



Programación

Act04_Funciones

Francisco José García Cutillas | 1FPGS_DAM



Índice

Ejercicio A01	3
Ejercicio 02	6
Ejercicio 03	8
Ejercicio 04	9
Ejercicio 05	11
Ejercicio 06	13
Ejercicio 07	16
Ejercicio 08	21
Ejercicio 09	25
Ejercicio 10	27
Ejercicio 11	29
Ejercicio 12	31
Ejercicio 13	33
Ejercicio 14	36
Ejercicio 15	38

Ejercicio A01

Implementa y utiliza un método que reciba tres números enteros y devuelva el mayor de ellos.

```
package com.mycompany.garciacutillasfranciscojoseut04funciones;

import java.util.Scanner;

/**
 *
 * @author fran
 */
public class GarciaCutillasFranciscoJoseUT04Funciones {

    public static void main(String[] args) {

        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"
        ejerciciosUt04 ej = new ejerciciosUt04();
        ej.ejercicio01();

    }

}
```

```

package com.mycompany.garciacutillasfranciscojoseut04funciones;

import java.util.Scanner;

public class ejerciciosUt04 {

    public void ejercicio01() {

        System.out.println(x: "****RESOLUCIÓN EJERCICIO 1****");
        System.out.println(x: "");

        //Creación de la variable Scanner para introducir valores por consola
        Scanner sc = new Scanner(System.in);

        System.out.println(x: "Introduce un número entero:");
        int num1 = sc.nextInt();
        System.out.println(x: "Introduce un número entero:");
        int num2 = sc.nextInt();
        System.out.println(x: "Introduce un número entero:");
        int num3 = sc.nextInt();

        //Creación del método que vamos a utilizar en el ejercicio
        funcionesTarea04 fun04 = new funcionesTarea04();

        //Cálculo del mayor usando la función devuelveMayor
        int mayor = fun04.devuelveMayor(num1,num2,num3);

        //Salida por pantalla del mayor
        System.out.println("El número mayor introducido es: "+mayor);
    }
}

```

```

package com.mycompany.garciacutillasfranciscojoseut04funciones;

public class funcionesTarea04 {

    public int devuelveMayor(int num1, int num2, int num3) {

        int mayor = 0;

        if (num1 >= num2 && num1 >= num3) {
            mayor = num1;
        } else if (num1 <= num2 && num2 <= num3) {
            mayor = num3;
        } else {
            mayor = num2;
        }

        return mayor;
    }
}

```

```
****RESOLUCIÓN EJERCICIO 1****
```

```
Introduce un número entero:
```

```
2
```

```
Introduce un número entero:
```

```
3
```

```
Introduce un número entero:
```

```
5
```

```
El número mayor introducido es: 5
```

```
-----  
BUILD SUCCESS  
-----
```

```
Total time: 7.941 s
```

```
Finished at: 2022-11-16T13:41:26+01:00  
-----
```

```
****RESOLUCIÓN EJERCICIO 1****
```

```
Introduce un número entero:
```

```
-5
```

```
Introduce un número entero:
```

```
0
```

```
Introduce un número entero:
```

```
-5
```

```
El número mayor introducido es: 0
```

```
-----  
BUILD SUCCESS  
-----
```

```
Total time: 11.035 s
```

```
Finished at: 2022-11-16T13:45:24+01:00
```

Ejercicio 02

Implementa y utiliza un método que reciba dos enteros y devuelva un boolean, indicando true si el segundo parámetro es múltiplo del primero y false en caso contrario.

```
public class GarciaCutillasFranciscoJoseUT04Funciones {

    public static void main(String[] args) {

        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"
        /*
        ejerciciosUt04 ej01 = new ejerciciosUt04();
        ej01.ejercicio01();
        */
        ejerciciosUt04 ej02 = new ejerciciosUt04();
        ej02.ejercicio02();

    }

}
```

```
public void ejercicio02() {

    System.out.println(:"****RESOLUCIÓN EJERCICIO 2****");
    System.out.println(:"");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(:"source: System.in");

    System.out.println(:"Introduce un número entero:");
    int num1 = sc.nextInt();
    System.out.println(:"Introduce un número entero:");
    int num2 = sc.nextInt();

    //Condición de error si se introduce el número 0
    if (num1 == 0 || num2 == 0) {

        System.out.println(:"Error. Debes introducir un valor distinto de 0");

    } else {

        //Creación del método que vamos a utilizar en el ejercicio
        funcionesTarea04 fun04 = new funcionesTarea04();

        //Cálculo si es múltiplo o no con la función "multiplo"
        boolean multiplo = fun04.multiplo(num1, num2);

        //Salida por pantalla
        System.out.println("El número " + num2 + " es múltiplo de " + num1 + ": " + multiplo);

    }

}
```

```

public boolean multiplo(int num1, int num2) {

    boolean multiplo = false;

    if (num1 % num2 == 0) {
        multiplo = true;
    }

    return multiplo;
}

```

****RESOLUCIÓN EJERCICIO 2****

Introduce un número entero:

3

Introduce un número entero:

0

Error. Debes introducir un valor distinto de 0

BUILD SUCCESS

Total time: 4.361 s

Finished at: 2022-11-16T14:15:04+01:00

****RESOLUCIÓN EJERCICIO 2****

Introduce un número entero:

24

Introduce un número entero:

2

El número 2 es múltiplo de 24: true

BUILD SUCCESS

Total time: 6.004 s

Finished at: 2022-11-16T14:15:39+01:00

****RESOLUCIÓN EJERCICIO 2****

Introduce un número entero:

48

Introduce un número entero:

