

EJERCICIOS JAVA (HOJA 01)

[Identificadores, Scanner, Operadores, Constantes, Tipos, Variables, Casting]

Ej adicionales: A01 – A12

Consejo. Guardar los ejercicios siguiendo esta plantilla: 01-holaMundo, 03-IVA, etc.

Un ejercicio está concluido cuando se construye un ejemplo y se comprueba su funcionamiento

Nivel de dificultad: (sin asteriscos) Asequible  (***) Difícil

01	Escribir un programa que escriba en pantalla “Hola Mundo”
02	Indicar cuáles de los siguientes identificadores son correctos. En el caso de que alguno de ellos no lo sea, indica el porqué. <i>mi carta</i> <i>unacarta</i> <i>una2carta</i> <i>4cartas</i> <i>na_de_na</i> <i>__na_de_na</i> <i>¿carta?</i> <i>carta?</i> <i>tengoMuChO\$\$\$\$</i> <i>cuántosQuieres</i> <i>çhola</i> <i>ÃËËÖholaskov</i>
03	El IVA para ciertos artículos es del 21%. Realiza un programa que pida un precio y calcule a) El IVA a pagar b) Suponiendo que sea un precio sin IVA, calcular la suma total (precio + IVA) c) Suponiendo que sea un precio con IVA, calcular el precio sin IVA
04	Dado en centímetro el ancho y el alto de un rectángulo. ¿Cuál será el perímetro y el área?
05	Escribir un programa que calcule el número de segundos que existen en un día.
06	¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())? <pre>int a = 16.5; System.out.println(a);</pre>
07	¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())? <pre>long a = 500; int b = 3; b = a * b; System.out.println(b);</pre>
08	¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())? <pre>long a = 500; int b = 3; b *= a; System.out.println(b);</pre>
09	(*) ¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())? <pre>short a = 5, b =10; short c = a + b; System.out.println(c);</pre>
10	¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())? <pre>byte a = 64, b = 2; byte c = (byte) (a * b); System.out.println(c);</pre>

EJERCICIOS JAVA (HOJA 02)

[Clases, constructores, toString, static]

Ej adicionales: A13 – A18

11	<p>Crear la clase Coche</p> <ul style="list-style-type: none">a) Guardar como información el modelo, su color, si la pintura es metalizada, la matrícula, el tipo de coche (utilitario, familiar, deportivo), el año de fabricación y la modalidad del seguro (terceros o todo riesgo).b) Crear un método que permita imprimir sus datos
12	<p>Escribir un programa que represente las asignaturas de una carrera. Una asignatura tiene un nombre, un código numérico y el curso en el que se imparte. Los valores iniciales se muestran mediante un constructor.</p> <p>El programa debe imprimir los datos de la asignatura.</p>
13	<p>Diseñar</p> <ul style="list-style-type: none">a) La clase Punto (definida por dos coordenadas en el espacio)b) Un método que permita calcular la distancia entre dos puntosc) Un método que permita calcula la distancia de un punto al centro de coordenadas
14	<p>Realizar un programa que</p> <ul style="list-style-type: none">a) Permita generar empleados: nombre, apellido, dirección, DNI, mailb) Permita generar Departamentos: nombre, ubicación, teléfonoc) Permita guardar el departamento al cual se asigna el empleado y el director de cada departamento (que es también un empleado)
15	<p>Diseñar la clase Persona que permita guardar Nombre y DNI</p> <p>Implementar, además, un método para saber cuando dos personas son iguales</p> <p>¿Cómo puedo saber cuántas instancias de la clase Persona se han creado?</p>

EJERCICIOS JAVA (HOJA 03)

[Condiciones, bucles]

Ej adicionales: A19 – A26

16	Obtener un número aleatorio. Indicar si es mayor o igual a 5
17	Definir un método “mayorDeEdad” que, dada una edad, indique si la persona es mayor de edad (definir la mayoría de edad como una constante).
18	<p>(*) El club de golf “El agujerito” quiere un programa para las cuotas de los socios.</p> <ul style="list-style-type: none"> - Cuota inicial de 500 euros - Mayores de 65 años: descuento 50% - Mujeres: Descuento 15% - Menores de 18 años: descuento 25% (si el padre no es socio) o 35% (si el padre es socio) <p>Si un usuario puede acceder a más de un descuento, sólo cogerá el mayor descuento.</p> <p>NOTA: Definir como constantes los porcentajes que se pueden descontar.</p>
19	(*) Realiza un simulador de lotería primitiva que permita varias apuestas
20	<p>(**) Realizar un programa que valide una edad e indique un mensaje dependiendo de la edad. Ej.-</p> <ul style="list-style-type: none"> • (0..4] → pequeñín • (4..10] → niño • (10..40] → adolescente • (40 .. 80] → talludito
21	<p>Dado un número n, realizar un programa que sume los primeros n números pares.</p> <p>Ej.- para $n = 7 \rightarrow 2 + 4 + 6$</p>
22	Realizar las tablas de multiplicar de los 10 primeros números. Hacerlas con WHILE y FOR
23	<p>(*) Escribir un programa que pida un número de 1 a 10 y dibuje un triangulo con asteriscos</p> <ul style="list-style-type: none"> - Tantas filas como indique el número. - La base tendrá tantas estrellas como indique el número - Cada fila tendrá una estrella menos (se quitan de la parte izquierda) <div style="display: flex; align-items: center; justify-content: center;"> <pre> * * * * * * * * * * } Para n = 4 </pre> </div>
24	Dado el nombre de un mes, indicar cuántos días tiene. Utiliza un SWITCH.
25	<p>Escribir el abecedario utilizando un bucle</p> <ol style="list-style-type: none"> a) WHILE b) FOR c) DO... WHILE
26	(**) Crear un programa que dada una fecha calcule el tiempo (días, meses, años) pasado hasta hoy

