

# 2a GUIA de POO 2a parte

## (Roberto Tecla Parra)

Nombre: Santos Méndez Ulises Jesús

Grupo: 2CM11

Fecha: 18/06/2021

**Nota** en los problemas de abajo incluir los import que hagan falta

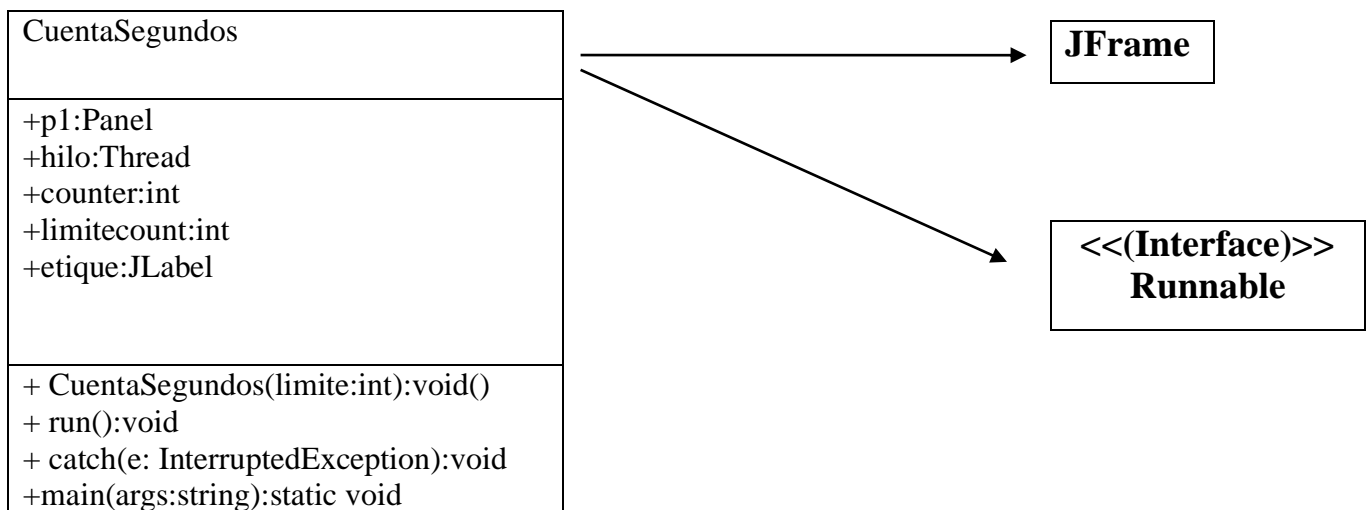
### Hilos

**Problema 1.-** Codificar una clase llamada **CuentaSegundos** que implemente la **interfaz Runnable**:

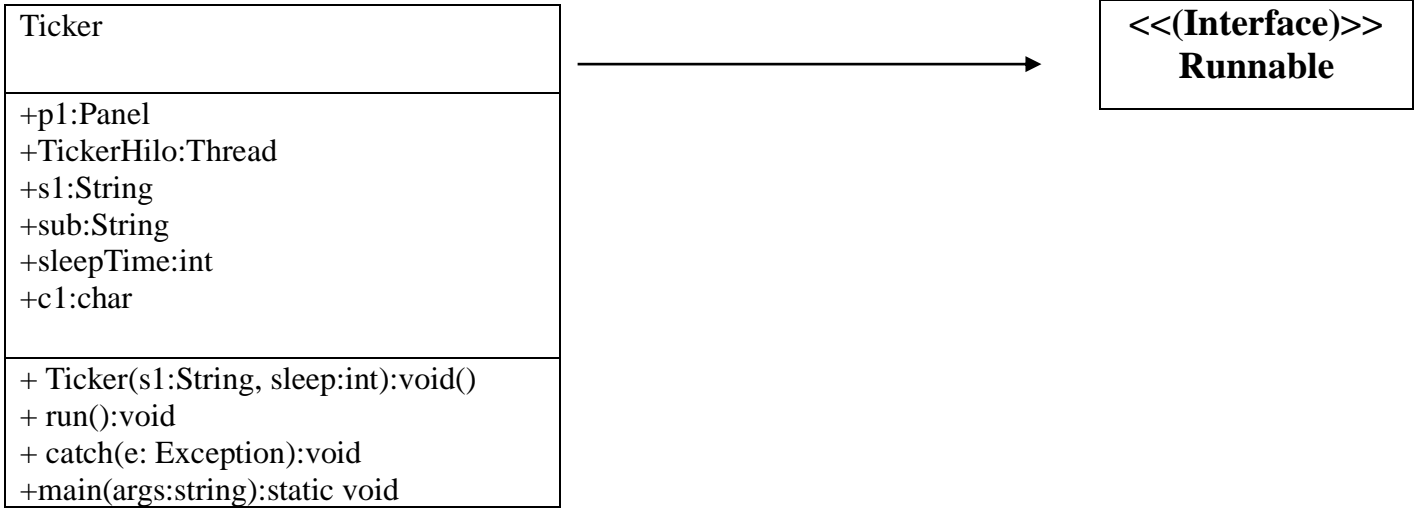
-Que cada segundo incremente en una unidad un contador y muestre el valor de dicho contador en una etiqueta.