15

El número 15 es múltiplo de 48: false

BUILD SUCCESS

Total time: 12.445 s

Finished at: 2022-11-16T14:16:30+01:00

Ejercicio 03

Implementa y utiliza un método que va a devolver un número aleatorio entre el 1 y el 100 (no debe recibir parámetros).

```
public class GarciaCutillasFranciscoJoseUT04Funciones {

    public static void main(String[] args) {

        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"
        /*
        ejerciciosUt04 ej01 = new ejerciciosUt04();
        ej01.ejercicio01();

        ejerciciosUt04 ej02 = new ejerciciosUt04();
        ej02.ejercicio02();
        */
        ejerciciosUt04 ej03 = new ejerciciosUt04();
        ej03.ejercicio03();

    }

}
```

```
public int aleatorio(int max, int min){

    int aleatorio = (int) (Math.random() * ((max - min) + 1)) + min;

    return aleatorio;

}
```

```
public void ejercicio03() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 3****");
    System.out.println(x: "");

    //Cálculo del aleatorio
    funcionesTarea04 fun04 = new funcionesTarea04();
    int aleatorio = fun04.aleatorio(max: 100, min: 1);

    System.out.println("El número aleatorio entre 1 y 100 calculado es: "+aleatorio);

}
```

```
****RESOLUCIÓN EJERCICIO 3****

El número aleatorio entre 1 y 100 calculado es: 34
-----
BUILD SUCCESS
-----
Total time: 0.913 s
Finished at: 2022-11-16T14:35:32+01:00
-----
```



```
****RESOLUCIÓN EJERCICIO 3****  
  
El número aleatorio entre 1 y 100 calculado es: 75  
-----  
BUILD SUCCESS  
-----  
Total time: 0.965 s  
Finished at: 2022-11-16T14:36:05+01:00  
-----  
|
```

Ejercicio 04

Implementa y utiliza un método que recibe dos enteros y devuelve un número aleatorio entre ambos números.

```
public class GarciaCutillasFranciscoJoseUT04Funciones {  
  
    public static void main(String[] args) {  
  
        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"  
        /*  
        ejerciciosUt04 ej01 = new ejerciciosUt04();  
        ej01.ejercicio01();  
  
        ejerciciosUt04 ej02 = new ejerciciosUt04();  
        ej02.ejercicio02();  
  
        ejerciciosUt04 ej03 = new ejerciciosUt04();  
        ej03.ejercicio03();  
        */  
        ejerciciosUt04 ej04 = new ejerciciosUt04();  
        ej04.ejercicio04();  
  
    }  
  
}
```

```

public void ejercicio04() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 4****");
    System.out.println(x: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    System.out.println(x: "Introduce un número entero para el mínimo del aleatorio:");
    int num1 = sc.nextInt();
    System.out.println(x: "Introduce un número entero para el máximo del aleatorio:");
    int num2 = sc.nextInt();

    //Cálculo del aleatorio
    funcionesTarea04 fun04 = new funcionesTarea04();
    int aleatorio = fun04.aleatorio(max: num2, min: num1);

    System.out.println("El número aleatorio entre "+num1+" y "+num2+" calculado es: "+aleatorio);

}

```

```

****RESOLUCIÓN EJERCICIO 4****

Introduce un número entero para el mínimo del aleatorio:
0
Introduce un número entero para el máximo del aleatorio:
20
El número aleatorio entre 0 y 20 calculado es: 7
-----
BUILD SUCCESS
-----
Total time: 12.868 s
Finished at: 2022-11-16T16:34:13+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 4****

Introduce un número entero para el mínimo del aleatorio:
-10
Introduce un número entero para el máximo del aleatorio:
0
El número aleatorio entre -10 y 0 calculado es: -2
-----
BUILD SUCCESS
-----
Total time: 14.224 s
Finished at: 2022-11-16T16:35:05+01:00
-----

```

Ejercicio 05

Implementa y utiliza un método que pide cinco precios y muestre por pantalla el precio de cada uno de ellos tras aplicarle un 21% de IVA. Por ejemplo, si se pasa como precio 12,5 debe devolver un mensaje que indique “El precio con IVA es: 15,125” (si salen más decimales no es problema).

```
public class GarciaCutillasFranciscoJoseUT04Funciones {

    public static void main(String[] args) {

        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"
        /*
        ejerciciosUt04 ej01 = new ejerciciosUt04();
        ej01.ejercicio01();

        ejerciciosUt04 ej02 = new ejerciciosUt04();
        ej02.ejercicio02();

        ejerciciosUt04 ej03 = new ejerciciosUt04();
        ej03.ejercicio03();

        ejerciciosUt04 ej04 = new ejerciciosUt04();
        ej04.ejercicio04();
        */
        ejerciciosUt04 ej05 = new ejerciciosUt04();
        ej05.ejercicio05();

    }

}
```

```
public void pide5PreciosDevuelveIva() {

    Scanner sc = new Scanner(System.in);
    final double IVA = 1.21;

    for (int i = 0; i < 5; i++) {

        System.out.println("Introduce el precio " + (i + 1));
        double precioSinIva = sc.nextDouble();
        double precioConIva = 0.0;

        if (precioSinIva <= 0) {

            System.out.println("Error, has introducido un precio no válido");

        } else {

            precioConIva = precioSinIva * IVA;
            System.out.println("El precio con IVA de " + precioSinIva + " € es " + precioConIva + " €");

        }

    }

}
```

```

public void ejercicio05() {

    System.out.println("****RESOLUCIÓN EJERCICIO 5****");
    System.out.println("");

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Llamada al método
    fun04.pide5PreciosDevuelveIva();

}