EJERCICIOS JAVA (HOJA 04)

[Estructuras, Arrays, String, enumerados, generics]

Ej adicionales: A27 – A44

27	<p>Escribir un método tal que, dado un array de números, muestre sus elementos por pantalla separados por comas.</p> <p>Nota: Utiliza este método, a partir de ahora, para mostrar el array en pantalla</p>
28	<p>Escribir un programa que pida tres números enteros y los almacene en un array.</p>
29	<p>Escribir un método que dado un array de números enteros, devuelva la suma de los números.</p>
30	<p>Escribir un método que dado un array, devuelva otro array con los datos invertidos</p> <p>Ej.- [3, 4, 6] devolvería [6, 4, 3]</p>
31	<p>(**) Escribir un método que dado un array de cadenas, devuelva la cadena más larga.</p>
32	<p>Escribir un programa que dada una matriz de enteros por parámetro (hazla de 2 dimensiones) rellene cada celda (primero la 1ª fila, luego la 2ª....) con el 0, 1, 2, 3, 4 ... respectivamente</p>
33	<p>(*) Realiza un método que reciba un String y cambie “endeluego” por “desde luego”.</p> <p>Realiza dos versiones: una con String y otra con StringBuilder</p>
34	<p>Un túnel de lavado dispone de tres tipos de lavados: BÁSICO, NORMAL, SUPER. Los tiempos para cada tipo son 3', 5' y 7' respectivamente.</p> <p>Diseñar una estructura mediante enumerados que pueda utilizarse.</p>
35	<p>Una heladería vende copas de helado de tipo MINI (con 2 bolas), SUPER (con 3 bolas) y MEGA (con 5 bolas). Diseñar una estructura para trabajar.</p>
36	<p>(*) Diseñar una estructura para poder trabajar con los meses del año</p>
37	<p>(*) Diseñar un traductor de idiomas que pida una palabra y el idioma a traducir</p> <ul style="list-style-type: none">• Alemán (vocal). Quitar vocal y añadir sufijo “ujem”: silla → sillujem• Alemán (consonante). Quitar últimas letras y añadir sufijo “ujem”: camión → camiujem• Búlgaro (vocal). Añadir sufijo “kov”: silla → sillakov• Búlgaro (consonante). Quitar consonante y añadir sufijo “kov”: camión → camiokov• Catalán. Quitar la última letra
38	<p>Añadiendo elementos de distintos tipos, realizar un ejemplo de una estructura con</p> <ul style="list-style-type: none">• List Set Queue Map <p>(*) Realizar una segunda versión empleando generics</p>
39	<p>Diseñar una clase Taza que permita, de la forma más optimizada posible, trabajar con distintos materiales (cristal, porcelana, etc.)</p> <p>¿Cómo se puede indicar que algunos elementos de la vajilla podrían romperse?</p>

EJERCICIOS JAVA (HOJA 05)

[Excepciones]

Ej adicionales: A52 – A57

40	Realizar un programa que pida dos números entre 0 y 10 y los divida
41	<p>Realizar un ejercicio en el cual</p> <ul style="list-style-type: none">• Se pida un número entre 1 y 100• Si el número no pertenece a ese rango se captura la excepción y se tramita• Si el número es correcto se indica en número de intentos hasta conseguirlo
42	<p>Realizar un ejercicio en el cual</p> <ul style="list-style-type: none">• El ordenador calcula un número entre 1 y 100• Se pida un número entre 1 y 100• Si el número no es correcto se tramita la excepción y se le indica si es mayor o menor• Si el número es correcto se indica en número de intentos hasta conseguirlo
43	<p>Dados 2 arrays de cinco números, indicar en la salida el resultado de dividir el primer elemento del primer array entre el primero del segundo array (hasta completar los cinco números). Incluir un cero en el array2 y además acceder a 6 elementos en vez de a 5.</p>
44	<p>Indicar la salida que obtendrá el siguiente programa. ¿Y qué ocurre si a Devuelve enteros le suministro un 2?</p> <pre>public class EjemploExcepciones { public static int devuelveEntero(int num){ try{ if(num % 2 == 0){ throw new Exception("Lanzando excepción"); } return 1; } catch(Exception e){ return 2; } finally{ return 3; } } public static void main(String args[]){ System.out.println(devuelveEntero(1)); } }</pre>

45

(*) Indicar la salida que obtendrá el siguiente programa

```
public class EjemploExcepciones3 {  
  
    public static int devuelveEntero(int num) throws IOException{  
        try{  
            if(num%2==0){  
  
                throw new Exception("Lanzando excepción");  
            }  
            else{  
                throw new IOException("Desde try");  
            }  
        }  
        catch(Exception e){  
            return 2;  
        }  
        finally{  
            throw new IOException("Desde finally");  
        }  
    }  
  
    public static void main(String args[]){  
        int a = 0;  
        try{  
            a = devuelveEntero(1);  
        }catch(IOException ioE){  
            System.out.println(ioE.getMessage());  
        }  
        System.out.println(a);  
    }  
}
```

46

(**) Dispongo la clase atleta que tiene como atributos el nombre y la energía.