-Que cuando llegue a un valor limite (que se establece cuando se crea un objeto) deje de incrementar el contador y de actualizar el valor mostrado.

Sugerencia extienda la clase Label y asigne cero al contador en el constructor.

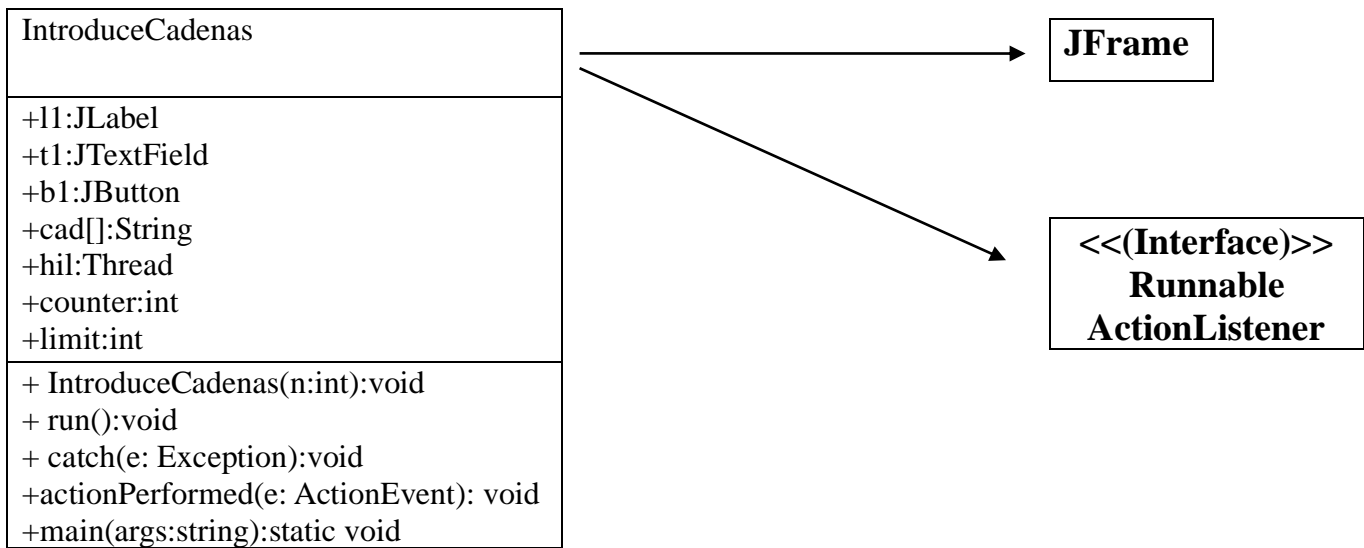


**Problema 2.-**Codificar una clase llamada **Ticker** que mueva las letras de un texto de izquierda a derecha (o de derecha a izquierda). Se puede quitar la primera letra y pegarla (con +) al final y repetir esto de forma periódica.