```

```

****RESOLUCIÓN EJERCICIO 5****

Introduce el precio 1
100
El precio con IVA de 100.0 € es 121.0 €
Introduce el precio 2
263,22
El precio con IVA de 263.22 € es 318.496200000000004 €
Introduce el precio 3
532,40
El precio con IVA de 532.4 € es 644.204 €
Introduce el precio 4
26,59
El precio con IVA de 26.59 € es 32.173899999999996 €
Introduce el precio 5
5,42
El precio con IVA de 5.42 € es 6.558199999999999 €
-----
BUILD SUCCESS
-----
Total time: 34.988 s
Finished at: 2022-11-16T17:08:19+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 5****

Introduce el precio 1
15,22
El precio con IVA de 15.22 € es 18.4162 €
Introduce el precio 2
0
Error, has introducido un precio no válido
Introduce el precio 3
633,21
El precio con IVA de 633.21 € es 766.18410000000001 €
Introduce el precio 4
-12
Error, has introducido un precio no válido
Introduce el precio 5
58,60
El precio con IVA de 58.6 € es 70.906 €
-----
BUILD SUCCESS
-----
Total time: 29.981 s
Finished at: 2022-11-16T17:14:06+01:00
-----

```

Ejercicio 06

Implementa y utiliza un método que escriba la tabla de multiplicar del número entero que se le pase como parámetro.

```
public class GarciaCutillasFranciscoJoseUT04Funciones {

    public static void main(String[] args) {

        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"
        /*
        ejerciciosUt04 ej01 = new ejerciciosUt04();
        ej01.ejercicio01();

        ejerciciosUt04 ej02 = new ejerciciosUt04();
        ej02.ejercicio02();

        ejerciciosUt04 ej03 = new ejerciciosUt04();
        ej03.ejercicio03();

        ejerciciosUt04 ej04 = new ejerciciosUt04();
        ej04.ejercicio04();

        ejerciciosUt04 ej05 = new ejerciciosUt04();
        ej05.ejercicio05();
        */
        ejerciciosUt04 ej06 = new ejerciciosUt04();
        ej06.ejercicio06();

    }

}
```

```
public void tablaMultiplicar(int tabla) {

    System.out.println("**** Tabla de multiplicar del " + tabla + " ****");
    System.out.println("x: ");

    for (int i = 0; i < 11; i++) {

        int resultado = tabla * i;
        System.out.println(tabla + " * " + i + " = " + resultado);

    }

}
```

```
public void ejercicio06() {  
  
    System.out.println(x: "****RESOLUCIÓN EJERCICIO 6****");  
    System.out.println(x: "");  
  
    //Creación de la variable Scanner para introducir valores por consola  
    Scanner sc = new Scanner(source: System.in);  
  
    //Petición de la tabla de multiplicar por consola  
    System.out.println(x: "Introduce la tabla de multiplicar que quieras resolver:");  
    int tabla = sc.nextInt();  
  
    //Creación de la variable para usar el método del ejercicio  
    funcionesTarea04 fun04 = new funcionesTarea04();  
  
    //Llamada al método  
    fun04.tablaMultiplicar(tabla);  
  
}
```

```
**** Tabla de multiplicar del 5 ****
```

```
5 * 0 = 0  
5 * 1 = 5  
5 * 2 = 10  
5 * 3 = 15  
5 * 4 = 20  
5 * 5 = 25  
5 * 6 = 30  
5 * 7 = 35  
5 * 8 = 40  
5 * 9 = 45  
5 * 10 = 50
```

```
-----  
BUILD SUCCESS  
-----
```

```
Total time: 6.041 s  
Finished at: 2022-11-16T17:34:59+01:00  
-----
```

```
**** Tabla de multiplicar del 26 ****
```

```
26 * 0 = 0
26 * 1 = 26
26 * 2 = 52
26 * 3 = 78
26 * 4 = 104
26 * 5 = 130
26 * 6 = 156
26 * 7 = 182
26 * 8 = 208
26 * 9 = 234
26 * 10 = 260
```

```
-----
BUILD SUCCESS
-----
```

```
Total time: 3.558 s
```

```
Finished at: 2022-11-16T17:36:03+01:00
-----
```

```
**** Tabla de multiplicar del -8 ****
```

```
-8 * 0 = 0
-8 * 1 = -8
-8 * 2 = -16
-8 * 3 = -24
-8 * 4 = -32
-8 * 5 = -40
-8 * 6 = -48
-8 * 7 = -56
-8 * 8 = -64
-8 * 9 = -72
-8 * 10 = -80
```

```
-----
BUILD SUCCESS
-----
```

```
Total time: 4.913 s
```

```
Finished at: 2022-11-16T17:36:37+01:00
```

Ejercicio 07

Implementa y utiliza un método que tenga como reciba dos parámetros, en el primero va a recibir el número del DNI y en el segundo va a recibir una única letra que puede ser L o C. Si recibe L devolverá la letra de dicho DNI y si recibe C lo devolverá de manera completa.

