

CONCEPTOS FUNDAMENTALES DE PROGRAMACION

PROFESOR:

JAVIER ORLANDO DAZA TORRES

INTEGRANTE DEL GRUPO:

OSWALD GIOVANNI VILLAMIL PALOMA

INSTITUCIÓN UNIVERSITARIA POLITÉCNICO GRANCOLOMBIANO

OCTUBRE DE 2024

```
import java.io.BufferedReader;  
import java.io.BufferedWriter;  
import java.io.FileReader;  
import java.io.FileWriter;  
import java.io.IOException;  
import java.util.Random;
```

```
public class GenerateInfoFiles {
```

```
//Metodo principal
```

```
public static void main (String[] args) {
```

```
//DATOS VENDEDORES
```

```
String [] nombresVendedores = {"GIOVANNI", "SAMUEL","YUMEI"};
```

```
String [] apellidosVendedores = {"VILLAMIL PALOMA", "MARQUEZ CRISTANCHO", "PAEZ  
RUIZ"};
```

```
String [] tipoDocumentos = {"CC", "CC", "CC"};
```

```
String[] numeroDocumentos = {"1888880", "10777770", "10555555"};
```

```
//DATOS DEL PRODUCTO
```

```
int [] idProductos = {101,102,103}; int [][] cantidadesProductos = {  
{12,24,32},  
{2,15,10},  
{20,10,4}  
};
```

```
int [] preciosProductos = {12000000, 67000000, 89000000};
```

```
String [] nombresProductos = {"VEHICULO", "CAMIONETA", "CAMION"};
```

```
//ARREGLO PARA ALMACENAR LAS VENTAS TOTALES POR VENDEDOR
```

```
int [] ventasTotalesPorVendedor = new int[nombresVendedores.length];
```

```
// GENERAR DATOS EN FORMATO CSV
```

```
System.out.println("Generando archivo CSV..."); System.out.println("Datos en formato CSV;");
```

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

```
//CALCULAR VENTAS TOTALES POR VENDEDOR
```

```
System.out.println("Ventas totales por vendedor ordenador de mayor a menor recaudo de  
dinero");
```

```
for (int i = 0; i < nombresVendedores.length; i++) {
```

```

for (int j = 0; j < nombresProductos.length; j++) {

    ventasTotalesPorVendedor[i] += cantidadesProductos[i][j] *
    preciosProductos[j];

}

// Mostrar las ventas totales por vendedor System.out.println(nombresVendedores[i] + ";" +
apellidosVendedores[i] +
";" + " $" + ventasTotalesPorVendedor[i]);
}

System.out.println();

// ORDENAR VENDEDORES POR VENTAS TOTALES (Burbuja)
for (int i = 0; i < ventasTotalesPorVendedor.length - 1; i++) {
    for (int j = 0; j < ventasTotalesPorVendedor.length - 1 - i; j++) {
        if (ventasTotalesPorVendedor[j] < ventasTotalesPorVendedor[j + 1]) {
            // Intercambiar valores

            int tempVentas = ventasTotalesPorVendedor[j]; ventasTotalesPorVendedor[j] =
ventasTotalesPorVendedor[j + 1]; ventasTotalesPorVendedor[j + 1] = tempVentas;

            // Intercambiar datos de vendedores correspondientes String tempNombre =
nombresVendedores[j]; nombresVendedores[j] = nombresVendedores[j + 1];
nombresVendedores[j + 1] = tempNombre;

            String tempApellido = apellidosVendedores[j]; apellidosVendedores[j] =
apellidosVendedores[j + 1]; apellidosVendedores[j + 1] = tempApellido;

            String tempTipoDocumento = tipoDocumentos[j]; tipoDocumentos[j] = tipoDocumentos[j +
1]; tipoDocumentos[j + 1] = tempTipoDocumento;