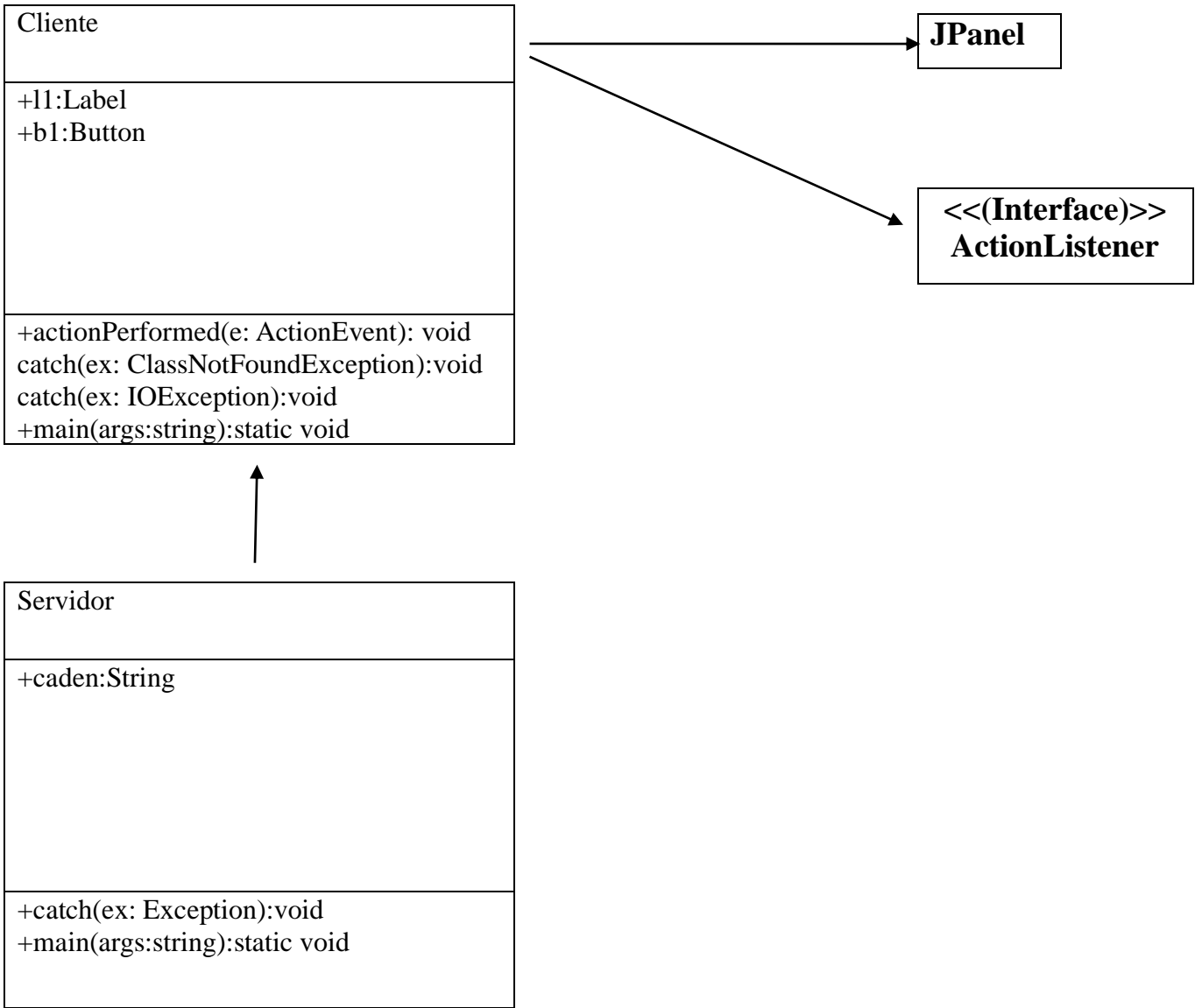
**Problema 3.**-Codificar un programa que permita al usuario introducir varios textos (el usuario escribe un texto en un campo de entrada y dicho texto se almacena cuando el usuario presiona un botón). Guarde los textos en un arreglo de cadenas y use un hilo para que después de un cierto número de segundos el texto actual en una etiqueta se sustituye por el siguiente texto en el arreglo.