Ejemplo si recibe de parámetros: 12345678 y L el método devolverá Z y si recibe de parámetros 12345678 y C el método devolverá 12345678X.

```
public class GarciaCutillasFranciscoJoseUT04Funciones {  
  
    public static void main(String[] args) {  
  
        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"  
        /*  
        ejerciciosUt04 ej01 = new ejerciciosUt04();  
        ej01.ejercicio01();  
  
        ejerciciosUt04 ej02 = new ejerciciosUt04();  
        ej02.ejercicio02();  
  
        ejerciciosUt04 ej03 = new ejerciciosUt04();  
        ej03.ejercicio03();  
  
        ejerciciosUt04 ej04 = new ejerciciosUt04();  
        ej04.ejercicio04();  
  
        ejerciciosUt04 ej05 = new ejerciciosUt04();  
        ej05.ejercicio05();  
  
        ejerciciosUt04 ej06 = new ejerciciosUt04();  
        ej06.ejercicio06();  
        */  
        ejerciciosUt04 ej07 = new ejerciciosUt04();  
        ej07.ejercicio07();  
  
    }  
}
```



```
public String numeroDni(int numDni, String accionARealizar) {  
  
    int modulo = numDni % 23;  
    char letraDni = ' ';  
    boolean errorDni = false;  
    String salida = "";  
  
    //Condición para DNI correcto y cálculo de letra  
    if ((numDni / 10000000 != 0) && (numDni <= 99999999 && numDni > 0)) {  
  
        switch (modulo) {  
            case 0:  
                letraDni = 'T';  
                break;  
            case 1:  
                letraDni = 'R';  
                break;  
            case 2:  
                letraDni = 'W';  
                break;  
            case 3:  
                letraDni = 'A';  
                break;  
            case 4:  
                letraDni = 'G';  
                break;  
  
            case 6:  
                letraDni = 'Y';  
                break;  
            case 7:  
                letraDni = 'F';  
                break;  
            case 8:  
                letraDni = 'P';  
                break;  
            case 9:  
                letraDni = 'D';  
                break;  
            case 10:  
                letraDni = 'X';  
                break;  
            case 11:  
                letraDni = 'B';  
                break;  
            case 12:  
                letraDni = 'N';  
                break;  
            case 13:  
                letraDni = 'J';  
                break;  
            case 14:  
                letraDni = 'Z';  
                break;  
        }  
    }  
}
```

```
    case 5:
        letraDni = 'M';
        break;
    case 6:
        letraDni = 'Y';
        break;
    case 7:
        letraDni = 'F';
        break;
    case 8:
        letraDni = 'P';
        break;
    case 9:
        letraDni = 'D';
        break;
    case 10:
        letraDni = 'X';
        break;
    case 11:
        letraDni = 'B';
        break;
    case 12:
        letraDni = 'N';
        break;
    case 13:
        letraDni = 'J';
        break;
    case 14:
        letraDni = 'Z';
        break;
    case 15:
        letraDni = 'S';
        break;
    case 16:
        letraDni = 'Q';
        break;
    case 17:
        letraDni = 'V';
        break;
    case 18:
        letraDni = 'H';
        break;
    case 19:
        letraDni = 'L';
        break;
    case 20:
        letraDni = 'C';
        break;
    case 21:
        letraDni = 'K';
        break;
    case 22:
        letraDni = 'E';
        break;
}
```

```

    } else {
        // System.out.println("Error. No has introducido un DNI válido");
        salida = "Error. No has introducido un DNI válido";
        errorDni = true;
    }

    //Tipo de acción a realizar
    if (!errorDni) {
        switch (accionARealizar) {
            case "L":
                salida = "Al DNI introducido le corresponde la letra: " + letraDni;
                break;
            case "C":
                salida = "El DNI completo introducido sería: " + numDni + letraDni;
                break;
            default:
                salida = "Debes introducir el carácter 'L' o 'C' únicamente";
        }
    }
    return salida;
}

```

```

public void ejercicio07() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 7****");
    System.out.println(x: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(source: System.in);

    //Petición del número de DNI y acción a realizar
    System.out.println(x: "Introduce el número del DNI:");
    int numDni = sc.nextInt();

    System.out.println("Introduce 'L' para recibir la letra del DNI únicamente, "
        + "o 'C' para recibir el DNI completo:");
    String accionARealizar = sc.next();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Llamada al método
    String salida = fun04.numeroDni(numDni, accionARealizar);

    System.out.println(x: salida);

}

```

****RESOLUCIÓN EJERCICIO 7****

Introduce el número del DNI:

7852

Introduce 'L' para recibir la letra del DNI únicamente, o 'C' para recibir el DNI completo:

L

Error. No has introducido un DNI válido

BUILD SUCCESS

Total time: 14.119 s

Finished at: 2022-11-16T20:14:08+01:00

****RESOLUCIÓN EJERCICIO 7****

Introduce el número del DNI:

12345678

Introduce 'L' para recibir la letra del DNI únicamente, o 'C' para recibir el DNI completo:

l

Debes introducir el carácter 'L' o 'C' únicamente

BUILD SUCCESS

Total time: 10.059 s

Finished at: 2022-11-16T20:14:50+01:00

****RESOLUCIÓN EJERCICIO 7****

Introduce el número del DNI:

12345678

Introduce 'L' para recibir la letra del DNI únicamente, o 'C' para recibir el DNI completo:

L

Al DNI introducido le corresponde la letra: Z

BUILD SUCCESS

Total time: 7.056 s

Finished at: 2022-11-16T20:15:28+01:00

****RESOLUCIÓN EJERCICIO 7****

Introduce el número del DNI:

12345678

Introduce 'L' para recibir la letra del DNI únicamente, o 'C' para recibir el DNI completo:

