**EJERCICIOS JAVA REPASO. NAVIDAD**

**NOTA:** Las constantes deben declararse como tal.

**SIN USAR OBJETOS**

1. Hacer un programa que introduzca una serie de números por el teclado, hasta meter el 0, y cuente cuántos de ellos son el 2 y cuántos el 5.

Habrá un método que es el que me introduzca cada número, otro método controlará el final del programa y contará cuántos son 2 y cuántos 5, y luego otro método imprime resultados

1. Hacer un programa que obtenga el producto de dos números enteros positivos mediante sumas sucesivas.

Habrá un método que me permita meter cada número (validar que sea positivo), otro método obtendrá el producto, y otro imprimirá el resultado. En el main se invocarán a los diferentes métodos.

1. Calcular 11 + 22 +......+ nn, sin utilizar pow.

El valor de n, debe introducirse en un método, otro calculará la suma, y otro método imprimirá el resultado. En el main se invocarán los diferentes métodos.

1. Hacer un programa que lea 4 números por el teclado, por cada número comprobar si es par o impar. Si es par, hacer la suma entre 1 y el número, si es impar el producto.

Utilizar un método que meta cada número, otro método comprobará si el número es par o impar, dependiendo del resultado otro método calculará la suma o el producto y sacará resultados. El main controlará que son 4 números a meter e invocará a los métodos.

1. Hacer un programa que meta un número que tiene que ser par (si es impar, el programa debe decirnos que se trata de un error y nos pedirá otro número), y calcule las divisiones de los números comprendidos entre 1 y el número, y sacarlas por pantalla.

El número lo debe meter un método, otro método debe calcular las divisiones e ir sacando resultados. En el main se invocarán los métodos.

1. Hacer un programa que lea la temperatura de 5 provincias, y nos diga cuál fue la temperatura mínima.

Se calculará en el main la temperatura mínima, el resultado lo sacaremos en un método llamado imprimir.

1. Hacer un programa que nos permita introducir las toneladas de carbón que llevan 10 camiones; si el cargamento excede de 200 toneladas, el programa debe decirnos en cuánto excedió. Así mismo, debemos calcular cuánto dinero se pagó al final, si sabemos que cada tonelada cuesta 20 euros.

Las toneladas de cada camión se meterán en un método, en otro miraremos si las toneladas exceden de 200 y decirnos en cuánto excedió, al final un método imprimir sacará el resultado. En el main entre otras cosas, se controlarán que tenemos 10 camiones y se invocarán los distintos métodos.

1. Introducir por el teclado una serie de valores hasta que metamos el 6, y decir cuántos de ellos son 2, cuántos múltiplos de 3, y cuántos al multiplicarlos por 3 exceden de 9.

Un método me permite meter cada número, y otro método hará el resto de los cálculos y sacará resultados.

En el main se controlarán que metamos números hasta meter el 6, y se invocará a los métodos.

1. Introducir una cantidad por el teclado, si es menor de 500, sumarle el 50%, si es mayor o igual a 500, pero menor de 1000, sumarle el 7%, si es mayor o igual a 1000 y menor o igual a 5000, sumarle el 15%, y si es mayor de 5000, restarle el 5%.

Debemos hacer el programa de tal forma que el tiempo de ejecución sea el menor posible.

Usaremos un método para meter la cantidad y otro para hacer los cálculos.

1. Introducir una serie de nombres de personas con sus sueldos. Imprimir el nombre y el sueldo de la persona que más gana. El programa parará cuando metamos "fin", como nombre de persona. Al final debemos sacar también el número de personas metidas.

Divide el programa en los métodos que consideres necesarios.

1. Introducir la notas en número de los diferentes módulos que tienes , y escribirlos en forma de cadenas de caracteres. Si la nota es 1,2,3 escribir: "Muy deficiente"; si es 4, escribir: "Insuficiente"; si es 5, escribir: "suficiente"; 6 "bien"; 7, 8 "notable"; 9, "sobresaliente"; 10, "matrícula". Utilizar un switch.

Divide el programa en los métodos que consideres necesarios.

1. Hacer un programa que me permita meter un número por el teclado, positivo e impar (validar hasta que sea correcto), y realizar el producto empezando desde ese número hasta 1, disminuyéndolo de 3 en 3. (Ejm: si meto el 15, tengo que multiplicar el 15\*12\*9\*6\*3\*1).

**USANDO OBJETOS**

1. Simula mediante un programa, los pasos que damos para tomar una taza de café. Para ello, desarrollaremos una clase Cafetera con los atributos siguientes:

- **capacidadMaxima**: La capacidad máxima de café que puede contener la cafetera. Su valor inicial será de 2000 (c.c)

- **cantidadActual:** La cantidad de café que tiene la cafetera en un momento dado. Inicialmente su valor es 0 (cafetera vacía).

