

1. (Círculo) Escribir un programa que calcule la longitud de la circunferencia y el área del círculo para un valor del radio introducido por el usuario.
 - La longitud de la circunferencia es $2 \cdot \text{PI} \cdot \text{radio}$
 - El área de un círculo es $\text{PI} \cdot \text{radio}^2$
2. (Pitágoras) Escribir un programa que calcule la hipotenusa de un triángulo rectángulo conocido el valor de sus dos catetos. El usuario introducirá el valor de los catetos.
3. (Dados) Escribir un programa que simula el lanzamiento de dos dados.

Ejemplo de ejecución:

*Dado 1 : 5
Dado 2: 4
Puntuación total: 9*

4. (Redondear) `Math.round(x)` redondea x de manera que éste se queda sin valor en su parte decimal. (`Math.round(35.5289)` da como resultado 35.0)

Escribe un programa en el que el usuario introduzca un número real y a continuación se muestre redondeado a un decimal. Pista : combinar producto, división y `Math.round()`

Ejemplos de ejecución:

*Introduce un número entero:
35.5289 ↵
El número, redondeado a un decimal es 35.5*

*Introduce un número entero:
35.5589 ↵
El número, redondeado a un decimal es 35.6*

5. (Nombre) Escribir un programa en el que el usuario pueda escribir su nombre. El programa le dirá si la primera y la última letra del nombre coinciden o no.
6. (Email) El usuario introduce su dirección de correo electrónico, del tipo `nombre@empresa.com`. Mostrar, por separado el nombre de usuario, el subdominio y el dominio:

Ejemplos de ejecución:

Introduce tu nombre dirección de email: paco@iesmatisse.es

Nombre de usuario: paco
Subdominio: iesmatisse
Dominio: es

7. (Palabras) Escribir un programa en el que se solicite al usuario y lea de teclado un texto, que puede estar compuesto de varias palabras. Una vez introducido el texto, el programa mostrará al usuario:

- La primera letra del texto introducido.
- La última letra del texto introducido.
- La primera palabra que contenga el texto
- La segunda palabra que contenga el texto (si contiene dos o más palabras)
- La tercera palabra que contenga el texto (si contiene tres o más palabras)

8. (Partir) Escribir un programa para partir un texto en dos partes por la posición que indique el usuario:

- En primer lugar se pedirá al usuario un texto
- En segundo lugar se le pedirá una posición.
- Si en la posición indicada hay un espacio en blanco (‘ ‘), se mostrará el texto partido: hasta la posición indicada en una línea y a partir de la posición indicada en otra.
- Si la posición es incorrecta o si no contiene un espacio en blanco se indicará al usuario que el texto no se puede partir.

Ejemplo de ejecución:

Introduce un texto:
La casa del final de la calle es grande
Introduce una posición: 17

Primera parte: La casa del final
Segunda parte: de la calle es grande

Ejemplo de ejecución:

Introduce un texto:
La casa del final de la calle es grande
Introduce una posición: 15

No se puede partir

Ejemplo de ejecución:

Introduce un texto:
La casa del final de la calle es grande
Introduce una posición: 150

No se puede partir

9. (DesplazarIzquierda) El usuario introduce una palabra, que leemos con next(). A continuación mostramos la palabra introducida pero con el primer carácter en la última posición, tal y como muestra el siguiente ejemplo:

Ejemplo:

Introduce una palabra: Abstracción
Texto desplazado: bstracciónA

10. (QuitarBlancosSobrantes) El usuario introduce un texto de dos palabras que leemos con nextLine(). Las dos palabras pueden estar separadas por varios espacios en blanco. Mostrarlas por pantalla con un solo espacio en medio

Ejemplo de ejecución:

Introduce dos palabras separadas por varios espacios:
casa grande
Quitando espacios sobrantes: casa grande

11. (LongitudEntero) El usuario introduce un entero largo leído con nextLong() y el programa muestra la longitud del entero introducido.

Para ello convertiremos el número entero en un String y a continuación averiguaremos la longitud del String con el método length. Para convertir el número entero en un String puedes utilizar el método estático valueOf() de la clase String, que transforma distintos tipos de datos a String.

Ejemplo de ejecución:

Introduce un entero largo: 654452131843165
Longitud: 15

12. (Calculadora) Escribir un programa que simule una calculadora básica. El programa solicitará al usuario dos números y la operación que quiere hacer y le mostrará el resultado. Solo hay cuatro operaciones posibles: + - * /

Ejemplo de ejecución:

Primer número: 10,5
Segundo número: 12,8

Operación: +

Resultado de $10,5 + 12,8$ es 23,3

13. (Taxi) Se desea calcular el coste de un trayecto en taxi en función de los kilómetros recorridos, el día de la semana y la hora en que se hace el viaje. Las tarifas son las siguientes:

- Días laborables, por el día: 0.73 €/km.
- Días laborables por la noche: 0.84 €/km.
- Domingos (en cualquier horario): 0.93 €/km.