Nota: El programa puede ser una aplicación en modo texto y el arreglo donde se almacenan las cadenas puede ser el que se pasa como parámetro al método main y la cadena actual se puede mostrar en consola en lugar de una etiqueta.

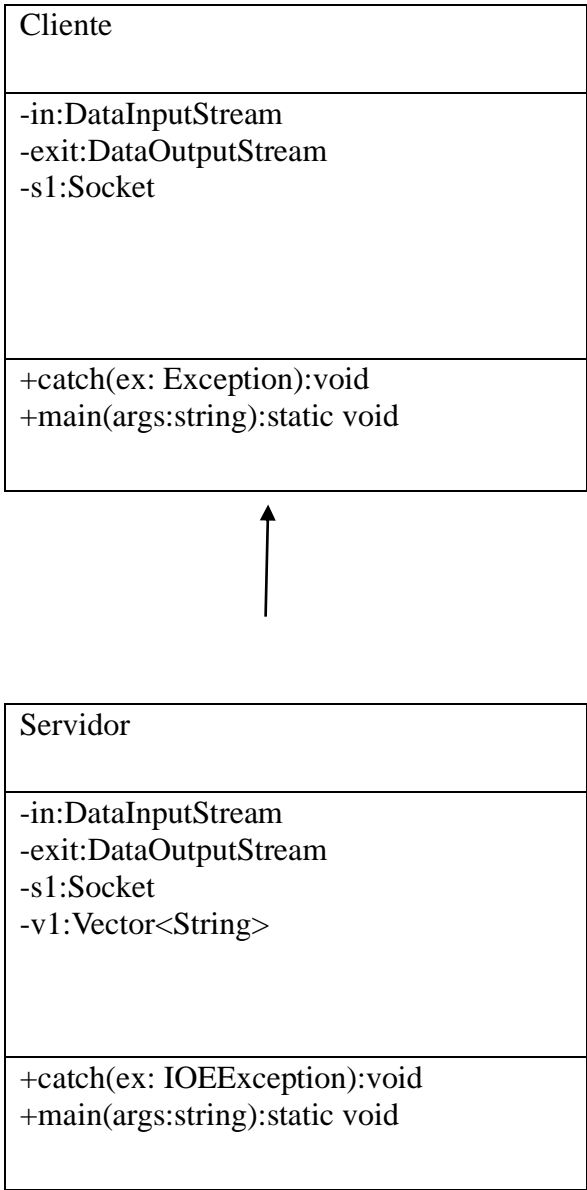


**Sockets**

**Problema 1.**-Codificar un servidor que cuando un cliente se conecte le envié la cadena “Hola Mundo”



**Problema 2.-**Codificar un servidor que cuando un cliente se conecte lea el nombre del cliente y le envíe a dicho cliente el nombre del ultimo usuario que se conectó.



## NOTA: TEMAS YA NO ABORDADOS EN CLASE

### RMI

En cada problema codificar la interfaz remota, la clase del objeto remoto (servidor), la clase del cliente y en los métodos remotos poner los parámetros que se necesiten.

**Problema 1.**-Escribir el código de un método remoto que calcule el **área** de un **triangulo**.

**Problema 2.**-Escribir el código de un método remoto que retorne una cadena indicando si un **punto** esta **dentro o fuera** de un **circulo** de radio 100 con centro en el origen.

pagina 1/3

**Problema 3.**-Escribir el código de un método remoto que calcule el **salario semanal** de un **trabajador** que gana el salario mínimo en base a los horas trabajadas por dicho trabajador durante la semana. Considere 40 horas normales de trabajo a la semana y que las horas extras (las que se trabajan después de transcurridas las 40) se pagan dobles.

**Problema 4.**-Escribir el código de un método remoto que **convierta** una **cadena** a **mayúsculas** y otro método remoto que obtenga la **longitud** de una **cadena**.

