**Actividad 3, “Expresiones lambda y aplanación de colecciones”, Clase 3**

Punto 1:

import java.util.ArrayList;

import java.util.List;

import java.util.stream.Collectors;

/\*\*

\*

\* @author diego vargas

\*/

public class Ejecutora\_Clase\_3\_App {

public static void main(String[] args) {

ArrayList<String> palabras\_min=new ArrayList();

palabras\_min.add("mesa");

palabras\_min.add("cerebro");

palabras\_min.add("cargador");

System.out.println("A continuacion se muestran las palabras almacenadas en la lista, en minúscula:" + " " + palabras\_min);

System.out.println("A continuacion se muestran las palabras almacenadas en la lista, en mayúscuka:" + " " + palabras\_Mayus(palabras\_min));

}

/\*Generamos un método que recibe una lista de palabras en minúscula, que luego las transforma a mayúscula y las devuelve en otro listado, utilizando la función de map\*\*/

public static List<String> palabras\_Mayus(ArrayList<String>palabras){

List<String>mostrar\_palabra\_mayus= palabras.stream().map(String::toUpperCase).collect(Collectors.toList());

return mostrar\_palabra\_mayus;

}

}

Punto 2:

import java.util.ArrayList;

import java.util.List;

import java.util.stream.Collectors;

/\*\*

\*

\* @author diego vargas

\*/

public class Ejecutora\_Clase\_3\_2\_App {

public static void main(String[] args) {

/\* declaramos una lista de palabras y contador de letras\*\*/

List<String> palabras= new ArrayList();

palabras.add("jarron");

palabras.add("silla");

palabras.add("led");

palabras.add("pantalla");

palabras.add("mouse");

int n\_letra=3;

/\* mostramos por pantalla el listado con todas las palabras que almacenamos,

luego mostramos las palabras que almacenamos en el listado en una variable del tipo String, pero concatenadas con un espacio y que contengan más de 3 letras\*\*/

System.out.println("mostando palabras sin concatenar:" + " " + palabras);

System.out.println("mostrando palabras concatenadas, que tienen mas de 3 letras:" + " " + mostrar\_Palabras\_Concatenadas(palabras, n\_letra));

}

/\*declaramos un metodo que recibe un listado de palabras y un contador de letras, el metodo va a devolver un string (para eso,usamos la funcion de joining) con las palabras del listado concatenadas y filtradas,solo las que contienen mas de 3 letras\*\*/

public static String mostrar\_Palabras\_Concatenadas(List<String> palabras,int contador\_letra){

String palabra\_concatenada= palabras.stream().filter(palabra -> palabra.length()> contador\_letra).collect(Collectors.joining(", "));

return palabra\_concatenada;

}

}