```

```
String tempNumeroDocumento = numeroDocumentos[j]; numeroDocumentos[j] =  
numeroDocumentos[j + 1]; numeroDocumentos[j + 1] = tempNumeroDocumento;  
}
```

```
}
```

```
}
```

```
//Bloque para abrir el archivo en modo de escritura try (BufferedWriter writer = new  
BufferedWriter(new
```

```
FileWriter("reporte_ventas_vendedores.csv")) {
```

```
writer.write("Nombre;Apellido;Tipo Documento;Numero Documento;Ventas Totales\n"); for  
(int i = 0; i < nombresVendedores.length; i++) {
```

```
writer.write(nombresVendedores[i] + ";" + apellidosVendedores[i] + ";" + tipoDocumentos[i] +  
";" + numeroDocumentos[i] + ";" + ventasTotalesPorVendedor[i] + "\n");
```

```
}
```

```
} catch (IOException e) {
```

```
System.err.println("Error al escribir en el archivo CSV: " + e.getMessage());
```

```
}
```

```
// Calcular la cantidad total vendida de cada producto
```

```
int[] cantidadesTotales = new int[nombresProductos.length]; for (int i = 0; i <  
nombresProductos.length; i++) {
```

```
for (int j = 0; j < cantidadesProductos.length; j++) { cantidadesTotales[i] +=  
cantidadesProductos[j][i];
```

```
}
```

```
}
```

```
// Ordenar los productos por cantidad vendida de manera descendente (usando el método de selección)
```

```
for (int i = 0; i < nombresProductos.length - 1; i++) { int maxIndex = i;
```

```
for (int j = i + 1; j < nombresProductos.length; j++) {
```

```
if (cantidadesTotales[j] > cantidadesTotales[maxIndex]) { maxIndex = j;
```

```
}
```

```
}
```

```
// Intercambiar el nombre, precio y cantidad de los productos String tempNombre =  
nombresProductos[maxIndex]; nombresProductos[maxIndex] = nombresProductos[i];  
nombresProductos[i] = tempNombre;
```

```
int tempPrecio = preciosProductos[maxIndex]; preciosProductos[maxIndex] =  
preciosProductos[i]; preciosProductos[i] = tempPrecio;
```

```
int tempCantidad = cantidadesTotales[maxIndex]; cantidadesTotales[maxIndex] =  
cantidadesTotales[i]; cantidadesTotales[i] = tempCantidad;
```

```
}
```

```
// IMPRIMIR PRODUCTOS VENDIDOS POR CANTIDAD, ORDENADOS EN FORMA  
DESCENDENTE
```

```
System.out.println("Productos vendidos por cantidad, ordenados en forma descendente:");
```

```
for (int i = 0; i < nombresProductos.length; i++) {
```

```
System.out.println(nombresProductos[i] + ";" + " $" + preciosProductos[i] + ";" +  
cantidadesTotales[i]);
```

```
}
```

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

```
// Llamar al método para crear el archivo de productos
```

```
createProductsFile(2); // el numero indica la cantidad de resultados que nos dará
```

```
// Imprimir el contenido del archivo de productos System.out.println("Contenido del archivo  
de productos:"); printProductsFile("productos.csv");
```

```
}
```

```
// Método para crear el archivo de productos con información pseudoaleatoria public static  
void createProductsFile( int productsCount) {
```

```
String[] nombresProductos = {"VEHICULO", "CAMIONETA", "CAMION"};
```

```
try (BufferedWriter writer = new BufferedWriter(new FileWriter("productos.csv"))) {  
writer.write("ID;Nombre;Precio\n");
```

```
Random rand = new Random();
```

```
for (int i = 1; i <= productsCount; i++) {
```

```
int randomIndex = rand.nextInt(nombresProductos.length);
```

```
int precio = rand.nextInt(10000) + 1000; // Precio aleatorio entre 1000 y 11000 writer.write(i +  
";" + nombresProductos[randomIndex] + ";" + precio + "\n");
```

```
}
```

```
} catch (IOException e) {
```

```
System.err.println("Error al escribir en el archivo de productos: " + e.getMessage());
```

```
}
```

```
System.out.println("Archivo de productos generado correctamente.");
```

```
}
```

```
// Método para imprimir el contenido de un archivo de productos public static void  
printProductsFile(String fileName) {  
  
    try (BufferedReader reader = new BufferedReader(new FileReader(fileName))) { String line;  
        while ((line = reader.readLine()) != null) { System.out.println(line);  
        }  
    } catch (IOException e) {  
        System.err.println("Error al leer el archivo de productos: " + e.getMessage());  
    }  
    System.out.println("Archivo CSV generado correctamente.");  
}  
  
}
```