

CLASE AppCompiladorLanzador

```
package appdepuradorlanzador;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;

/**
 *
 * @author Roi Iglesias
 */
public class AppCompiladorLanzador {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws IOException, InterruptedException {

        Process pbEjecutaJar;

        /*Compilación..
        //Generamos la clase .class
        Process pbCreaJar;
        pbCreaJar = Runtime.getRuntime().exec("javac "
            + "C:\\Users\\pc\\Desktop\\PSP_ejercicios_preparacion_febrero\\AppGeneradorPalabras\\src\\appgeneradorpalabras\\
AppGeneradorPalabras.java");
        pbCreaJar.waitFor();

        //Para este paso hemos creado un archivo con nombre temp.mf
        //Ese archivo, creado con el bloc de notas le especifica la clase principal
        //Main-Class: AppGeneradorPalabras
        //Sealed: true
        pbCreaJar = Runtime.getRuntime().exec("jar -cf "
            + "C:\\Users\\pc\\Desktop\\PSP_ejercicios_preparacion_febrero\\AppGeneradorPalabras\\src\\appgeneradorpalabras\\
AppGeneradorPalabras.jar AppGeneradorPalabras.class");

        pbCreaJar.waitFor();
        pbCreaJar = Runtime.getRuntime().exec("jar cmf " +
            "C:\\Users\\pc\\Desktop\\PSP_ejercicios_preparacion_febrero\\AppGeneradorPalabras\\src\\appgeneradorpalabras\\MANIFEST.mf
AppGeneradorPalabras.jar AppGeneradorPalabras.class");
        pbCreaJar.waitFor();*/

        //Lanzamos la aplicación Generadora de palabras
        pbEjecutaJar = Runtime.getRuntime().exec("java -jar " +
            "C:\\Users\\pc\\Desktop\\PSP_ejercicios_preparacion_febrero\\AppGeneradorPalabras\\dist\\AppGeneradorPalabras.jar");

        pbEjecutaJar.waitFor();

        //recogemos el resultado...
        String linea = null;
        InputStream is = pbEjecutaJar.getInputStream();
        BufferedReader br = new BufferedReader(new InputStreamReader(is));
        linea = br.readLine();

        //mostramos por pantalla la palabra a analizar
        System.out.println("La palabra a analizar es.. " + linea);

        //Lanzamos aplicación de comprobación de Palíndromos pasándole como argumento la palabra..
        pbEjecutaJar = Runtime.getRuntime().exec("java -jar " +
            "C:\\Users\\pc\\Desktop\\PSP_ejercicios_preparacion_febrero\\AppComprobadorPalindromos\\dist\\AppComprobadorPalindromos.jar " +
            linea );

        pbEjecutaJar.waitFor();

        //recogemos el resultado...
        is = pbEjecutaJar.getInputStream();
        br = new BufferedReader(new InputStreamReader(is));
        linea = br.readLine();
        System.out.println("El comprobador me ha dicho : " + linea);
        /* bucle por si fueran varias lineas
        while((linea = br.readLine())!= null)
            System.out.println(linea);*/

    }
}
```

CLASE AppGeneradorPalabras

```
package appgeneradorpalabras;

import java.util.Random;

/**
 *
 * @author Roi Iglesias
 */
public class AppGeneradorPalabras {

    public static void main(String[] args) {

        String palabras[] = {"orejero", "radar", "oso", "camioneta", "reconocer",
            "sobreverbos", "colchoneta", "sometemos", "sotos", "personaje"};

        System.out.println(generaPalabraAlea(palabras));
    }

    public static String generaPalabraAlea(String[] palabras) {
        String palabraAlAzar;
        Random rnd = new Random();
        palabraAlAzar = palabras[rnd.nextInt(palabras.length)];

        return palabraAlAzar;
    }
}
```

CLASE AppComprobadorPalindromos

```
package appcomprobadorpalindromos;

/**
 *
 * @author Roi Iglesias
 */
public class AppComprobadorPalindromos {

    public static void main(String[] args) {

        String palabra;
        palabra = args[0];

        if (esPalindromo(palabra))
            System.out.println("La palabra " + palabra + " es palindromo");
        else
            System.out.println("La palabra " + palabra + " no es palindromo");
    }

    public static boolean esPalindromo(String cadena) {

        cadena = cadena.toLowerCase().replace("á", "a").replace("é", "e").replace("í", "i").replace("ó", "o")
            .replace("ú", "u").replace(" ", "").replace(".", "").replace(",", "");

        String invertida = new StringBuilder(cadena).reverse().toString();
        return invertida.equals(cadena);
    }
}
```