C

El DNI completo introducido sería: 12345678Z

BUILD SUCCESS

Total time: 7.559 s

Finished at: 2022-11-16T20:16:02+01:00

Ejercicio 08

Implementa un método que devuelva el mayor de dos números. Utilizando dicho método (las veces que sea necesario) crea un nuevo método que recibiendo cuatro números enteros devuelva el mayor de los cuatro.

```
public class GarciaCutillasFranciscoJoseUT04Funciones {  
  
    public static void main(String[] args) {  
  
        //Llamada a los ejercicios resueltos en la clase "ejerciciosUt04"  
        /*  
        ejerciciosUt04 ej01 = new ejerciciosUt04();  
        ej01.ejercicio01();  
  
        ejerciciosUt04 ej02 = new ejerciciosUt04();  
        ej02.ejercicio02();  
  
        ejerciciosUt04 ej03 = new ejerciciosUt04();  
        ej03.ejercicio03();  
  
        ejerciciosUt04 ej04 = new ejerciciosUt04();  
        ej04.ejercicio04();  
  
        ejerciciosUt04 ej05 = new ejerciciosUt04();  
        ej05.ejercicio05();  
  
        ejerciciosUt04 ej06 = new ejerciciosUt04();  
        ej06.ejercicio06();  
  
        ejerciciosUt04 ej07 = new ejerciciosUt04();  
        ej07.ejercicio07();  
        */  
        ejerciciosUt04 ej08 = new ejerciciosUt04();  
        ej08.ejercicio08();  
  
    }  
  
}
```

```
public void ejercicio08() {  
  
    System.out.println(x: "****RESOLUCIÓN EJERCICIO 8****");  
    System.out.println(x: "");  
  
    //Creación de la variable Scanner para introducir valores por consola  
    Scanner sc = new Scanner(System.in);  
  
    //Petición de los números por consola  
    System.out.println(x: "Introduce el primer número:");  
    int num1 = sc.nextInt();  
  
    System.out.println(x: "Introduce el segundo número:");  
    int num2 = sc.nextInt();  
  
    System.out.println(x: "Introduce el tercer número:");  
    int num3 = sc.nextInt();  
  
    System.out.println(x: "Introduce el cuarto número:");  
    int num4 = sc.nextInt();  
  
    //Creación de la variable para usar el método del ejercicio  
    funcionesTarea04 fun04 = new funcionesTarea04();  
  
    //Llamada al método y asignación del mayor  
    int mayor = fun04.devuelveMayor4(num1, num2, num3, num4);  
  
    System.out.println("El mayor de "+num1+", "+num2+", "+num3+", "+num4+" es: "+mayor);  
}
```

```

public int devuelveMayor2(int num1, int num2) {

    int mayor = 0;

    if (num1 >= num2) {
        mayor = num1;
    } else {
        mayor = num2;
    }

    return mayor;

}

public int devuelveMayor4(int num1, int num2, int num3, int num4){

    int mayor1 = 0;
    int mayor2 = 0;
    int mayor3 = 0;

    mayor1 = devuelveMayor2(num1, num2);
    mayor2 = devuelveMayor2(num1: num3, num2: num4);
    mayor3 = devuelveMayor2(num1: mayor1, num2: mayor2);

    return mayor3;

}

```

****RESOLUCIÓN EJERCICIO 8****

```

Introduce el primer número:
2
Introduce el segundo número:
2
Introduce el tercer número:
3
Introduce el cuarto número:
4
El mayor de 2, 2, 3, 4 es: 4

```

BUILD SUCCESS

```

Total time: 8.457 s
Finished at: 2022-11-17T12:01:35+01:00
-----

```

```
****RESOLUCIÓN EJERCICIO 8****

Introduce el primer número:
0
Introduce el segundo número:
3
Introduce el tercer número:
45
Introduce el cuarto número:
158
El mayor de 0, 3, 45, 158 es: 158
```

BUILD SUCCESS

Total time: 9.330 s
Finished at: 2022-11-17T12:02:39+01:00

```
****RESOLUCIÓN EJERCICIO 8****

Introduce el primer número:
-98
Introduce el segundo número:
-74
Introduce el tercer número:
-236
Introduce el cuarto número:
-5
El mayor de -98, -74, -236, -5 es: -5
```

BUILD SUCCESS

Total time: 11.573 s
Finished at: 2022-11-17T12:03:15+01:00

Ejercicio 09

Implementa y utiliza un método que reciba un carácter y devuelva verdadero si este es una vocal y false en caso contrario (debe devolver un boolean).

```
ejerciciosUt04 ej09 = new ejerciciosUt04();
ej09.ejercicio09();
}
```

```
public boolean vocal(String vocal) {
    boolean esVocal = false;

    if ("a".equals(vocal) || "e".equals(vocal) || "i".equals(vocal) || "o".equals(vocal) || "u".equals(vocal)) {
        esVocal = true;
    }

    return esVocal;
}
```

```
public void ejercicio09() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 9****");
    System.out.println(x: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    //Petición de los números por consola
    System.out.println(x: "Introduce un carácter en minúscula:");
    String letra = sc.next();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Llamada al método
    boolean vocal = fun04.vocal(vocal: letra);

    //Condición vocal o no vocal
    if (vocal) {
        System.out.println(x: "El carácter introducido es una vocal");
    } else {
        System.out.println(x: "El carácter introducido no es una vocal");
    }
}
```

```
****RESOLUCIÓN EJERCICIO 9****

Introduce un carácter en minúscula:
0
El carácter introducido no es una vocal
-----
BUILD SUCCESS
-----
Total time: 4.094 s
Finished at: 2022-11-17T19:26:49+01:00
```

```
****RESOLUCIÓN EJERCICIO 9****

Introduce un carácter en minúscula:
c
El carácter introducido no es una vocal
-----
BUILD SUCCESS
-----
Total time: 7.691 s
Finished at: 2022-11-17T19:27:21+01:00
-----
```

```
****RESOLUCIÓN EJERCICIO 9****

Introduce un carácter en minúscula:
i
El carácter introducido es una vocal
-----
BUILD SUCCESS
-----
Total time: 3.332 s
Finished at: 2022-11-17T19:27:55+01:00
-----
```