Al menos para resolver el ejercicio debemos implementar los siguientes métodos:

* **llenarCafetera():** LLenamos la cafetera de café.
* **servir\_Tazacafe():** Simula la acción de servir una taza con la capacidad indicada.
* **vaciarCafetera():** Tiramos el café sobrante.

Nos tomamos café 5 días a la semana (el sábado y domingo tomamos chocolate). Para cada día de la semana hay que imprimir el nombre.

1. Nuestra empresa tiene 3 trabajadores. Para cada trabajador nos interesa almacenar la siguiente información:

**- nombre.**

**- edad.**

**- categoría:** Pueden ser empleados, encargados o directivos.

**- antiguedad:** Pueden ser novatos, maduros o expertos.

Debemos calcular el sueldo de cada empleado de nuestra empresa, a partir de su antigüedad y categoría profesional. La forma de calcular el sueldo del empleado, se hará mediante un método al que llamaremos **calcularSueldo()** y de acuerdo a la siguiente tabla.

|  |  |
| --- | --- |
| **Sueldo base** | 607 € |
| **Empleado**  **Encargado**  **Directivo** | 15% sueldo base  35% sueldo base  60% sueldo base |
| **Novato**  **Maduro**  **Experto** | +150 €  +300 €  +600 € |

Los datos de los empleados debemos meterlos por el teclado, la categoría y antigüedad deben ser validados.

Además del método calcularSueldo(), debemos utilizar todos aquellos métodos que consideremos necesarios.

1. Diseñar una clase llamada Marciano con los siguientes atributos:

* **vivo** de tipo boolean , se debe inicializar a true;

- **nombre** de tipo String.

- **num\_marcianos:** permitirá contabilizar el número de marcianos vivos que hay en cada momento.

Además tendremos una constante llamada **SOY**, cuyo valor será “marciano”

**Los métodos que usaremos serán:**

* **constructor**
* **indicar\_nacimiento:** Visualizará el siguiente mensaje: “Hola, he nacido y soy el marciano xxxxxxx”.
  + - **indicar\_muerte:** Si el marciano está vivo nos debe permitir: cambiar el valor del atributo vivo, actualizar el valor del atributo número de marcianos y visualizar el mensaje “el marciano xxxxx ha muerto. En caso contrario debemos visualizar “el marciano ya está muerto”.
    - **comprobar\_vivo** : Nos indica si el marciano está vivo o no.
    - **contar\_marcianos**: Nos dice cuántos marcianos hay vivos.
    - Incluir las funciones getter y setter necesarias.

Una vez creada la clase e implementados los métodos, debemos diseñar la interface de usuario dentro de una clase llamada Marte, que permite crear y matar marcianos, de la siguiente forma:

* Se deben crear tres marcianos et1, et2, et3. Cada vez que se cree un nuevo marciano, deberá aparecer el mensaje: “Hola he nacido y soy el marciano (nombre del marciano”, y mostrar el número de marcianos vivos en ese momento.
* Mataremos al segundo marciano et2.
* Nacerá o crearemos un nuevo marciano et4.

Al final se deberá mostrar la situación en la que está cada uno de los marcianos (si está vivo o no).

1. En una academia imparten clase dos profesores. Debemoshacer un programa que imprima la nómina de cada trabajador, y el total pagado por la empresa durante un determinado mes que se debe introducir por el teclado.

En el entorno de trabajo que estamos tratando, consideramos los siguientes atributos para el profesor: nombre, tipo (primaria, secundaria) y horas\_ trabajadas.

Debemos tener en cuenta que el sueldo bruto de cada categoría es:

* secundaria=horas\_trabajadas\*20
* primaria=horas\_trabajadas\*15

La nómina tendrá el siguiente formato:

Nómina de: (visualizar el mes actual que debe introducirse por el teclado previamente)

Trabajador: (nombre del profesor) Categoría (primaria o secundaria)

Horas trabajadas: (nº horas)

Sueldo bruto: (calcularlo)

Descuento aplicado: (debe introducirse previamente por el teclado)

Sueldo neto (resultado de aplicar el descuento al sueldo bruto)

Para resolver el ejercicio, además del constructor, métodos getter y setter y otros que consideréis necesarios, usaremos los siguientes métodos:

* **calcular\_sueldoBruto**: calcula el sueldo bruto de un trabajador.
* **aplicar\_descuento**: nos dará el descuento a aplicar en cada caso contreto.
* **determinarMes:** nos dará el mes de emisión de la nómina.
* **retornar\_sueldoNeto**: calcula el sueldo neto, aplicando al sueldo bruto el descuento pertinente.
* main: implementa la interfaz de usuario.