

COLEXIO VIVAS S.L.	RAMA:	Informática	CICLO:	DAM				
	MÓDULO	Programación					CURSO:	1º
	PROTOCOLO:	Apuntes clases	AVAL:		DATA:			
	AUTOR		Francisco Bellas Aláez (Curro)					

Boletín 1: Ejercicios de entrenamiento

El objetivo de este boletín es empezar a entender la **estructura de un programa** y la creación básica del mismo con pequeños programas muy simples e incluso guiados en algunos caso.

También tienen como objetivo coger ritmo, disciplina y agilidad a la hora de realizar programas. Se busca que queden los conceptos de **compilación** y estructura de un programa así de los conceptos más básicos como **variables, literales y tipos de datos**. También son un primer acercamiento a la entrada y salida de datos. Como verás solo se usan **estructuras secuenciales**.


Aunque te parezca simple haz los ejercicios, ya que es posible que cometas algún error y es más fácil aprender sobre esos errores en casos simples como estos que no en otros ejercicios que harás más adelante.

Evita, en la medida de lo posible, buscar soluciones en internet. Es mejor comentar con un compañero o con el profesor las dudas que surjan.

Tema 1

1. Realiza un programa según indica los sucesivos apartados. Cada apartado es un paso a realizar y debes compilarlo y ejecutarlo para ver que no da problemas. Si los da, debes corregirlos antes de pasar al siguiente apartado.

- Crea un nuevo programa cuya clase se denomine Basico y contenga el main pero vacío, sin nada de código en su interior. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- Añade un mensaje de bienvenida que diga: "Bienvenidos a mi superprograma v1.0". Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- Añade otro mensaje en otra línea distinta que diga: "Ahora unas preguntas". Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- Añade otro mensaje en otra línea distinta que diga: "Dime tu nombre". Fíjate que sólo debe mostrar el mensaje, en este apartado aún no debe guardar el dato. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.

	RAMA:	Informática	CICLO:	DAM				
	MÓDULO	Programación					CURSO:	1º
	PROTOCOLO:	Apuntes clases	AVAL:		DATA:			
	AUTOR		Francisco Bellas Aláez (Curro)					

- e) A continuación debes declarar una variable tipo String (cadena), pues el nombre contiene caracteres. Luego úsala para guardar en ella el nombre introducido por el usuario mediante un `nextLine()`. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- f) Añade otro mensaje en otra línea distinta que diga: "Dime tu dirección". Fíjate que sólo debe mostrar el mensaje, en este apartado aún no debe guardar el dato. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- g) A continuación debes declarar una variable tipo cadena, pues la dirección contiene caracteres y números. Luego úsala para guardar en ella la dirección introducida por el usuario mediante un `nextLine()`. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- h) Añade otro mensaje en otra línea distinta que diga: "Dime tu edad". Fíjate que sólo debe mostrar el mensaje, en este apartado aún no debe guardar el dato. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- i) A continuación debes declarar una variable tipo int, pues la edad es un entero. Luego úsala para guardar en ella la edad introducida por el usuario mediante un `nextInt()`. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- j) Muestra la variable nombre. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- k) Muestra la variable dirección. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.
- l) Muestra la variable edad.
- m) Ahora intenta crear una frase mediante concatenación de cadenas que diga "Te llamas **nombre** vives en **dirección** y tienes **edad** años.". Por supuesto las palabras en negrita deben ser los datos. Consigue compilarlo y ejecutarlo observando y corrigiendo errores si se producen.

COLEXIO VIVAS S.L.	RAMA:	Informática	CICLO:	DAM	
	MÓDULO	Programación			CURSO: 1º
	PROTOCOLO:	Apuntes clases	AVAL:	DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)			

2. Siguiendo un esquema similar al ejercicio anterior trata de realizar un programa que pida primero un número al usuario, luego otro. Tras la petición de datos los sumará y los guardará en otra variable. Finalmente muestra dicha variable en la que está el resultado de la suma.

Como consejo, si no lo ves claro, trata de desmenuzar el problema en partes más pequeñas como en el ejercicio anterior. Por ejemplo:

- Muestra mensaje de bienvenida.
- Muestra mensaje de pedir primer número.
- Declarar variable y pedir primer número.
- Muestra mensaje de pedir segundo número.
- Declarar variable y pedir segundo número.
- Declarar variable, sumar los números y guardar el resultado.
- Mostrar variable resultado.


Te recomiendo ir compilando y probando en cada paso.

3. a) Haz un programa que pida al usuario dos números enteros. El programa realizará la suma de dichos números y mostrará el resultado por pantalla.

b) Una vez que te funcione lo anterior, en el mismo programa y a continuación debe calcular y mostrar también el cociente de la división y el resto de la división.

4. Escribe un programa que tenga una única instrucción `System.out.print` y con ella debes escribir en pantalla lo siguiente: Tu nombre en una línea, tus apellidos en la siguiente con una tabulación y tu edad entre comillas dobles (la misma que usas para Strings) en la tercera línea. Finalmente en la cuarta mostrará tu directorio de usuario, algo así como `C:\Users\minombre` (escríbelo tú, no hace falta que busques comandos que lo averigüen, ni que le preguntes nada al usuario, ni que uses variables).