Ejercicio 10

Implementa y utiliza un método que calcule x^y (x elevado a y). X es un número real e Y es un número entero no negativo.

```
ejerciciosUt04 ej10 = new ejerciciosUt04();
ej10.ejercicio10();
```

```
public double potencia(double num, int potencia){

    double pot = potencia;

    double resultado = Math.pow(a:num, b:pot);

    return resultado;

}
```

```
public void ejercicio10() {

    System.out.println(s: "****RESOLUCIÓN EJERCICIO 10****");
    System.out.println(s: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(source: System.in);

    //Petición de los números por consola
    System.out.println(s: "Introduce un número:");
    double num = sc.nextDouble();

    System.out.println(s: "Introduce su potencia:");
    int potencia = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Condición error o potencia correcta
    if (potencia > 0) {

        //Llamada al método
        double resultado = fun04.potencia(num, potencia);

        System.out.println("El resultado del número " + num + " elevado a " + potencia + " es: " + resultado);

    } else {
        System.out.println(s: "La potencia debe ser un número positivo");
    }

}
```

```
****RESOLUCIÓN EJERCICIO 10****

Introduce un número:
-2
Introduce su potencia:
6
El resultado del número -2.0 elevado a 6 es: 64.0
-----
BUILD SUCCESS
-----
Total time: 8.033 s
Finished at: 2022-11-17T19:57:05+01:00
-----
```

```
****RESOLUCIÓN EJERCICIO 10****

Introduce un número:
5
Introduce su potencia:
-1
La potencia debe ser un número positivo
-----
BUILD SUCCESS
-----
Total time: 7.220 s
Finished at: 2022-11-17T19:56:30+01:00
-----
```

```
****RESOLUCIÓN EJERCICIO 10****

Introduce un número:
2
Introduce su potencia:
6
El resultado del número 2.0 elevado a 6 es: 64.0
-----
BUILD SUCCESS
-----
Total time: 4.157 s
Finished at: 2022-11-17T19:55:47+01:00
-----
```

Ejercicio 11

Implementa y utiliza un método que a partir del número de días, horas y minutos devuelve el número de segundos que existen en los datos de entrada.

```
ejerciciosUt04 ej11 = new ejerciciosUt04();
ej11.ejercicioll();
```

```
public int devuelveSegundos(int dias, int horas, int minutos){
    int segundos = (dias * (24 * 3600)) + (horas * 3600) + (minutos * 60);
    return segundos;
}
```

```
public void ejercicioll() {
    System.out.println(x: "****RESOLUCIÓN EJERCICIO 11****");
    System.out.println(x: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    //Petición de los datos por pantalla
    System.out.println(x: "Introduce los días:");
    int dias = sc.nextInt();

    System.out.println(x: "Introduce las horas:");
    int horas = sc.nextInt();

    System.out.println(x: "Introduce los minutos:");
    int minutos = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Condición para cálculo o error
    if (dias >= 0 && horas >= 0 && minutos >= 0) {
        //Llamada al método
        int segundosTotales = fun04.devuelveSegundos(dias, horas, minutos);

        System.out.println("El resultado de " + dias + " días, " + horas + " horas y "
            + minutos + " minutos en segundos es: " + segundosTotales);
    } else {
        System.out.println(x: "Los días, horas y minutos no pueden ser negativos");
    }
}
```

```
****RESOLUCIÓN EJERCICIO 11****

Introduce los días:
1
Introduce las horas:
0
Introduce los minutos:
0
El resultado de 1 días, 0 horas y 0 minutos en segundos es: 86400
-----
BUILD SUCCESS
-----
Total time: 9.810 s
Finished at: 2022-11-17T20:19:48+01:00
```

```
****RESOLUCIÓN EJERCICIO 11****

Introduce los días:
12
Introduce las horas:
3
Introduce los minutos:
54
El resultado de 12 días, 3 horas y 54 minutos en segundos es: 1050840
-----
BUILD SUCCESS
-----
Total time: 9.835 s
Finished at: 2022-11-17T20:20:41+01:00
```

```
****RESOLUCIÓN EJERCICIO 11****

Introduce los días:
5
Introduce las horas:
6
Introduce los minutos:
-1
Los días, horas y minutos no pueden ser negativos
-----
BUILD SUCCESS
-----
Total time: 6.502 s
Finished at: 2022-11-17T20:21:26+01:00
```