Cada vez que corre el objeto "Corredor" decrementa de 10 en 10 su energía hasta llegar al valor cero.

Crear un entrenamiento con las siguientes características

- Crea un objeto de la clase "Corredor". Ese objeto tiene energía 50.
- Se hace correr al corredor hasta que agote su energía 3 veces.
- La primera vez que se agote → pon la energía a 30
- La segunda vez que se agote → pon la energía a 10
- La tercera vez que se agote → tomar un aquarius

EJERCICIOS JAVA (HOJA 06)

[Threads]

Ej adicionales: A66 – A69

47	Realizar un programa que una vez que haya pasado un segundo escriba en pantalla un mensaje
48	Simular tres threads lanzados a la vez y que se duerman un tiempo aleatorio. Imprimir en pantalla un mensaje cada vez que se despierten.
49	Modificar el 48 para que el programa principal termine una vez que hayan acabado los hijos
50	Simular una tarea concreta que comienza en un momento concreto y acaba tras un tiempo aleatorio. Comenzar después la segunda tarea y mostrar un mensaje que indique cuanto tiempo ha transcurrido hasta que comenzara la segunda tarea. La aplicación debe dar los mensajes de cuando se crea el thread, comienza, finaliza, etc.
51	(*) Realiza un productor-consumidor en el cual se incluya un almacén sincronizado
52	Generar dos threads <ul style="list-style-type: none">• Uno de los thread tiene prioridad máxima, el otro tiene prioridad mínima• Lanzar los threads y, al acabar cada uno debe indicar su prioridad
53	(***) Realiza el ejercicio de la cena de los filósofos http://es.wikipedia.org/wiki/Problema_de_la_cena_de_los_filósofos

EJERCICIOS JAVA (HOJA 07)

[Entrada/Salida]

Ej adicionales: A58 – A65

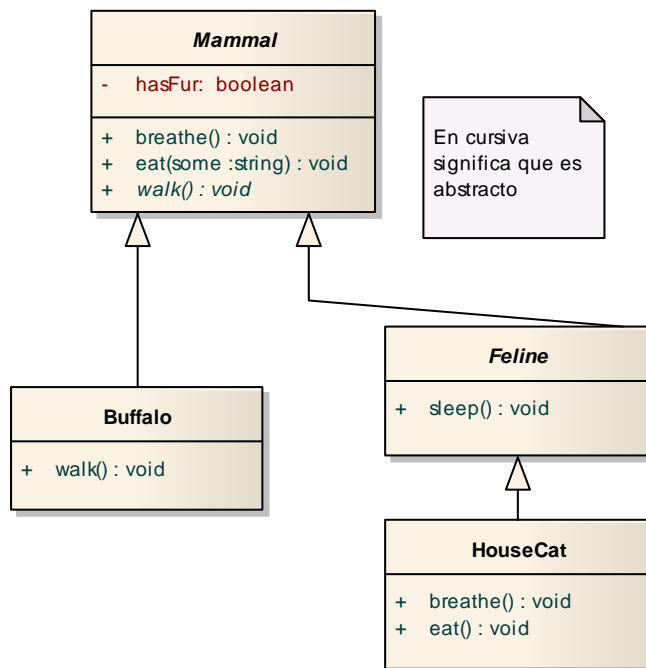
54	Realizar un programa que lea del teclado líneas y haga un “eco”. (**) Si fuera un número debe incrementarlo en una unidad y mostrarlo en pantalla
55	Escribir un programa que lea los datos de un fichero de texto y los escriba en pantalla
56	Diseñar un programa que emplee un fichero “prueba.txt”. Escriba los caracteres de la “a” a la “z”
57	Realizar un programa que pida 3 palabras, las escriba en un fichero de texto, las lea y las imprima desde el fichero. (*) Realiza otra versión pidiendo sólo números y guardándolo en 2 ficheros (según sea par o impar)
58	(*) Dado un fichero de texto con una lista de palabras desordenadas, escribirlas de forma ordenada en un segundo fichero. Controla que ambos ficheros existen
59	Dado un archivo de texto, indicar en pantalla el número de letras, el número de palabras y el número de líneas del texto incluido en el fichero
60	Realizar un programa al que se le pasen como argumento el fichero de entrada y el fichero de salida y realice en el segundo una copia del primero
61	Escribir un programa que pueda guardar una fecha en un fichero binario y que luego la lea para mostrarla en pantalla

EJERCICIOS JAVA (HOJA 08)

[Diseño aplicaciones, Interfaces, herencia, polimorfismo, generics]

Ej adicionales: A45 – A51





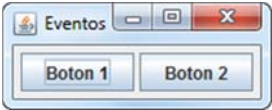
62	Escribir un programa que escriba en pantalla "Hola Mundo"
63	Diseñar la estructura Estudiante y Profesor de la forma más optimizada posible
64	Diseñar la clase Pixel (*) Diseñar la clase imagen BMP
65	(***) Realiza un sistema que permita círculo, rectángulo, dibujable, geometría, rectángulo grafico (con color), sol, folio, campo de futbol, tablero ajedrez
66	(**) Un ordenador dispone de un sistema operativo. Este sistema operativo disponer de un conjunto de driver. A un controlador SCSI se le puede asociar hasta 4 discos duros. Representar esta estructura usando como atributos sólo uno o dos por clase.
67	(*) Diseñar la clase Examen que guarde la asignatura, el aula, la Fecha y la Hora. Para guardar estos dos atributos, debe construirse las clases Fecha y Hora (usar java.time)
68	¿Existe algún problema en esta estructura?