5. Escribe un programa que pida al usuario su nombre, la edad (entero) y la temperatura actual (real). Debe mostrar luego por pantalla los tres datos, cada uno en una línea, pero usando un único comando `System.out.println` (debes concatenar los retornos de carro).

	RAMA:	Informática	CICLO:	DAM				
	MÓDULO	Programación					CURSO:	1º
	PROTOCOLO:	Apuntes clases	AVAL:		DATA:			
	AUTOR	Francisco Bellas Aláez (Curro)						

6. Realiza un programa en el que se pida al usuario dos números reales y muestre por pantalla el resultado de la suma, la resta, la multiplicación y la división de los mismos. Debe indicarse qué es cada resultado.

(Avanzado) Realiza además las siguientes operaciones:


- Añade las operaciones AND y OR a nivel de bit entre ambos números.
- Busca la forma de hacer (aún no lo vimos: estructura selectiva) que si se produce una división entre cero que informe del problema al usuario.
- Qué el programa indique también si los números introducidos son múltiplos entre sí.
- Debe decir cual es el mayor de los dos números.
- Finalmente haz que indique al usuario que introduzca 1 para repetir el programa u otro número para terminar. En caso de pulsar 1, lógicamente, el programa empezará de nuevo (Tampoco lo hemos visto aún: estructura repetitiva).

Tema 2

Selectivas

1. En un único programa pide al principio dos números enteros al usuario y luego ve haciendo mediante if las siguientes cosas (son independientes):

- Indica qué número es mayor.
- Indica si son o no iguales.
- Indica si el primero es positivo
- Di si el primero está entre 1 y 10 (Debes comprobar si es mayor que 1 **Y** menor que 10).
- Di si el segundo NO está entre 1 y 10 (Debes comprobar si es menor que 1 **O** mayor que 100).

	RAMA:	Informática	CICLO:	DAM				
	MÓDULO	Programación					CURSO:	1º
	PROTOCOLO:	Apuntes clases	AVAL:		DATA:			
	AUTOR		Francisco Bellas Aláez (Curro)					

2. Pide un número al usuario y, mediante un switch, escribe en pantalla el día de la semana a que corresponda (1: lunes, 2: martes, ... 7: domingo). Los case serán los números de día y en cada uno haces un print con el nombre del día. Si hubiera un número distinto muestra un mensaje de error (esto hazlo en el default).


Repetitivas

3. En el siguiente ejercicio vas realizar varios bucles con contador. Coge la estructura de bucle que te resulte más cómoda (while, do-while o for). Haz uno detrás de otro para que se vaya mostrando en la misma línea lo indicado (usa print). Recuerda, al terminar cada bucle hacer un println() para pasar a la línea siguiente. También recuerda inicializar el contador al principio de cada bucle, así puedes usar el mismo en todos y no declarar demasiadas variables:

- Muestra números del 1 al 10 de forma creciente.
- Muestra números del 10 al 1 de forma descendente.
- Muestra números pares entre 1 y 20 usando un contador de 2 en 2.
- Muestra números pares entre 1 y 20 usando un contador de 1 en 1 y dentro un if comprobando el resto de la división entre 2.
- Muestra números impares entre 1 y 20 (con el método que prefieras).
- Muestra números múltiplos de 5 del 10 al 50 con un contador que vaya de 5 en 5.
- Muestra del 100 al 0 de forma decreciente pero con contador de 10 en 10.

4. Haz el ejercicio 3 con bucle for. Si ya lo habías hecho con ese bucle hazlo con bucle while.

5. Mediante contador y acumulador suma los números del 1 al 20. Hazlo con los 3 bucles. En el primer bucle que hagas debe mostrarse dentro del bucle la evolución del acumulador y del contador. En los otros dos solo el resultado final (que debe ser el mismo en los 3 casos, claro). Recuerda que la estructura de contador será $c=c+1$ para ir de 1 en 1 y la del acumulador $acu=acu+c$ para sumar cada uno de los valores de c.

	RAMA:	Informática	CICLO:	DAM			
	MÓDULO	Programación				CURSO:	1º
	PROTOCOLO:	Apuntes clases	AVAL:		DATA:		
	AUTOR	Francisco Bellas Aláez (Curro)					

6. Haz un bucle do-while que pida números enteros al usuario hasta que este meta uno positivo. Una vez que te funcione, añade dentro del bucle un if que saque un mensaje de error si el número es negativo. Ya fuera del bucle muestra el número con printf ocupando 5 espacios.

7. Haz un bucle do-while que pida números double al usuario pero si dicho número no está entre 1 y 10 lo vuelve a pedir y además informa del error (como el if del ejercicio previo). La condición es similar al último apartado del ejercicio 6. Ya fuera del bucle muestra el número con printf con 3 decimales.

8. Haz un bucle que pida números al usuario mientras estos sean positivos y cada uno que meta lo va sumando en un acumulador. Ojo porque ahora no acumulas el contador (es más, no hay contador) si no el número que mete el usuario. Acaba cuando el usuario mete uno negativo. Ya fuera del bucle muestra el resultado del acumulador (suma de los números del usuario).