	4.e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
0,80%	4.f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
3,00%	4.h) Se han creado y utilizado métodos estáticos.