Ejercicio 12

Implementa y utiliza un método al que se le pasan las horas y minutos de dos instantes de tiempo y devolverá la cantidad de minutos de diferencia que existen entre los dos instantes utilizados (el primer instante de tiempo siempre debe ser mayor al segundo). Ejemplo si paso que el primer instante son las 18 horas y 46 minutos y el segundo instante son 14 horas y 42 minutos, la diferencia en minutos serán 4 horas y 4 minutos, pero al tener que devolverlo en minutos se devolverá 244.

```
ejerciciosUt04 ej12 = new ejerciciosUt04();
ej12.ejerciciol2();

}
```

```
public int devuelveMinutos(int horas1, int minutos1, int horas2, int minutos2) {

    int totalMinutos1 = (horas1 * 60) + minutos1;
    int totalMinutos2 = (horas2 * 60) + minutos2;
    int minutoSalida = 0;

    if (minutos1 >= 60 || minutos2 >= 60) {

        minutoSalida = -1;

    } else if (totalMinutos1 < totalMinutos2) {

        minutoSalida = -2;

    } else {

        minutoSalida = totalMinutos1 - totalMinutos2;

    }

    return minutoSalida;
}
```

```
public void ejerciciol2() {

    System.out.println("****RESOLUCIÓN EJERCICIO 12****");
    System.out.println("");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    //Petición de los datos por pantalla
    System.out.println("Introduce las horas del primer instante:");
    int horas1 = sc.nextInt();

    System.out.println("Introduce los minutos del primer instante:");
    int minutos1 = sc.nextInt();

    System.out.println("Introduce las horas del segundo instante:");
    int horas2 = sc.nextInt();

    System.out.println("Introduce los minutos del segundo instante:");
    int minutos2 = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();
}
```

```

/*Cálculo de la diferencia. Si no se cumple primer valor de horas y minutos
mayor que el segundo, la función devuelve el valor -2. Si no se cumple minutos menor de 60 devuelve -1*/
int resultado = fun04.devuelveMinutos(horas1, minutos1, horas2, minutos2);

//Condición para cálculo o error
if (resultado >= 0) {

    System.out.println("La diferencia en minutos de " + horas1 + " horas y " + minutos1 + " minutos, menos "
        + horas2 + " horas y " + minutos2 + " minutos es: " + resultado + " minutos");

} else if (resultado == -1) {

    System.out.println("Los minutos no pueden ser mayores de 59");

} else if (resultado == -2) {

    System.out.println("Los primeros valores de horas y minutos deben ser mayores que los segundos");

}

```

****RESOLUCIÓN EJERCICIO 12****

Introduce las horas del primer instante:

15

Introduce los minutos del primer instante:

60

Introduce las horas del segundo instante:

14

Introduce los minutos del segundo instante:

59

Los minutos no pueden ser mayores de 59

BUILD SUCCESS

Total time: 8.580 s

Finished at: 2022-11-20T12:38:21+01:00

****RESOLUCIÓN EJERCICIO 12****

Introduce las horas del primer instante:

8

Introduce los minutos del primer instante:

56

Introduce las horas del segundo instante:

10

Introduce los minutos del segundo instante:

22

Los primeros valores de horas y minutos deben ser mayores que los segundos

BUILD SUCCESS

Total time: 9.517 s

Finished at: 2022-11-20T12:36:45+01:00


```

****RESOLUCIÓN EJERCICIO 12****

Introduce las horas del primer instante:
17
Introduce los minutos del primer instante:
33
Introduce las horas del segundo instante:
17
Introduce los minutos del segundo instante:
33
La diferencia en minutos de 17 horas y 33 minutos, menos 17 horas y 33 minutos es: 0 minutos
-----
BUILD SUCCESS
-----
Total time: 10.737 s
Finished at: 2022-11-20T12:30:24+01:00

****RESOLUCIÓN EJERCICIO 12****

Introduce las horas del primer instante:
15
Introduce los minutos del primer instante:
20
Introduce las horas del segundo instante:
15
Introduce los minutos del segundo instante:
19
La diferencia en minutos de 15 horas y 20 minutos, menos 15 horas y 19 minutos es: 1 minutos
-----
BUILD SUCCESS
-----
Total time: 27.221 s
Finished at: 2022-11-20T12:29:15+01:00

```

Ejercicio 13

Haciendo uso del método del ejercicio anterior, implementa y utiliza un método al que se le pasan las horas y minutos de dos instantes de tiempo y devolverá la cantidad de minutos de diferencia que existen entre los dos instantes utilizados, pudiendo ser el segundo instante mayor que el primero. El resultado siempre deberá ser positivo.

En este ejercicio se utilizará el mismo código que en el anterior para el método, pero he tenido que modificar dicho método, porque las condiciones de error las había realizado dentro del mismo. Por lo tanto, sin modificar el método no podría usarlo en este ejercicio. Se realizará otro nuevo para calcular el valor absoluto de la diferencia.

```

ejerciciosUt04 ej13 = new ejerciciosUt04();
ej13.ejercicio13();
}

```

```

public int devuelveMinutosAbsoluto(int horas1, int minutos1, int horas2, int minutos2) {

    int totalMinutos1 = (horas1 * 60) + minutos1;
    int totalMinutos2 = (horas2 * 60) + minutos2;
    int minutoSalida = 0;

    if (minutos1 >= 60 || minutos2 >= 60) {

        minutoSalida = -1;

    } else {

        minutoSalida = Math.abs(totalMinutos1 - totalMinutos2);

    }

    return minutoSalida;
}

```

```

public void ejercicio13() {

    System.out.println("****RESOLUCIÓN EJERCICIO 13****");
    System.out.println("");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    //Petición de los datos por pantalla
    System.out.println("Introduce las horas del primer instante:");
    int horas1 = sc.nextInt();

    System.out.println("Introduce los minutos del primer instante:");
    int minutos1 = sc.nextInt();

    System.out.println("Introduce las horas del segundo instante:");
    int horas2 = sc.nextInt();

    System.out.println("Introduce los minutos del segundo instante:");
    int minutos2 = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Cálculo de la diferencia. Si no se cumple minutos menor de 60 devuelve -1
    int resultado = fun04.devuelveMinutosAbsoluto(horas1, minutos1, horas2, minutos2);

    //Condición para cálculo o error
    if (resultado != -1) {

        System.out.println("La diferencia en minutos de " + horas1 + " horas y " + minutos1 + " minutos, menos "
            + horas2 + " horas y " + minutos2 + " minutos en valor absoluto es: " + resultado + " minutos");

    } else {

        System.out.println("Los minutos no pueden ser mayores de 59");

    }

}

