

UF5. [PAC02] Solución

Para la realización de las siguientes actividades deberéis ayudaros del libro de texto, capítulo 10, y consultar, si lo creéis necesario, internet.

Debéis subir un único archivo comprimido que contenga los ficheros .java o los proyectos creados, con los comentarios que creáis necesarios.

Los ejercicios tienen la siguiente puntuación: el ejercicio 1 y el 3 valen 2 puntos cada uno, mientras que el ejercicio 2 tiene un valor de 6 puntos.

Se valorará positivamente el correcto uso del nombre en los nombres de variables, métodos y clases.

Recordad que la fecha límite para la entrega de esta PAC es el 18 de NOVIEMBRE.

Sugerencia: Para la realización de estos ejercicios es necesario utilizar Eclipse.

Información: Cualquier PAC copiada y/o en la que se haya utilizado “copy-paste” de código ya escrito será puntuada con un 0.

Actividades

Parte práctica

1. Cree en el paquete **ilerna.utiles** una clase **UtilesArrays** con los siguientes métodos:
 - a. Un método que dado un número **n**, un límite inferior y un límite superior, devuelva un **array** con **n** números enteros aleatorios con valores comprendidos entre el límite inferior y el límite superior.
Ejemplo:
Crear un array con 6 enteros aleatorios comprendidos entre 5 y 15.
Array aleatorio: [8, 12, 7, 13, 5, 14]
 - b. Un método que dado un **array** devuelva el mismo **array** pero invertido.
Ejemplo:
Crear un array inverso al [1,2,3,4,5]
Array inverso: [5, 4, 3, 2, 1]
 - c. Un método que reciba un número **entero tam** y devuelva un **array** de enteros de tamaño **tam** donde cada elemento sea igual a su posición

natural (1 para la primera posición, 2 para la segunda, y así sucesivamente)

- d. Un método que dados dos **arrays** del mismo tamaño devuelva otro **array** que contenga en cada posición el resultado de multiplicar los elementos situados en la misma posición de ambos **arrays**.

```
package ilerana.utiles;
import java.util.Random;

public class UtilesArrays {
    static Random rnd = new Random();

    public static int[] random_array (int n, int min, int max) {
        int numero [] = new int[n];
        max = max - min + 1;
        for (int i = 0; i < n; i++) {
            numero[i] = (int) (rnd.nextDouble() * max + min);
        }
        return (numero);
    }

    public static int[] invertir (int inout[]) {
        int j = inout.length - 1, n = inout.length;
        int aux[] = new int[n];

        for (int i = 0; i < n; i++) {
            aux[i] = inout[j--];
        }

        for (int i = 0; i < n; i++) {
            inout[i] = aux[i];
        }

        return (inout);
    }

    public static int[] natpos (int tam) {
        int aux[] = new int[tam];
        for (int i = 0; i < tam; i++) {
            aux[i] = i + 1;
        }
        return aux;
    }

    public static int[] mult_arrays (int a1[], int a2[]) {
        int res[] = new int[a1.length];
        for (int i = 0; i < a1.length; i++) {
            res[i] = a1[i] * a2[i];
        }
        return res;
    }
}
```

2. Cree en el paquete **ilerna.utiles** una clase **Cadenas** con los siguientes métodos:
 - a. Un método que reciba una cadena de texto y devuelva el número de vocales que contiene la cadena. Tengan en cuenta que las vocales pueden aparecer en minúsculas o en mayúsculas. *No tenga en cuenta las posibles vocales con tildes, diéresis u otros signos ortográficos.*
 - b. Un método que dado un número y un carácter indique si el DNI formado por la unión de ambos es correcto.
El método para calcular la letra del DNI puede verse en el siguiente enlace:
https://es.wikipedia.org/wiki/N%C3%BAmero_de_identificaci%C3%B3n_fiscal
 - c. Un método que reciba una cadena de texto y devuelva otra cadena igual a la primera pero con los caracteres en orden inverso.
 - d. Un método que reciba una cadena de texto y un carácter y devuelva el número de apariciones del carácter en la cadena. *Utilice los métodos **indexOf** y **lastIndexOf** de la clase **String** para acotar la búsqueda.*

```
package ilerna.utiles;

public class Cadenas {

    public int n_vocales (String cadena) {
        int cont = 0;
        cadena = cadena.toLowerCase();

        for (int i = 0; i < cadena.length(); i++) {
            if (cadena.charAt(i) == 'a' || cadena.charAt(i) ==
'e' || cadena.charAt(i) == 'i' || cadena.charAt(i) == 'o' ||
cadena.charAt(i) == 'u') {
                cont++;
            }
        }

        return cont;
    }

    public boolean check_dni (int ndni, char letra) {
        String ldni = "TRWAGMYFPDXBNJZSQVHLCKE";
        int mod = ndni % 23;
        if (ldni.charAt(mod) == Character.toUpperCase(letra)) {
            return true;
        } else {
            return false;
        }
    }

    public String invertir (String cadena) {
        String invertida = "";

        for (int i = cadena.length() - 1; i >= 0; i--) {
```

```

        invertida += cadena.charAt(i);
    }

    return invertida;
}

public int apariciones (String in, char c) {
    int numero = 0, cont = 0;

    while (in.indexOf(c, numero) != -1) {
        numero = in.indexOf(c, numero) + 1;
        cont++;
    }

    return cont;
}
}

```

3. Escriba un programa principal para probar todos los métodos anteriores.

```

package ilerna.utiles;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Principal {
    static Scanner scan = new Scanner(System.in);

    public static void main (String[] args) {
        String cadena;
        int dni, n, limSup, limInf;
        Character letra;
        Cadenas metodo = new Cadenas();
        int[] array, array2;

        System.out.println("Escriba una cadena: ");
        cadena = scan.nextLine();
        System.out.println("Escriba un DNI: ");
        dni = scan.nextInt();
        System.out.println("Escriba la letra del DNI: ");
        letra = scan.next().charAt(0);

        System.out.println("El número de vocales de la cadena es: " + metodo.n_vocales(cadena));
        System.out.println("La cadena invertida es: " + metodo.invertir(cadena));
        System.out.println("El DNI es: " + metodo.check_dni(dni, letra));

        int numero = metodo.apariciones(cadena, letra) ;
        System.out.println(numero);
        System.out.println("El caracter " + letra + " aparece " + metodo.apariciones(cadena, letra) + " veces ");

        System.out.println("Escribe el número de elementos: ");
    }
}

```

```

n = scan.nextInt();
System.out.println("Escribe el límite superior: ");
limSup = scan.nextInt();
System.out.println("Escribe el límite inferior: ");
limInf = scan.next().charAt(0);

array = UtilesArrays.random_array(n, limSup, limInf);

System.out.print ("El array es: ");
for (int i = 0; i < n; i++) {
    System.out.print(" " + array[i] + " ");
}
System.out.println("");

array = UtilesArrays.invertir(array);

System.out.print ("El array invertido es: ");
for (int i = 0; i < n; i++) {
    System.out.print(" " + array[i] + " ");
}
System.out.println("");

array2 = UtilesArrays.natpos (n);

System.out.print ("Las posiciones del array son: ");
for (int i = 0; i < n; i++) {
    System.out.print(" " + array2[i] + " ");
}
System.out.println("");

array = UtilesArrays.mult_arrays(array, array2);

System.out.print ("La multiplicación de los arrays es: ");
for (int i = 0; i < n; i++) {
    System.out.print(" " + array[i] + " ");
}
System.out.println("");

    }
}

```