**Problema 5.**- Escribir el código de los siguientes métodos remotos: el que calcula el **máximo** de un **arreglo** de enteros, el que calcula el **mínimo** de un arreglo de enteros y el que calcula el **promedio** de un arreglo de enteros

**Problema 6.**- Escribir el código de un chat en rmi y que use callbacks del lado del cliente.

### Javabeans

Para resolver los problemas de abajo tenga en cuenta lo siguiente:

- Un Javabeans tiene que tener un **constructor por defecto** (sin argumentos)
- Un Javabeans tiene que tener **persistencia**, es decir, debe implementar la interfaz *Serializable*

**Problema 1**-Codificar un JavaBean llamado **Viaje** que tenga las siguientes propiedades:  
Origen ( lectura/escritura), Destino ( lectura/escritura), y Costo ( lectura/escritura)

**Problema 2**-Codificar un JavaBean llamado **Pelicula** que tenga las siguientes propiedades:  
Titulo ( lectura/escritura), Director ( lectura/escritura), y Año ( lectura/escritura)

**Problema 3.**-Codificar un JavaBean llamado **Perro** que tenga las siguientes propiedades:  
Nombre ( lectura/escritura), Raza ( lectura/escritura), Edad ( lectura/escritura), Dueño ( lectura/escritura) ,  
Genero ( lectura/escritura), y AñosRestantesDeVida ( solo lectura). Suponga que si la raza es:

Pastor Aleman el perro vive 15 años  
Boxer el perro vive 11 años  
Terrier el perro vive 8 años,  
Doberman el perro vive 13 años  
para las otras razas vive 14 años

**Problema 4.**-En el siguiente JavaBean (**Car**) determinar cuales son sus propiedades y si son de lectura o escritura o ambas cosas.

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
public class Car implements Serializable{
    private String make, model;
    double price;
    public Car() { this ("", "", 0.0); }
    public String getMake() { return make; }
    public void setMake (String m) { make = m; }
    public String getModel() { return model; }
```

pagina 2/3

```
public void setModel (String mo) { model = mo; }
public double getPrice() { return price; }
}
```

## Servlets

Para cada problema de abajo escribir el código **HTML** del **formulario** con los campos de entradas que sean necesarios y el botón de enviar.

1.-Codificar un Servlet que **calcule el área de un círculo** a partir de los datos de un formulario enviados por el navegador y envíe el resultado de dicho calculo al cliente como texto plano o HTML. Escribir el código HTML del formulario con un campo de entrada para el radio y el botón de enviar.

2.-Codificar un Servlet que **calcule el perímetro de un pentágono no regular** a partir de los datos de un formulario enviados por el navegador y envíe el resultado de dicho calculo al cliente como texto plano o HTML Escribir el código HTML del formulario con un campo de entrada para cada lado del pentágono no regular y el botón de enviar.

3.-Codificar un Servlet que envíe al cliente la **capital del país** que el usuario escribió en un formulario. Escribir el código HTML del formulario con un campo de entrada para el pais y el botón de enviar.

4.-Codificar un Servlet que **calcule el promedio de las calificaciones** de un alumno en las siguientes materias: física, matemáticas , química , y español a partir de los datos de un formulario enviados por el navegador y envíe el resultado de dicho calculo al cliente como texto plano o HTML. Escribir el código HTML del formulario con un campo de entrada para cada materia y el botón de enviar.

\*5.-Codificar un Servlet que dibuje una **imagen en memoria** y luego la envíe a un cliente

\*6.-Codificar un Servlet que reciba el nombre de una **imagen** y que lea la imagen de un **archivo** y luego la envíe a un cliente

\*7.-Codificar un Servlet que reciba el nombre de una **imagen** y que obtenga la imagen de una **base de datos** mediante **JDBC** y luego la envíe a un cliente

\*8.-Codificar un Servlet que reciba el nombre de un perro y que obtenga los datos del **perro** (incluyendo su **foto**) de una **base de datos** mediante **JDBC** y luego envíe dichos datos a un cliente

\*9.-Codificar un Servlet que reciba el nombre de un **Mp3** y que lea el Mp3 de un **archivo** y luego lo envíe a un cliente

\*10.-Codificar un Servlet que que reciba el nombre de un Avi , lea el vídeo **Avi** de un **archivo** y luego lo envíe a un cliente y codificar una aplicación cliente que lo reproduzca.