EJERCICIOS JAVA (HOJA 09)

[Swing]

Ej adicionales: A70 – A77

69	<p>Construir un frame que Incluya un botón y un texto</p> <p>El texto debe mostrar el número de clics.</p> <p>Cada vez que se haga clic en el botón, el texto debe incrementarse en una unidad.</p>	
70	<p>Realizar este diseño</p>	
71	<p>Realizar este diseño</p>	
72	<p>Realizar un programa que muestre una interfaz con un botón y que al hacer clic lo indique en pantalla</p>	
73	<p>Realizar un programa que muestre dos botones. Al hacer clic en cualquiera de ellos debe indicarse en pantalla y al hacer rollover sobre el segundo también debe indicarlo</p>	

EJERCICIOS ADICIONALES JAVA

A01	Escribe un programa que pida un número de segundos y lo transforme a días, horas, minutos y sg.
A02	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = 2, b = 4; System.out.println(-a + 5 % b - a * a); System.out.println(5 + 3 % 7 * b * a - b % a); System.out.println((a + 1) * (b + 1) - b / a); System.out.println(!(a > b && 2 * a <= b)); System.out.println(b++ > 3 a + b <= 8 && !(a > b)); System.out.println(a++ < 6 && (b += 2) < a); </pre>
A03	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> byte a = 64, b = 2; byte c = (byte) (a * b); System.out.println(c); </pre>
A04	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> System.out.println(15 - 3 + "H" + 3 + 4); </pre>
A05	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = (3 < 4); System.out.println(a); </pre>
A06	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = 25, b = 4; double c = a/b; System.out.println(c); </pre>
A07	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = 25; float b = a/4.0f; System.out.println(b); </pre>
A08	<p>Las condiciones en un banco para un depósito son: interés anual del 2.75% TAE y abono de intereses semestral.</p> <p>Sabiendo que Hacienda somos todos y que retiene un 18% de las ganancias, escribir un programa que pida una cantidad numérica y muestre cuánto dinero dará el banco pasado el sexto mes.</p>
A09	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = 6; System.out.println(a++); System.out.println(++a); </pre>
A10	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre> int a = 6; int b = a-- * 3 / --a; System.out.println(a); System.out.println(b); </pre>

A11	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre>byte a = 121; int b = a; System.out.println(b);</pre>
A12	<p>¿Qué obtendremos en pantalla al ejecutar el siguiente programa (se ha obviado el main())?</p> <pre>int a = 6; System.out.println(a+++); System.out.println(++a);</pre>
A13	Escribir un programa que calcule cuánto vale el número PI
A14	<p>Diseñar</p> <ol style="list-style-type: none"> La clase Rectángulo Un método que pueda calcular el área de un rectángulo a partir de los datos proporcionados por teclado
A15	<p>Diseñar</p> <ol style="list-style-type: none"> La clase Triangulo Un método tal que dado un triangulo indique cuánto mide el lado más largo Un método tal que dado por teclado los atributos, indique cual es el lado más largo
A16	<p>Diseñar</p> <ol style="list-style-type: none"> La clase “Cuenta Corriente” que tiene un identificador, un cliente y permite definir la cantidad depositada. Un método para realizar ingresos. Un ingreso se define por el ordenante, la cantidad a ingresar y el número de cuenta. (**) La Cuenta Corriente dispondrá de unos intereses del 3% anual. Teniendo en cuenta que al abrir la cuenta se marca la fecha. Indicar los beneficios a día de hoy. Ej.- La cuenta se ha abierto el día 01/01/2012 con 100 euros. Hoy es 01/01/2013. El beneficio sería de 3 euros.
A17	<p>Diseñar</p> <ol style="list-style-type: none"> La clase “ElementoTablaPeriodica” que guarda el nombre, símbolo y número. El método “numeroDeElementos()” que indique cuantas instancias se han creado.
A18	<p>(**) Se quiere diseñar una clase para llevar las notas de un curso de Java. Crear las clases necesarias para poder guardar la información de alumnos, profesor, asignaturas y las notas del examen realizado por cada alumno en esa asignatura.</p> <p>NOTA: Si todavía no se han explicado estructuras de almacenamiento, puedes realizar el ejercicio con 3 alumnos, 2 profesores y 2 asignaturas.</p>
A19	Escribir un método que pida un número entero e indique si es par o impar
A20	Escribir un método que recibe 3 números como parámetros y los ordene en pantalla de menor a mayor.
A21	<p>Escribir un método para que, dada una puntuación escriba en pantalla la calificación</p> <ul style="list-style-type: none"> A si la puntuación es ≥ 90 B para puntuaciones entre 80 (incluido) y 90 C para puntuaciones entre 70 (incluido) y 80 D para puntuaciones entre 6 (incluido) y 70 F en cualquier otro caso