```

```

****RESOLUCIÓN EJERCICIO 13****

Introduce las horas del primer instante:
1
Introduce los minutos del primer instante:
12
Introduce las horas del segundo instante:
1
Introduce los minutos del segundo instante:
15
La diferencia en minutos de 1 horas y 12 minutos, menos 1 horas y 15 minutos en valor absoluto es: 3 minutos
-----
BUILD SUCCESS
-----
Total time: 20.390 s
Finished at: 2022-11-20T13:06:39+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 13****

Introduce las horas del primer instante:
1
Introduce los minutos del primer instante:
15
Introduce las horas del segundo instante:
1
Introduce los minutos del segundo instante:
12
La diferencia en minutos de 1 horas y 15 minutos, menos 1 horas y 12 minutos en valor absoluto es: 3 minutos
-----
BUILD SUCCESS
-----
Total time: 6.379 s
Finished at: 2022-11-20T13:07:19+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 13****

Introduce las horas del primer instante:
25
Introduce los minutos del primer instante:
63
Introduce las horas del segundo instante:
12
Introduce los minutos del segundo instante:
55
Los minutos no pueden ser mayores de 59
-----
BUILD SUCCESS
-----
Total time: 10.402 s
Finished at: 2022-11-20T13:08:03+01:00
-----

```

Ejercicio 14

Implementa y utiliza un método que devuelva true o false si el número pasado es primo (un número es primo si no es divisible por ningún número desde el 2 hasta dicho número - 1, es decir, si el resto de la división por todos los números nunca va a dar cero).

```
ejerciciosUt04 ejl4 = new ejerciciosUt04();  
ejl4.ejercicio14();  
}
```

```
public boolean esPrimo(int numero) {  
    boolean esPrimo = true;  
    for (int i = 2; i < numero - 1; i++) {  
        if (numero % i == 0) {  
            esPrimo = false;  
        }  
    }  
    return esPrimo;  
}
```

```

public void ejerciciol4() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 14****");
    System.out.println(x: "");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(System.in);

    //Petición de los datos por pantalla
    System.out.println(x: "Introduce un número para calcular si es primo o no:");
    int numero = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();

    //Cálculo para saber si es primo o no
    boolean resultado = fun04.esPrimo(numero);

    //Condición para cálculo o error
    if (numero > 1) {

        System.out.println("El número introducido es primo: "+resultado);

    } else {

        System.out.println(x: "El número que introduzcas debe ser mayor de 1");

    }

}

```

```

****RESOLUCIÓN EJERCICIO 14****

Introduce un número para calcular si es primo o no:
35
El número introducido es primo: false

-----
BUILD SUCCESS
-----
Total time: 7.427 s
Finished at: 2022-11-20T13:45:40+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 14****

Introduce un número para calcular si es primo o no:
43
El número introducido es primo: true

-----
BUILD SUCCESS
-----
Total time: 7.909 s
Finished at: 2022-11-20T13:46:45+01:00
-----

```

```

****RESOLUCIÓN EJERCICIO 14****

Introduce un número para calcular si es primo o no:
1
El número que introduzcas debe ser mayor de 1
-----
BUILD SUCCESS
-----
Total time: 3.046 s
Finished at: 2022-11-20T13:47:10+01:00
-----

```

Ejercicio 15

Implementa y utiliza un método que imprima un mensaje con todos los números primos que van desde un número hasta otro. Por ejemplo, si pasa el número 150 y el 310 nos mostraría el mensaje: 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307.

```

ejerciciosUt04 ej15 = new ejerciciosUt04();
ej15.ejercicio15();
}

```

```

public void primosIntervalo(int numeroInicio, int numeroFin) {

    for (int i = numeroInicio; i <= numeroFin; i++) {

        if (esPrimo(numero: i)) {
            System.out.println(":" i);
        }
    }
}

```

```

public void ejercicio15() {

    System.out.println("****RESOLUCIÓN EJERCICIO 15****");
    System.out.println(":");

    //Creación de la variable Scanner para introducir valores por consola
    Scanner sc = new Scanner(source: System.in);

    //Petición de los datos por pantalla
    System.out.println(":Introduce el rango bajo para calcular los primos:");
    int numeroInicio = sc.nextInt();

    System.out.println(":Introduce el rango alto para calcular los primos:");
    int numeroFin = sc.nextInt();

    //Creación de la variable para usar el método del ejercicio
    funcionesTarea04 fun04 = new funcionesTarea04();
}

```

```

//Condición para cálculo o error
if (numeroInicio > 1 && numeroFin > 1) {

    System.out.println("Los números primos del rango desde el "+numeroInicio+" al "+numeroFin+" son:");

    //Llamada a la función para que muestre los primos del rango
    fun04.primosIntervalo(numeroInicio, numeroFin);

} else {

    System.out.println("El número que introduzcas debe ser mayor de 1");

}
}

```

****RESOLUCIÓN EJERCICIO 15****

Introduce el rango bajo para calcular los primos:

150

Introduce el rango alto para calcular los primos:

310

Los números primos del rango desde el 150 al 310 son:

151

157

163

167

173

179

181

191

193

197

199

211

223

227

229

233

239

241

251

257

263

269

271

277

281

283

293

307

****RESOLUCIÓN EJERCICIO 15****

Introduce el rango bajo para calcular los primos:

-1

Introduce el rango alto para calcular los primos:

100

El número que introduzcas debe ser mayor de 1

BUILD SUCCESS

Total time: 6.873 s

Finished at: 2022-11-20T17:52:27+01:00