Además,

- Por el día, como mínimo se cobrará 2.95€
- Por la noche, como mínimo se cobrará 4€.

Escribir un programa que solicite al usuario:

- Momento del trayecto: DIA o NOCHE.
- El día de la semana: El usuario introduce un valor entre 1 y 7, siendo 1 el lunes y 7 el domingo
- Los kilómetros recorridos.

y muestre el coste del trayecto

14. (*ImparesHastaN*) Dado un nº entero n que introduce el usuario, mostrar por pantalla los números impares que hay entre 1 y n . Por ejemplo, si n es 8 mostrará 1 3 5 7

15. (*NImpares*) Dado un nº entero n que introduce el usuario,, mostrar los n primeros números impares. Por ejemplo, si n es 3 mostrará 1 3 5 (3 primeros impares)

16. (*CuentaAtras*) Dado un nº entero n que introduce el usuario, mostrar una cuenta atrás partiendo de n : $n, n-1, \dots, 5, 4, 3, 2, 1, 0$

17. (*SumaNPrimeros*) Dado un nº entero n que introduce el usuario,, mostrar la suma de los números entre 1 y n .

18. (*MostrarDivisoresN*) Dado un n° entero n que introduce el usuario, mostrar todos sus divisores, incluidos el 1 y el mismo n . Por ejemplo, si n es 12 mostraría 1, 2, 3, 4, 6 y 12
19. (*SumaDivisoresN*) Dado un entero n recibido como parámetro, mostrar la suma de todos sus divisores, incluidos el 1 y el mismo n .
20. (*Etapas*) El ser humano pasa por una serie de etapas en su vida que, con carácter general se asocian a las edades que aparecen en la tabla siguiente.

Infancia	Hasta los 10 años
Pubertad	De 11 a 14 años
Adolescencia	De 15 a 21 años
Adulthood	De 22 a 55 años
Vejez	De 56 a 70 años
Ancianidad	A partir de los 71 años

Escribe un programa en el que el usuario introduzca las edades de una serie de personas y calcule y muestre que **porcentaje** de personas que se encuentran en cada etapa. En primer lugar el programa pedirá el número de personas que participan en la muestra y a continuación solicitará la edad de cada una de ellas. El resultado será similar al siguiente (intenta que los datos aparezcan bien tabulados):

Infancia:	5.30 %
Pubertad:	10.70 %
Adolescencia:	21.20 %
...	
...	

21. (*SumaSerie*) Dado un número n , introducido por el usuario, calcula y muestra por pantalla el resultado de la siguiente suma $1/1 + 1/2 + 1/3 + \dots + 1/n$
22. (*NumeroDeCifras*) Escribir un programa en el que el usuario introduzca un número entero y se le diga cuantas cifras tiene. Pistas: ¿Cuántas cifras tiene el n° 25688? ¿Cuántas veces podemos dividir el n° 25688 por 10 hasta que se hace cero? Cuidado, el n° 0 tiene una cifra.
23. (*Círculo*) Escribir un programa que calcule la longitud de la circunferencia y el área del círculo y el volumen de una esfera para un valor del radio introducido por el usuario. El programa utilizará tres métodos (además del método main):
- `public static double longitudCircunferencia (double radio)`
 - `public static double areaCirculo (double radio)`
 - `public static double volumenEsfera (double radio)`

24. (ReplicarCaracteres) Escribir el método

public static String replicar (char letra, int veces), que devuelva un String compuesto por tantas repeticiones del carácter *letra* como indique el parámetro *veces*. Por ejemplo *replicar('a',5)* devolverá "aaaaa".

Probar el método desde el método main

25. (LetraNif) Escribir un método

public static char calcularLetraNif(int numero): devuelve la letra que corresponde al número de nif que se recibe como parámetro.

Escribir un programa en el que el usuario escriba un nif (que incluye el número y la letra) y se diga al usuario si el nif introducido es correcto o no. Se hará uso del método *calcularLetraNif* hecho anteriormente.

26. (Mes) Escribir los siguientes métodos y probarlos desde el método main:

public static int diasDe(int mes): devuelve el número de días que tiene el mes que se indica como parámetro.

public static int diasDe(String mes): devuelve el número de días que tiene el mes que se indica como parámetro.

27. (String) Escribir los siguientes métodos y probarlos desde el método main:

public static int veces(String texto, char letra): devuelve el número de veces que el carácter *letra* aparece en el String *texto*

public static void repetir(String texto, int veces): imprime en pantalla el texto indicado, tantas veces como indique el parámetro *veces*.