A22	(**) Escribir un programa que pida una cantidad de euros y los devuelva en monedas existentes (1 céntimo, 2 céntimos, 5 céntimos, etc.) Ej.- 5.35 euros → 2 monedas de 2 euros + 1 moneda de 1 euro + 1 moneda de 20 céntimos + 1 moneda de 10 céntimos + 1 moneda de 5 céntimos.
A23	(*) Escribir un método para que, dada una Fecha, indique si es correcta o falsa
A24	(**) Escribir un método que indique si un número dado es primo o no es primo.
A25	Dado un número n , realizar el factorial (Ej $5! \rightarrow 5 * 4 * 3 * 2 * 1$)
A26	(*)Escribir un programa que pida los límites de un rango (límite superior e inferior) y que a continuación vaya pidiendo números. Si el número se encuentra dentro de ese rango, escribirá el cubo de ese número.
A27	(*) Escribir un programa que vaya recibiendo números y que cuando se escriba un 0 se muestren los números escritos, la media aritmética, el mayor, el menor y el número que más veces aparece
A28	Escribir un método que rellene el array (atributo) con el número que se le pase como argumento.
A29	Diseñar un programa para saber si Math.random() es equitativo o no lo es
A30	Escribir un método que reciba un array y un número entero y devuelva otro array resultado de multiplicar cada elemento el array por ese número. Ej.- [2,4,5] y 3 devolverían [6, 12, 15]
A31	(***) Dada una matriz que representa un conjunto de puntos, diseñar un método que gire el conjunto de puntos 90º a la derecha.
A32	(*) Escribir un programa que reciba un array de cadenas y devuelva un array de enteros con los tamaños de cada una de las cadenas.
A33	(***) Dado un array con 1000 números aleatorios, crear dos métodos de ordenación y comprobar cuál de los dos es más rápido
A34	Realizar 100 tiradas de un dado e indicar la frecuencia de cada resultado
A35	Realizar un enumerado para llevar una lista de consolas de juegos (Wii, Play, etc.) y que permita escribir un mensaje personalizado dependiendo del tipo de consola elegida.
A36	Escribir un método que reciba un array de string y devuelva otro array ordenado alfabéticamente
A37	(**) Realizar un método para que, dado un texto, indique si ese texto es un palíndromo.
A38	Realiza un método que dada una cadena indique cuantas letras, dígitos y espacios en blanco tiene (la clase Character incluye isLetter, isDigit, isWhitespace).
A39	(*) Realiza una estructura que guarde notas de 8 alumnos que han realizado 5 exámenes/cu. Cada examen de cada alumno tiene 2 notas (teoría + práctica). La práctica vale el 60% de la nota final y la teoría el 40%. Realiza un método que reciba un array e indique los alumnos que han aprobado cada examen. Ej.- (Alumno 4)(Examen 3) tiene un 8 (práctica) y un 1 (teoría) → Nota final = 5,20
A40	Dada una matriz, realizar un método que la recorra por columnas
A41	Dada una matriz, realizar un método que comprueba si es simétrica (es decir la parte izquierda de la diagonal es igual a la parte derecha de la diagonal)

A42	Realizar un programa al cual se le proporcione una colección de palabras y las muestre primero por posición y después por orden lexicográfico
A43	(*) Escribir un método para que, dada una colección de palabras, indique cuantas letras tienen
A44	(*) Escribir un método “frecuencia” que indique cuantas veces aparece un elemento <code><E> int frecuencia(Collection<E> col, E elem)</code>
A45	(**) Diseñar las clases rectángulo, cuadrado, rombo y pentágono
A46	(***) Diseñar una estructura de Pila que incluya los métodos estaVacia(), extraer(), primero(), incluir()
A47	(**) Retocar el ejercicio anterior sabiendo que se quiere realizar una estructura de Pila y una de Cola. ¿Qué se debería emplear?
A48	Completar la clase Pila con un método imprimirYVaciar()
A49	(*) Realizar la estructura PilaPalabras que permita incluir palabras y disponga los métodos obtenerPalabraCima() y apilarPalabra()
A50	(*) ¿Varía mucho si se quiere implementar la Pila con un LinkedList o un ArrayList? ¿Se podría haber realizado de otra forma?
A51	(*) ¿Se puede realizar una Pila de tipo generic? ¿Tiene sentido?
A52	(*) Realiza un ejercicio que lea de un fichero y, mediante excepciones, cuente cuantos números, blancos, consonantes y vocales existen.
A53	Realizar un programa que sume los números enteros proporcionados como argumento de entrada
A54	Completar la clase “Punto” (Ej13) añadiendo una excepción para indicar si el punto es NULL
A55	Realizar un programa con un contador. Cuando llegue a cero debe emitir una alarma
A56	Indicar la salida que obtendrá el siguiente programa <pre> public class EjemploExcepciones2 { public static int devuelveEntero(int num) throws IOException { try { if (num % 2 == 0) { throw new Exception("Lanzando excepcion"); } else { throw new IOException("ioE"); } } catch (Exception e) { return 2; } finally { return 3; } } public static void main(String args[]) { int a = 0; try { a = devuelveEntero(1); } catch (IOException ioE) { } System.out.println(a); } } </pre>

