

ArrayList en Java

ArrayList de ArrayList

Por último vamos a rizar el rizo y vamos a poner un ejemplo de como trabajar con ArrayList de ArrayList. Para ello vamos a hacer un ejemplo que en principio puede parecer complejo pero que en realidad no lo es, solo es un poco lioso y hay que ir con cuidado. INTENTA HACER POR TU CUENTA ESTE EJERCICIO ANTES DE MIRARLO, PORQUE SI LO CONSIGUES HACER, NO HABRA TRABAJO CON ARRAYLIST QUE SE TE RESISTA

El ejercicio es el siguiente. Dado un fichero con una serie de números, hay que eliminar todos los ceros que hay en el fichero, y si una línea está llena de ceros, hay que eliminar la línea. En este caso tenemos en el fichero el contenido siguiente:

```
1 2 0 3 0 7
0 0 0 0
0 6 9 8 0 9 6
0 0 0 0 7 9 0 0
0 0 0 0 0
8 7 0 8 9 0 8
```

Y lo que hay que conseguir es transformar esos datos en los siguientes:

```
1 2 3 7
6 9 8 9 6
7 9
8 7 8 9 8
```

Es decir se eliminan todos los ceros y como las filas 2 y 5 contienen todo ceros, se han de eliminar esas filas. Para ello nos tenemos que declarar un ArrayList de ArrayList de enteros de la siguiente forma:

```
ArrayList<ArrayList<Integer>> arrayListNumeros = new ArrayList<ArrayList<Integer>>();
```

Lo siguiente que haremos sera leer los números del fichero y nos crearemos por cada fila un ArrayList de enteros en el que guardaremos cada número en una posición del ArrayList y luego guardaremos ese ArrayList en el ArrayList que hemos llamado *"arrayListNumeros"*. Esto lo hacemos de la siguiente forma:

```
// Instanciamos el fichero donde estan los datos
File fichero = new File(nombreFichero);
Scanner s = null;

try {
    // Leemos el contenido del fichero
    System.out.println("... Leemos el contenido del fichero ...");
    s = new Scanner(fichero);
    // Obtengo los datos de los partidos de fútbol del fichero
    while (s.hasNextLine()){
        String linea = s.nextLine();    // Obtengo una línea del fi
        chero con los numeros
        String [] cortarString = linea.split(" ");    //
        Obtengo los numeros de la línea en un array
        ArrayList<Integer> numeros = new ArrayList<Integer>();
        // Pongo los numeros de la fila en el ArrayList
        for (int i=0; i<cortarString.length; i++){
            numeros.add(Integer.parseInt(cortarString[i]));
        }
        // Añado el ArrayList de enteros al ArrayList de ArrayList
        arrayListNumeros.add(numeros);
    }
}
```

```

} catch (Exception e) {
    e.printStackTrace();
} finally{
    try {
        if (s != null)
            s.close();
    } catch (Exception e2) {
        e2.printStackTrace();
    }
}
}

```

Para comprobar que hemos copiado bien los datos y para recorrer el ArrayList de ArrayList de Enteros, vamos a imprimir por pantalla el contenido del ArrayList y lo hacemos de la siguiente forma:

```

// Para ver que hemos leído bien los números del fichero, recorremos el Array
List de
// ArrayList y lo mostramos por pantalla.
System.out.println("n... Contenido de los ArrayList del ArrayList ...");
Iterator<ArrayList<Integer>> itrArrayListNumeros = arrayListNumeros.iterator(
);
while(itrArrayListNumeros.hasNext()){
    ArrayList<Integer> numeros = itrArrayListNumeros.next();
    Iterator<Integer> itrNumeros = numeros.iterator();
    while(itrNumeros.hasNext()){
        int numero = itrNumeros.next();
        System.out.print(numero+" ");
    }
    System.out.println();
}
}

```

Como salida tenemos el contenido del fichero.

Ahora lo que vamos a hacer es recorrer el ArrayList de ArrayList de Enteros, y eliminaremos de los ArrayList, todos los elementos que tengan como contenido '0'. Esto lo hacemos de la siguiente forma:

```
// Recorremos los ArrayList que hay en el ArrayList y eliminamos los numeros
que son cero
itrArrayListNumeros = arrayListNumeros.iterator();
while(itrArrayListNumeros.hasNext()){
    ArrayList<Integer> numeros = itrArrayListNumeros.next();
    Iterator<Integer> itrNumeros = numeros.iterator();
    while(itrNumeros.hasNext()){
        int numero = itrNumeros.next();
        if (numero == 0)
            itrNumeros.remove();
    }
}
```

Ahora borramos del ArrayList, los ArrayList que están vacíos, preguntando con el método *"isEmpty()"*. Esto lo hacemos de la siguiente forma:

```
// Recorremos el ArrayList de los ArrayList y eliminamos los ArrayList que no
// contengan elementos
itrArrayListNumeros = arrayListNumeros.iterator();
while(itrArrayListNumeros.hasNext()){
    ArrayList<Integer> numeros = itrArrayListNumeros.next();
    // Si el ArrayList esta vacio, lo eliminamos del ArrayList
    if (numeros.isEmpty())
        itrArrayListNumeros.remove();
}
```

Con todo esto ya hemos eliminado los ceros, y los ArrayList que estaban vacíos. Ahora solo queda recorrer de nuevo el ArrayList de ArrayList y mostrar el contenido tras realizar las eliminaciones de ceros. Esto lo hacemos de la siguiente manera:

```
// Mostramos el ArrayList tras eliminar todos los ceros
String salida = "";
```

```
System.out.println("n... Salida sin ceros ...");
```

```
itrArrayListNumeros = arrayListNumeros.iterator();
while(itrArrayListNumeros.hasNext()){
    ArrayList<Integer> numeros = itrArrayListNumeros.next();
    Iterator<Integer> itrNumeros = numeros.iterator();
    while(itrNumeros.hasNext()){
        int numero = itrNumeros.next();
        salida += numero+" ";
    }
    salida += "n";
}

System.out.println(salida);
```

Tras ejecutar este código, vemos que tenemos la salida esperada, que es la siguiente:

```
1 2 3 7
6 9 8 9 6
7 9
8 7 8 9 8
```

Calcule el tamaño de cada arraylist, tanto del general como de cada uno de los internos

Con todo esto ya hemos visto prácticamente todo sobre los ArrayList, con ejemplo de como tratarlos con elementos atómicos (String, int, float, double, ...) y con objetos y sobre todo ver como se trabaja con ArrayList de ArrayList que como se ha visto la complejidad esta en saber recorrer esa estructura de datos.

Adicionar:

Haga un menú donde se pueda visualizar cada una de las opciones,

- 1- Cargar el arrayList-ArrayList
- 2- Eliminar un entero dado
- 3- Mostar el ArrayList
- 4- Borrar ArrayList vacíos
- 5- Calcular tamaño de cada arrayList
- 6- Adicionar Datos

como tambien

- 1- Realizar la suma y el promedio de cada fila (ArrayList)
- 2- Poderle adicionar un(s) elementos en una fila(ArrayList) determinado
- 3- Diseñe un ejemplo donde se utilice una arraylist de arralist y dentro de este ultimo un objeto con varios atributos

Nota: La presentación sean utilizando ventanas (Grafica)