



TECNOLÓGICO NACIONAL DE MÉXICO  
INSTITUTO TECNOLÓGICO DE LEÓN



**MATERIA**

Lenguajes y Automatas

**CARRERA**

Ingeniería en sistemas computacionales



**PRESENTA: Programa para Identificar un palíndromo**

**NOMBRE DEL ALUMNO**

Emmanuel Jacob Maldonado López

**NOMBRE DEL MAESTRA:**

Ing. Sáez de Nanclares Rodríguez Ruth

LEÓN, GUANAJUATO

Periodo: Agosto-Diciembre 2018

**Programa para identificar si una palabra es palíndromo o no usando estructuras de datos.****Clase Main**

```
package Palindromo;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        //obtenemos la palabra del usuario
        String palabra = s.nextLine();
        //creamos el palindromo
        Palindromo palindromo = new Palindromo(palabra);
        //si es palindromo, lo notificamos
        if(palindromo.esPalindromo()){
            System.out.println("La palabra es un palindromo");
        }else{
            System.out.println("La palabra no es un palindromo");
        }
    }
}
```

**Clase Palíndromo**

```
package Palindromo;
public class Palindromo {
    private String palabra;
    private Pila<Character> pila;

    public Palindromo(String palabra){
        this.palabra = palabra;
        pila = new Pila();
    }

    public boolean esPalindromo(){
        String palabraInicial , palabraInvertida;
        //obtenemos la palabra sin espacios y se inserta a la pila
        palabraInicial = insertarPalabraEnPila();

        //obtenemos la palabra invertida
        palabraInvertida = obtenerPalabraAlrevez();

        //se compara para saber si es palindromo
        return palabraInicial.equals(palabraInvertida);
    }
}
```

```
}

private String insertarPalabraEnPila(){
    //creamos un String para almacenar la palabra sin espacios
    String palabraSinEspacios = "";
    for(int i = 0; i < palabra.length(); i++){
        if(palabra.charAt(i) != ' '){
            //agregamos cada caracter a la pila
            pila.push(palabra.charAt(i));
            //agregamos el caracter al String nuevo
            palabraSinEspacios += palabra.charAt(i);
        }
    }
    return palabraSinEspacios;
}

private String obtenerPalabraAlrevez(){
    //creamos un String y lo inicializamos vacio
    String palabraAlrevez = "";
    while(!pila.isEmpty()){
        //agregamos letra por letra a el nuevo String
        try{
            palabraAlrevez += pila.pop().toString();
        }catch(Exception e){
            System.out.println("Exepcion : " + e.getMessage());
        }
    }
    return palabraAlrevez;
}

public void setPalabra(String palabra) {
    this.palabra = palabra;
}

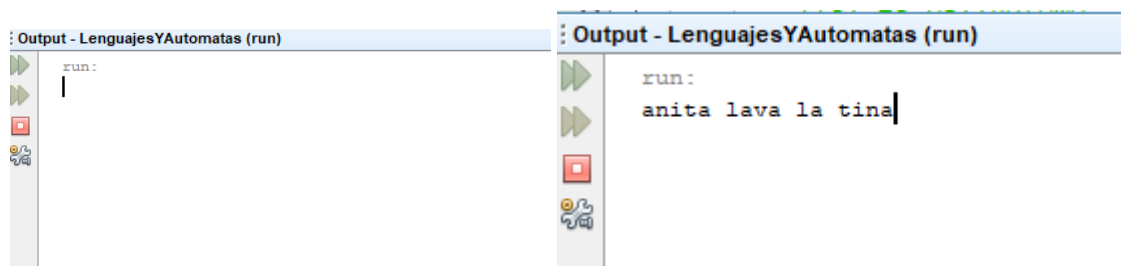
public String getPalabra() {return palabra;}
}
```

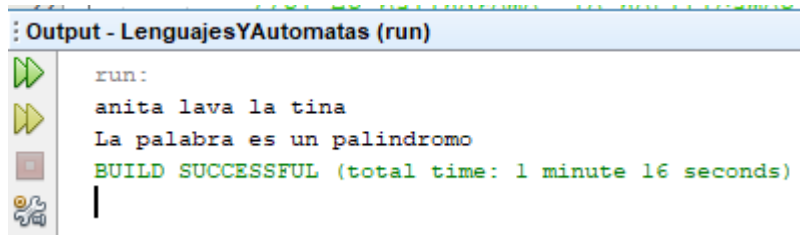
### Clase Pila

```
package Palindromo;
import java.util.Vector;
```

```
public class Pila<E> {  
    private Vector<E> listaPila;  
    private int items;  
  
    public Pila () {  
        listaPila = new Vector();  
        items = -1;  
    }  
  
    public void push(E elemento) {  
        listaPila.add(elemento);  
        items ++;  
    }  
  
    public E pop() throws Exception {  
        //si es inferior a 0, se regresa nulo  
        if(items >= 0){  
            E elemento = listaPila.get(items);  
            items --;  
            return elemento;  
        }  
        //si la pila esta vacia, tira un error  
        new Exception("La pila esta vacia");  
        return null;  
    }  
  
    public boolean isEmpty(){  
        //regresa la condicion  
        return items < 0;  
    }  
}
```

### Corrida





```
run:
anita lava la tina
La palabra es un palindromo
BUILD SUCCESSFUL (total time: 1 minute 16 seconds)
|
```