A57	<p>Define un método main que cree un array de números que simulen edades, tales como {-2, 19, 69, 95, 1, 4, 2, 7}</p> <p>Con cada persona, mandar la edad a un método llamado "montañaRusa" y a otro método "castilloHinchable" de forma que:</p> <p>montañaRusa:</p> <ul style="list-style-type: none"> • Si la edad es menor de 0 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "EdadNegativaException" • Si la edad es mayor de 70 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "DemasiadoMayorException" • Si la edad es menor de 5 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "DemasiadoJovenException" • En otro caso dejar montar en la montaña rusa <p>castilloHinchable:</p> <ul style="list-style-type: none"> • Si la edad es menor de 0 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "EdadNegativaException" • Si la edad es mayor de 10 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "DemasiadoMayorException" • Si la edad es menor de 3 <ul style="list-style-type: none"> ○ Lanzar objeto excepción "DemasiadoJovenException" <p>En otro caso dejar montar en el castillo hinchable</p>
A58	Realizar un programa que lea datos de un fichero de texto e indique cuantos de los elementos leídos son números o palabras
A59	Crear una serie de contactos en una agenda y volcarlos a un fichero de salida
A60	Realizar un método al cual se le pasen como argumento dos ficheros binarios y realice una copia del primero en el segundo
A61	Realizar un método que sea capaz de escribir en pantalla el listado del directorio actual.
A62	Realizar un método al cual se le pase un fichero de texto y escriba en mayúsculas todas las "a"
A63	Realizar un método que indique el número de mayúsculas que existen en el fichero que se le ha proporcionado
A64	<p>(**) Realizar el método calendarioAMedida() que recibe el número de mes (1-12) y el día de la semana en que cae el 1 (L, M, X, J, V, S, D) y genera un archivo cuyo sufijo sea el número de mes y su contenido un listado con el formato DiaMesDiaSemana- DiaMesDiaSemana</p> <p>Ej.- 01M-02X-03J-04V-05S-06D-07L ...</p>
A65	(**) Realizar un programa que lea una serie de números de un fichero y los escriba en otro ficheros como números romanos.
A66	<p>(***) Realiza el ejercicio del peluquero Dormilón.</p> <p>http://es.wikipedia.org/wiki/Problema_del_barbero_durmiente</p>
A67	Demostrar cómo se pueden aplicar las prioridades cuando se trabaja con hilos

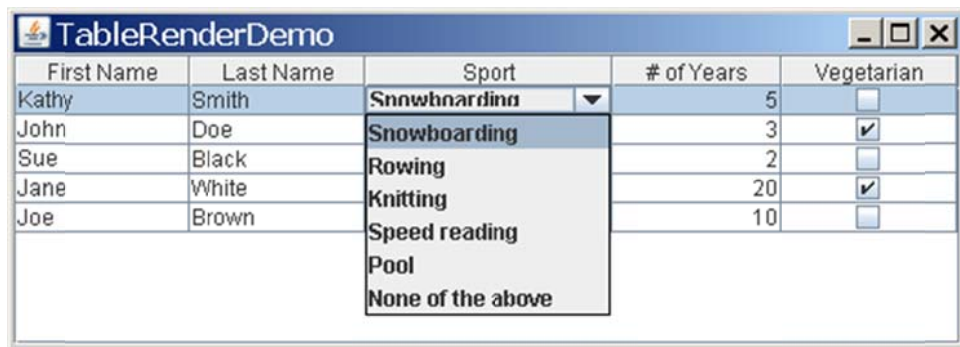
A68 Realizar un código en el cual dos threads confluyan para realizar un contador.
Este contador contara de 1 a 20000 pero ese contador será sincronizado
El contador que acceda escribirá de 2500 en 2500 números (es decir, sólo aparecerá el 0,2500, 5000,7500...). Cuando el bucle llegue a esa cifra mostrará el dato.

A69 (**) Adaptar el productor consumidor para que puedan almacenar 5 elementos.
Se vacía una vez que se encuentre completo

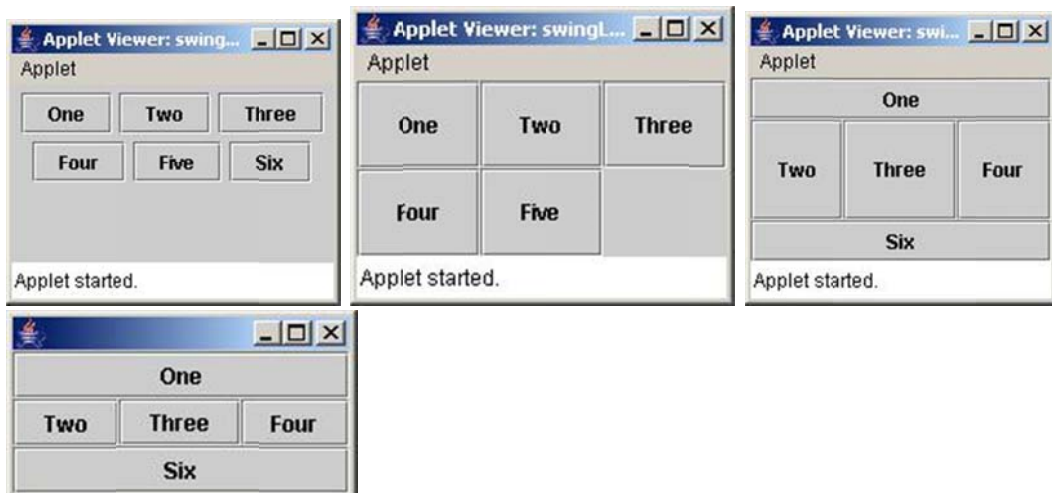
A70 Realizar este diseño



A71 Realizar este diseño



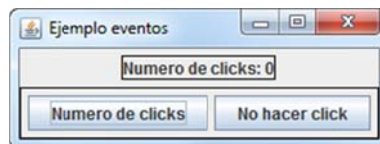
A72 Realizar estos diseños



A73 Realizar este diseño

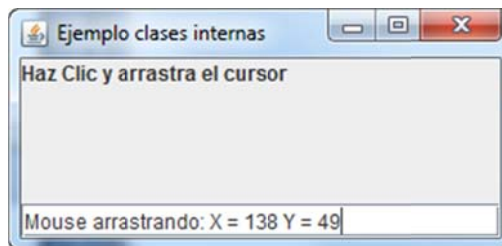


A74 Realizar un programa que al hacer clic en un botón se muestre en pantalla un contador que se incrementa y que con el otro botón decremente

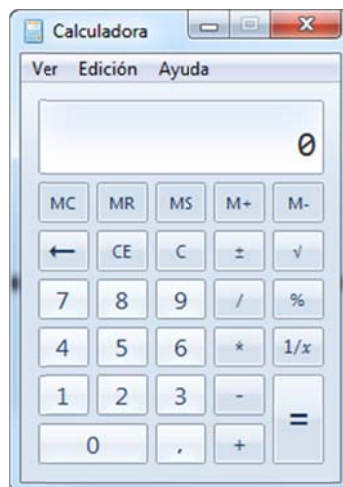


A75 Realizar el ejercicio anterior empleando adaptadores

A76 Realice un programa que muestre la posición del cursor



A77 (***) Realizar una calculadora con las funciones básicas.
Como muestra se proporciona la calculadora de Windows



EJERCICIOS JAVA (ESPECIALES)

E01 - EMPLEADOS

[Uso de Herencia, encapsulación]

En una aplicación creada para una empresa, se quieren guardar los datos de los empleados: nombre, salario, fecha de Nacimiento.

Los gerentes son empleados que además se hacen cargo de un departamento de la empresa. Su método "incentivo" multiplica el salario por 1.05

Cuando un gerente llega a director se le asigna un coche de empresa. La aplicación guarda la matrícula de ese coche. Su método "incentivo" multiplica el salario por 1.10 y le incrementa 100

Comprobar que las clases creadas funcionan y revisar si el método de incentivación funciona. Para verlo, crea un array de 5 empleados: 3 empleados normales, 1 gerente, 1 director.

E02 - POLIGON-PRET-A-PORTER

[Uso de Herencia, encapsulación, interfaces]

Una empresa está teniendo mucho éxito con la venta de polígonos.

Los polígonos los hacen de distintos materiales, principalmente cartón, plástico y madera.

Cada polígono tiene un precio, 100, 300 y 500 euros respectivamente... todo un chollo, oiga.

Los polígonos se encuentran formados de puntos. Este año se llevan mucho los polígonos coloreados. Se diferencian de los polígonos normales porque llevan color. Valen un 15% más.

Para la colección primavera-verano del 2013 se han sacado al mercado pentágonos coloreados con las firmas de los principales diseñadores... esto va a ser un pelotazo. Y sólo valen un 20% más. Cómprate uno ya!!!

E07 - AGENDA

[Uso de concurrencia, E/S, encapsulación, colecciones]

Diseñar una agenda de teléfonos (y su menú) que permita incluir contactos, eliminarlos, buscarlos, listarlos (lista completa y lista a partir de una letra), y modificarlos.

Además, debe permitir agrupar los teléfonos (por ejemplo "familia", amigos") y poder listar los teléfonos de esos grupos.

E03 - BIBLIOTECA

[Colecciones, herencia, interfaces, encapsulación]

En la biblioteca hay libros, comics y revistas. Interesa guardar siempre la fecha de publicación y el nombre. Además, de los libros se debe guardar la edición, el ISBN y el autor (que es único); de las revistas el número y la periodicidad y de los comics el número y la colección.

Sólo presto mis libros y debo saber si un libro está prestado o no. Aunque de momento no lo necesito puede que el futuro quiera prestar otros elementos de mi biblioteca. ¿Utilizo una interfaz o herencia? ¿Qué métodos necesito?

Escribir la clase Disco -hereda de Multimedia (titulo (String), formato(enumero), duración) teniendo en cuenta que los discos se pueden prestar.

Diseñar el sistema de biblioteca y crear un menú para las opciones

- a) Número de libros prestados ¿Se podría reutilizar el método para discos?
- b) Publicaciones anteriores a una fecha
- c) Imprimir el listado de publicaciones y discos
- d) Imprimir el listado de publicaciones y discos diferenciando por prestado o no

E04 - KENIA

[Clases abstractas, interfaces]

Se pide desarrollar un programa que permita gestionar todos los parques nacionales que existen en Kenia.

Cada parque posee una extensión (km²), un número de especies de animales y un nombre.

Un parque puede ser de uno de estos dos tipos: las Reservas de Caza o Áreas Protegidas.

- Los primeros, tienen un coste de licencia y un sólo tipo de arma a utilizar en él
- Los segundos, poseen una subvención del gobierno (en Chelines) y colabora en ellos una ONG.

Las Áreas Protegidas pueden ser acuáticas (incluyen número de lagos), terrestres (tipo de terreno) y el resto, que no se clasifican bajo ningún tipo específico.

Se pide:

- a) Crear 2 Reservas de Caza y 3 Áreas Protegidas (una de cada) e insertarlas en un array.
- b) Mostrar toda la información de los parques por consola recorriendo dicho array.
- c) Mostrar sólo el nombre de las áreas protegidas almacenadas.
- d) Algunos parques son visitables.
 - a. Crear un array de "zonas visitables".
 - b. Hacer que las Áreas Protegidas sean visitables.
 - c. Definir el método visitar(), que devolverá "Visitando área protegida XXXX".

E05 - SUPERMERCADO

[Colecciones, herencia, interfaces]

El Supermercado online “La compra Alegre” acaba de abrir sus puertas y quiere disponer de un sistema informático para su utilización.

Para que funcione se quiere diseñar un backoffice para dar de alta los productos. El sistema debe disponer de un menú con las siguientes opciones

- 1) Dar de alta un producto
- 2) Eliminar un producto
- 3) Listar los productos existentes
- 4) Dado un producto, indicar las cantidades existentes
- 5) Listado de los productos con existencias menores a 5 unidades

De un producto siempre se deben guardar cuatro valores: Código (debe ser único), Nombre del producto, Categoría (frutas, lácteos, etc.), Cantidad existente del producto

Crear el programa correspondiente

E06 - JUKEBOX

[Herencia, enumerado, interfaces]

Desarrollar un programa que permita trabajar con dos tipos de ficheros de música.

- Tipo 1: se modelará en base a los siguientes requerimientos:
 - Se pretende realizar un programa que gestione la música en un formato determinado que poseemos en nuestra casa. Toda la música que tenemos, está en el mismo formato.
 - Como datos de interés estarán el propio formato de audio, el nombre del fichero, la calidad expresada en kbits/seg (int), una valoración (0 mala y 5 muy buena) y una categoría (pop, rock, melódica, etc.).
- Tipo 2: posee la misma información que el anterior, con la diferencia que además incluye el video-clip de la canción.

Cuando se reproduzca un fichero de música simple, se escuchará la canción, mientras que si se reproduce un archivo de tipo 2 se escuchará y, además, se podrá visualizar el video.

El menú del programa incluirá las siguientes opciones

1. Listado De canciones
2. Canciones (solo música)
3. Canciones (sólo vídeos)
4. play (al azar)
5. Valoración media de mis canciones
6. play (sólo canciones de valoración 5)
7. play (solo rock)
8. play (de un listado)

E08 - PARKING

[Uso de Generics, ArrayList, Threads y E/S]

Diseñar una aplicación que modele la gestión de un parking.

En nuestro parking se permiten sólo tres tipos de vehículos (coches, motos y camiones). El sistema quiere almacenar de cada vehículo la matrícula, la marca, el modelo y color.

Adicionalmente, de cada uno de ellos, se quiere recoger estos datos:

- **COCHES:** numero de puertas
- **MOTOS:** si tiene sidecar
- **CAMIONES:** tara

Nada más arrancar la aplicación se solicita el **número de plazas** (puede aparecer como constante). A continuación se muestra al usuario un menú con las siguientes opciones (y siempre debe aparecer el número de plazas disponibles).

1. Añadir un nuevo vehículo
2. Eliminar un vehículo
3. Búsqueda de un vehículo (por matrícula)
4. Listado de vehículos aparcados
5. Listado de vehículos (por categorías de motos/coches/camiones)
6. Pagar parking (por matrícula)
7. Serializar parking
8. Salida de la aplicación
9. Leer datos de prueba (desde método)
0. Leer datos de prueba (desde fichero)

Simular el funcionamiento del parking (opciones 1, 2, 3, 4, 5, 8), utilizando Generics (ten en cuenta que pueden entrar a la vez tanto coches, camiones como motos).

Modif 1) Modificar las opciones 4 y 5 para preguntar si se quiere obtener el listado en un fichero. De ser así, generar un fichero de texto y guardar la información

Modif 2) Añadir una opción adicional en el menú (Opción 6 del “menú”) para obtener lo que debe pagar cada vehículo. Para calcular tiempo puedes emplear `System.currentTimeMillis()` ;

COCHE: 30 €/hora	MOTO: 20 €/hora	CAMIÓN: 50 €/hora
------------------	-----------------	-------------------

Modif 3) Añadir una opción adicional en el menú (Opción 0 del “menú”) para cargar datos desde un fichero de texto externo

Modif 4) Añadir una opción adicional en el menú (Opción 7 del “menú”) para serializar en un fichero binario el estado actual del parking.

Modif 5) Realiza un duplicado y crea una nueva versión que no tenga menú y en la cual se trabaje con thread (es decir, no haya interacción). Realiza estos casos

- a) Imagina que tienes varios carriles de acceso al parking, pero sólo una barrera de entrada (y una de salida). Modifica el código empleando threads.
- b) Añade varios carriles de salida pero una única barrera de salida.
- c) Añade 3 barreras (una para coches, para motos y otra para camiones)

NOTA Para usar dinero emplea dos decimales:

```
import java.text.DecimalFormat;

DecimalFormat formateador = new DecimalFormat("#.##");

System.out.println (formateador.format(3.43242383)); //3,43
```